Using Oracle Database Actions for Oracle Cloud
Contents

Preface

Audience x
Documentation Accessibility x
Product Accessibility x
Related Documents x
Conventions xi
Third-Party License Information xi

1 Changes in Oracle Database Actions

2 About Oracle Database Actions

About the Database Actions User Interface 2-1
   Using the Omnisearch Bar 2-3
Accessing Database Actions 2-5
   Enabling User Access to Database Actions 2-5
The Home Page - Launchpad 2-6
About Session State in Database Actions 2-8

3 The SQL and Data Modeler Pages

The Overview Page 3-1
The SQL Page 3-2
   Object Navigator and Files 3-2
   Executing SQL Statements in the Code Editor 3-6
      About Session State in Database Actions 3-9
      Keyboard Shortcuts 3-9
      Support for Multilingual Engine 3-12
   Viewing the SQL Output 3-14
Loading Data 3-17
   Loading Data from a Local File to a New Table 3-17
   Loading Data from a Local File to an Existing Table 3-19
### Format Specifications for JSON, AVRO, and XML Files

- The Data Modeler Page 3-27
  - Navigating Diagrams and Objects 3-27
  - About the Data Modeler Editor 3-29
  - About Viewing JSON Data Guides 3-31
- Creating and Editing Database Objects 3-36
  - The Table Properties Dialog 3-36
    - Columns Pane 3-38
    - Primary Key Pane 3-41
    - Unique Keys Pane 3-41
    - Indexes Pane 3-42
    - Foreign Keys Pane 3-42
    - Table Constraints Pane 3-43
    - Comments Pane 3-43
    - Storage Pane 3-43
    - External Table Properties Pane 3-43
    - Materialized View Pane 3-46
    - DDL Pane 3-48
    - Output Pane 3-48
- The Index Properties Dialog 3-48
- The Sequence Properties Dialog 3-49
- The View Properties Dialog 3-50
- The Synonym Properties Dialog 3-52
- The Materialized View Log Properties Dialog 3-53
- Implied Foreign Keys 3-55

### The REST Pages

- About RESTful Web Services 4-1
- RESTful Services Terminology 4-1
- About the Overview Page 4-2
- About the AutoREST Page 4-3
  - Enabling REST Access for a Database Object 4-4
  - Disabling REST Access for a Database Object 4-4
  - Generating cURL Requests for a REST-Enabled Database Object 4-5
- About the REST Search Toolbar 4-5
- Creating RESTful Web Services 4-6
- Managing Resource Modules 4-6
  - Creating a Resource Module 4-7
  - Importing a Resource Module 4-8
  - Editing a Resource Module 4-8
Deleting a Resource Module 4-8
Publishing/Unpublishing a Resource Module 4-9
Exporting a Resource Module 4-9
Viewing the Module in OpenAPI View 4-9
Managing Resource Templates 4-9
  Creating a Resource Template 4-10
  Editing a Resource Template 4-11
  Deleting a Resource Template 4-11
Managing Resource Handlers 4-11
  Creating a Resource Handler 4-12
  Editing a Resource Handler 4-13
  Deleting a Resource Handler 4-13
Example: Inserting a Record using a POST Handler 4-13
Viewing Resource Handler Details and Managing Parameters 4-16
  Creating a Parameter 4-17
  Editing a Parameter 4-17
  Deleting a Parameter 4-17
  Implicit Parameters 4-18
Securing RESTful Web Services 4-21
Managing Roles 4-22
  Creating a Role 4-22
  Editing a Role 4-22
  Deleting a Role 4-23
  Viewing Assigned Privileges 4-23
Managing Privileges 4-23
  Creating a Privilege 4-23
  Editing a Privilege 4-24
  Deleting a Privilege 4-24
Managing OAuth Clients 4-24
  Creating an OAuth Client 4-25
  Editing an OAuth Client 4-26
  Deleting an OAuth Client 4-26
Example: Creating an OAuth Client 4-26

5 The JSON Page

About the JSON User Interface 5-1
Managing JSON Collections 5-4
  Creating a Collection 5-4
About Adding or Editing a JSON Document 5-5
  About Database Differences in JSON Documents 5-6
About Querying Documents in a Collection 5-7
Using the In-Context Autocomplete Feature in the JSON Editor 5-10
Creating Indexes for JSON Collections 5-11
Creating Relational Views of JSON Documents 5-13
Viewing the JSON Data Guide Diagram for a Collection 5-14

6 The Charts and Dashboards Page
Overview 6-1
Creating or Editing a Chart 6-2
Example: Creating and Editing a Chart 6-3
Creating or Editing a Dashboard 6-6

7 The Scheduling Pages
Overview 7-1
Jobs 7-2
  Summary 7-2
  Running 7-3
  History 7-4
  Forecast 7-4
  Notifications 7-4
  Create or Edit Job 7-5
Chains 7-8
  Create or Edit Chain 7-10
  Add or Edit Step 7-10
  Add or Edit Rule 7-10
Programs 7-11
  Create or Edit Program 7-11
Schedules 7-13
  Create or Edit Schedule 7-13
Objects 7-13
  Create Job Class 7-15
  Create File Watcher 7-15
  Create Window 7-16
  Create Window Group 7-17

8 The Liquibase Page
Generate a Deployment 8-1
# The Data Pump Page

## The Data Load Page

- Checking Data Load Jobs 10-1
- Exploring Data 10-4
- Managing Cloud Storage Connections 10-4
- Loading Data 10-9
  - Loading Data From Local Files 10-9
  - Loading Data from Other Databases 10-12
  - Loading Data from Cloud Storage 10-14
    - Manage Cloud Storage Links for Data Load Jobs 10-15
    - Prepare the Data Load Job 10-15
    - Add Files or Folders for the Data Load Job 10-16
    - Enter Details for the Data Load Job 10-17
    - Run the Data Load Job 10-21
    - View Details About the Data Load Job After It Is Run 10-22
    - View the Table Resulting from the Data Load Job 10-22
- Linking Data 10-23
  - Linking to Other Databases 10-23
  - Linking to Objects in Cloud Storage 10-25
    - Manage Cloud Storage Links for Data Link Jobs 10-25
    - Prepare the Data Link Job 10-26
    - Add Files or Folders for the Data Link Job 10-26
    - Enter Details for the Data Link Job 10-27
    - Run the Data Link Job 10-31
    - View Details About the Data Link Job After It Is Run 10-31
    - View the Table Resulting from the Data Link Job 10-32
- Feeding Data 10-32
  - Creating a Notification-Based Live Table Feed 10-36
    - Creating a Notification-Based Live Table Feed using Amazon Simple Storage Service (S3) 10-39
    - Creating a Notification-Based Live Table Feed using Microsoft Azure 10-45

# The Catalog Page

- About the Catalog Page 11-1
- Sorting and Filtering Entities 11-5
- Searching for Entities 11-6
12 The Data Insights Page

About Insights 12-1
Generate Insights and View Reports 12-2

13 The Data Analysis Tool

Searching and obtaining information about Analytic Views 13-2
Creating Analytic Views 13-3
Using Calculation Templates 13-14
  Cumulative Aggregates 13-14
  Prior and Future Period 13-15
  Period to Date 13-16
  Parallel Period 13-17
  Moving Aggregates 13-17
  Share 13-18
  Rank 13-19
Analyzing data 13-20
Oracle Autonomous Database Add-in for Excel 13-24
  Install the Oracle Autonomous Database add-in on Mac 13-24
  Install the Oracle Autonomous Database add-in on Windows 13-26
  Uninstall the Oracle Autonomous Database add-in 13-28
  Using Oracle Autonomous Database Add-in for Excel 13-28
  Run native SQL queries in an Excel worksheet 13-30
  Query an Analytic View in an Excel worksheet 13-31
  FAQs for Troubleshooting errors with Excel Add-in 13-39

14 The Business Models Page

About Business Models 14-1
Finding and Getting Information about Business Models 14-2
Creating and Editing Business Models 14-3

15 The APEX Workspaces Page

Create a Workspace 15-1
The Database Users Page

16

About Create/Edit User 16-2
About REST Enable User 16-3
About Delete User 16-4

The Monitoring Pages

17

The Monitoring Overview Page 17-1
The Performance Hub Page 17-2
The Alerts Page 17-3
The Sessions Page 17-4
The Storage Page 17-5
The Parameters Page 17-6
The Database Monitor Page 17-7
  Database Monitor Overview 17-7
  Database Monitor Activity 17-12
The AWR Page 17-13

Supported SQL*Plus and SQLcl Commands

A

Supported SQL*Plus Commands A-1
Supported SQLcl Commands A-2
  SODA Commands A-3
Preface

This document provides information about Database Actions, a web-based interface that provides development and administration features for Oracle Database.

Audience

This online help is intended for those using Database Actions in Oracle Autonomous Database.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Product Accessibility

Database Actions provides features to support accessibility. See Oracle Database Actions Accessibility Guide.

Related Documents

Oracle REST Data Services Installation and Configuration Guide
Oracle REST Data Services Developer’s Guide
Oracle Autonomous Database
Oracle SQL Developer User’s Guide

To download release notes, installation documentation, white papers, or other collateral for SQL Developer, go to the Oracle Technology Network (OTN) at http://www.oracle.com/technetwork/

For the PL/SQL page on OTN, see http://www.oracle.com/technetwork/database/features/plsql/
Conventions

The following text conventions are used in this document:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td><em>italic</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td><strong>monospace</strong></td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>

Third-Party License Information

See Third-Party License Information in *Oracle REST Data Services Developer’s Guide*
Changes in Oracle Database Actions

This section lists the changes in this release.

New Features

New features in this release are:

- Support for Multilingual Engine in Database Actions
  See Support for Multilingual Engine
- Create and edit job classes, file watchers, windows and window groups in Scheduling
  See Objects
About Oracle Database Actions

Oracle Database Actions is a web-based interface that uses Oracle REST Data Services to provide development, data tools, administration and monitoring features for Oracle Autonomous Database.

The main features include executing your SQL statements and scripts, creating Data Modeler diagrams, developing RESTful web services, managing JSON collections, and using the Data Load, Catalog, Data Insights, Business Models, and Data Transforms tools to load data from local and remote sources, view data in your tables and views, view objects in your data dictionary, and organize, analyze, and transform your data.

Note:

Some features are only available if you sign in as a user with database administration rights. For such features, a “restricted availability” statement appears at the start of the feature description. For example:

Available only if you signed in as a database user with administrator rights.

Database Actions is also available for download and deployment in your own on-premises Oracle Database or in customer-managed Oracle Database cloud services. For more information, see Oracle Database Actions for On-Premises Oracle Database.

See Also:

- About Create/Edit User for creating users and assigning roles.

About the Database Actions User Interface

This section describes the Database Actions user interface.

The Database Actions user interface has three components:

- The Header at the top
- The page body, whose content varies depending on which page you are viewing
- The Status Bar at the bottom

Header

The header contains the Selector icon, a Search field, the help icon, and the user drop-down list.
• **Selector Icon**

Click the Selector icon to see the main navigation menu slide into view. Click **Oracle Database Actions** in the header to go to the Launchpad page.

• **Search field**

To enter a search term, click in the Search field or use the shortcut key **Ctrl+K** (**Command+K** for Apple computers). For more information, see Using the **Omnisearch Bar**.

• **Help Icon**

Click the help icon to open the contextual or online help for the page you are viewing.

• **User Drop-Down List**

The user drop-down list shows the database user you are signed in as, and provides the following items when you open it:

  – **Preferences**

    The options are:

    **Region**

    * **Language**: Select one of the following languages for the user interface: English, German, Spanish, French, Italian, Japanese, Korean, Portuguese, and Chinese

    * **Timezone**: Select **UTC** or **Local time zone** from the drop-down list.

  **Code Editor**

  Provides global settings that apply to the code editor, such as theme, indentation, font family and line numbers.

  **Worksheet**

  * **SQL History**: Controls whether the history of commands executed in the code editor is enabled in the browser or not.

  – **Log**: Opens a dialog that shows the list of HTTP calls made during your session.

  – **About**: Opens a dialog providing version information for the database and other components as well as copyright and licensing information.

  – **Sign Out**: Signs you out of your database session.

• **Status Bar**

The status bar contains icons that link to log files. The three icons (Errors, Warnings, Processes) are filters that have been applied to the log file.
Errors, Warnings: Displays an Errors or Warnings dialog, which lists log entries from unsuccessful REST calls or from any other problem in the application.

Processes: Displays a Processes dialog, which logs REST calls that are either finished or ongoing.

Log notification link: Displays a Log dialog, containing log entries of the following types: Errors, Warnings, Processes, SQL History and SQL Result.

Using the Omnisearch Bar

You can access the Omnisearch bar (Search field) in the header from any page in Database Actions.

To enter a search term, click in the search field located at the top right of the header, or use the shortcut key Ctrl+K (Command+K for Apple computers).

In the Omnisearch bar, you can filter the search entry by selecting a category (such as tables, views, templates, and so on) from the displayed list.

If you do select a category, you will then need to enter the exact search term.

If you do not select a category, ALL is selected by default. After you enter the search term, a search is performed across all categories. In some cases, the search term is searched across multiple attributes of a category. The following table lists the attributes that are searched for each category:
## Category

<table>
<thead>
<tr>
<th>Category</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tables</td>
<td>Name</td>
</tr>
<tr>
<td>Views</td>
<td>Name</td>
</tr>
<tr>
<td>Indexes</td>
<td>Name</td>
</tr>
<tr>
<td>Packages</td>
<td>Name</td>
</tr>
<tr>
<td>Functions</td>
<td>Name</td>
</tr>
<tr>
<td>Procedures</td>
<td>Name</td>
</tr>
<tr>
<td>Triggers</td>
<td>Name</td>
</tr>
<tr>
<td>Types</td>
<td>Name</td>
</tr>
<tr>
<td>Sequences</td>
<td>Name</td>
</tr>
<tr>
<td>Charts</td>
<td>Name, Comments, URI_Prefix</td>
</tr>
<tr>
<td>Dashboards</td>
<td>Name, Comments, URI_Prefix</td>
</tr>
<tr>
<td>Modules</td>
<td>Name, Comments, URI_Prefix</td>
</tr>
<tr>
<td>Templates</td>
<td>Comments, URI_Prefix</td>
</tr>
<tr>
<td>Handlers</td>
<td>Comments</td>
</tr>
<tr>
<td>Roles</td>
<td>Name</td>
</tr>
<tr>
<td>Privileges</td>
<td>Label, Name, Comments, Description</td>
</tr>
<tr>
<td>OAuth Clients</td>
<td>Name, Description</td>
</tr>
<tr>
<td>Database Users</td>
<td>Username, Alias (alias is used depending on user permissions)</td>
</tr>
<tr>
<td>APEX Workspaces</td>
<td>Workspace, Workspace display name</td>
</tr>
</tbody>
</table>

### Displaying Results for the Search Term

In the results displayed, categories are sorted based on the following criteria:

1. Number of exact matches
2. Most number of matched items

The results within a category are displayed as two groups. The first group consists of items that exactly match the search term. The second group consists of items that contain the search term but do not start with it. Within each group, the items are sorted alphabetically.
You can quickly access previous search terms using the **Recent Searches** list.

## Accessing Database Actions

Database Actions runs in Oracle REST Data Services and access to it is provided through schema-based authentication. To access Database Actions, you must sign in as a database user whose schema has been enabled for Database Actions.

In Oracle Autonomous Database databases, the ADMIN user is pre-enabled. To enable another database user's schema, see [Enabling User Access to Database Actions](#).

## Enabling User Access to Database Actions

To enable a database user to access Database Actions, run the following code as the ADMIN user:

```sql
BEGIN
  ords_admin.enable_schema(
    p_enabled => TRUE,
    p_schema => 'schema-name',
    p_url_mapping_type => 'BASE_PATH',
    p_url_mapping_pattern => 'schema-alias',
    p_auto_rest_auth => NULL
  );
```
where:

- **schema-name** is the database schema name in all-uppercase.
- **schema-alias** is an alias for the schema name that will appear in the URL the user will use to access Database Actions. Oracle recommends that you do not use the schema name itself as a security measure to keep the schema name from being exposed.

After enabling user access, in the Autonomous Database Details page, click **Database Actions**. The Database Actions Launchpad page appears.

### The Home Page - Launchpad

When you log in to Database Actions, the Launchpad page appears.

Launchpad is the home page for Database Actions. It contains six main groups: Development, Data Tools, Administration, Monitoring, Downloads, and Related Services. Each group consists of feature-based icons that you can click to navigate to the respective pages.

The main features are:

#### Development

- **SQL**: Enter and execute SQL and PL/SQL commands, and create database objects. See [The SQL Page](#)
- **Data Modeler**: Create diagrams from existing database schemas, generate DDL statements, and create reports. See [The Data Modeler Page](#)
- **APEX**: Link to the Oracle Application Express sign-in page. Application Express is a rapid web application development platform for the Oracle database. See Oracle Application Express documentation
- **REST**: Develop RESTful web services and ensure secure access. See [The REST Pages](#)
- **JSON**: Manage and query JSON collections. JSON is available only if you are signed in as a database user with the SODA_APP role. See [The JSON Page](#)
- **Scheduling**: Provide details of scheduled jobs, chains, programs and schedules. See [The Scheduling Pages](#)
- **Oracle Machine Learning**: Link to the Oracle Machine Learning sign-in page. See Creating Dashboards, Reports and Notebooks

#### Data Tools

- **Data Pump**: Monitor Data Pump jobs initiated through the available Database API endpoints, the DBMS_DATAPUMP package, or the SQL Developer Data Pump Export and Import wizards. See [The Data Pump Page](#)
- **Data Load**: Load or access data from local files or remote databases. See [The Data Load Page](#)
- **Catalog**: Understand data dependencies and the impact of changes. See [The Catalog Page](#)
• **Data Insights**: Discover anomalies, outliers, and hidden patterns in your data. See The Data Insights Page

• **Data Transforms**: Design your data flows and workflows graphically. Data Transforms is available only to an Oracle Data Integrator on Oracle Cloud Marketplace user that has connectivity enabled from Database Actions in the Oracle Data Integrator user interface. See Using ODI Web Studio

**Administration**

Administration is available only if you are signed in as a database user with administrator rights.

• **Database Users**: Perform user management tasks such as create, edit, and REST enable users. See The Database Users Page

• **APEX Workspaces**: Create and manage APEX workspaces. See Creating Applications with Oracle APEX

• **Download Client Credentials**: Download and use the wallet file to securely connect your existing tools and applications to Autonomous Database. See Download Database Connection Information

• **Set Resource Management Rules**: Set resource management rules to allocate CPU/IO shares to consumer groups and to cancel SQL statements based on their runtime and amount of IO. See Manage Runaway SQL Statements on Autonomous Database and Manage CPU/IO Shares on Autonomous Database

**Monitoring**

Monitoring is available only if you are signed in as a database user with administrator rights.

• **Performance Hub**: Shows the performance data for the specified time period. See The Performance Hub Page

• **Database Monitor**: Provides information about the performance of an Autonomous Database instance. See The Database Monitor Page

**Downloads**

• **Download Oracle Instant Client**: Link to the Oracle Instant Client page on OTN. See Import Data Using Oracle Data Pump on Autonomous Database

• **Download SODA Drivers**: Link to the Oracle JSON Document Database page on OTN. See Simple Oracle Document Access (SODA)

**Related Services**

• **RESTful Services and SODA**: Oracle REST Data Services (ORDS) provides HTTPS interfaces for working with the contents of your Oracle Database in one or more REST enabled schemas.. See Developing RESTful Services in Autonomous Database

• **Oracle Machine Learning RESTful Services**: Oracle Machine Learning provides REST APIs for OML4Py Embedded Python Execution and OML Services.

• **Oracle Database API for MongoDB**: Oracle Database API for MongoDB enables Mongo-DB compatible clients and drivers to connect directly to Autonomous Database. See Using Oracle Database API for MongoDB
About Session State in Database Actions

In Database Actions, a connection to the database is stateless.

In a stateless environment, each HTTPS request from a client maps to a new database session. Therefore, a session begins and ends with every SQL statement or script execution.

As the session state is not maintained, session attributes do not persist and commands such as ROLLBACK and COMMIT do not apply. If a SQL statement or script executes successfully, an implicit commit is performed. If it executes with an error, an implicit rollback is performed.

Therefore, when needed, include the ROLLBACK and COMMIT commands or session attributes in the PL/SQL code block that is sent to the database for a session.
The SQL and Data Modeler Pages

Use the SQL page to enter and execute SQL and PL/SQL statements and create database objects. The Data Modeler page enables you to create diagrams from existing schemas, retrieve data dictionary information, generate DDL statements, and export diagrams.

Topics

• The Overview Page
• The SQL Page
• The Data Modeler Page
• Creating and Editing Database Objects

The Overview Page

The Overview page contains widgets that provide a general overview of the activity in the SQL and Data Modeler pages.

To navigate to the Overview page, click Selector and then select Development.

• My Worksheets: Displays your saved worksheets. You can click the name of the worksheet to open it in the Worksheet page.
• My Diagrams: Displays the Data Modeler diagrams that have been saved. You can click the name of the diagram to open it in the Data Modeler page.
• Recently Modified Objects: Displays a timeline of the created, modified, and dropped objects in the database. You can zoom in and out using the + and – icons. You can also move horizontally by dragging the cursor to the right or left.
• Invalid Objects: Displays the invalid objects in your schema.
• Table Stats Freshness: Displays the time period since the tables were last analyzed.

You can right-click the header in Invalid Objects, Table Stats Freshness, My Worksheets, or My Diagrams to manage or sort columns:

• Columns: Enables you to select columns to show or hide.
• Sort: Displays a dialog box for selecting columns to sort by. For each column, you can specify ascending or descending order, and you can specify that null values be displayed first.

Right-click the body of the display table to count rows or to view records:

• Count Rows: Displays the number of rows in the table.
• Single Record View: Enables you to view data for a table or view, one record at a time.
• Copy:Copies data from a cell or a row or a range of rows. To copy from more than one row, select the rows you want to copy, right-click by pressing the SHIFT or CTRL key, and select Copy.
The SQL Page

The SQL page enables you to enter and execute SQL and PL/SQL statements, and create database objects.

To navigate to the SQL page, do either of the following:

- In the Launchpad page, click SQL.
- Click Selector to display the navigation menu. Under Development, select SQL.

You can use SQL and PL/SQL statements in the worksheet to create a table, insert data, create and edit a trigger, select data from a table, and save that data to a file. Some other features are syntax highlighting and error detection.

The SQL page consists of the left pane for navigating worksheets and objects, the editor for executing SQL statements, and the output pane for viewing the results. These panes are described in the following sections:

- Object Navigator and Files
- Executing SQL Statements in the Code Editor
  - Keyboard Shortcuts
- Viewing the SQL Output
- Loading Data

Object Navigator and Files

The Navigator tab in the left pane displays saved objects for the selected schema. The Files tab enables you to view and open files saved in your browser or local system.

The following figure shows the various elements in the left pane.
Figure 3-1  Left Pane in SQL

Navigator Tab
Displays saved objects for the selected schema.

- **Schema and Object Type selector**: Use the drop-down lists to select the schema and filter the results by object type.
- **Search**: Searches the contents of a saved worksheet or search for objects in the Navigator tab by name. The search functionality is not case-sensitive, retrieves all matching entries, and does not require the use of wild card characters.
- **Context menu**: Options in the context menu are:
  - **Open** to browse properties and data relevant to the object type (tables and views).
    The Data pane displays the data for a table, view, or materialized view.
    To edit an entry, double-click a cell to make edits. You can also click 🖋 and enter the value. When you make an edit, the border of the gutter cell in that row changes to blue.
    The icons available in the Data pane are:
      - **Create Row**: Insert a new row into the database table. When you insert a row using the Create Row icon, the row is committed into the database.
      - **Delete Selected**: Mark the selected row for deletion. When you mark a row for deletion, the border of the row changes to red.
      - **Commit**: Commit all changes made to the database.
* **Undo All**: Revert all changes that are marked for commit.

Use the **Filter** icon at the top right corner to filter the column data. You can also right-click a cell to access the context menu to count rows, view a single record, export or copy the cell text to clipboard.

To view a Binary Large Object data type (BLOB), click the pencil icon for a BLOB data type. In the View Value dialog:

* The **Image** tab displays the loaded image, if the loaded BLOB type is an image.
* The **Text** tab displays the text file, if the loaded BLOB type is text.
* The **Information** tab displays the details and allows you to perform the following actions:
  * **Download**: To download the image/text file of BLOB data type.
  * **Load**: To insert an image/text of BLOB data type.
  * **Set NULL**: To delete the object and set the value as NULL.

  ![View Value Dialog](image)

  - **Edit** edits the properties of an existing object.
  - **Add** creates an object based on the object type selected.
  - **Use as Template** creates an object by using the properties of an existing object as the template.

- **REST**
  * **Enable** enables REST access for the database object. See [Enabling REST Access for a Database Object](#)
  
  A REST enabled object is indicated by a **REST Enabled** icon 🏛 in the Navigator tab.
  *
  * **Disable** disables REST access for the database object after it is enabled. See [Disabling REST Access for a Database Object](#)
  *
  * **cURL Command** generates a cURL call for a selected HTTP method for the database object. See [Generating cURL Requests for a REST-Enabled Database Object](#)

  - **Data Loading** loads data from local files into a table.
  - **Quick DDL** generates Data Definition Language statements for the object.
• **Refresh** ː Refreshes the objects or worksheets listed in the left pane.

• **Object Submenu** ː Opens the Create Object dialog to create a new object based on the type selected in the drop-down list.

• **Help** ː Provides contextual help documentation.

**Files Tab**

Enables you to open files from your browser or local device.

**Note:**

The DEVICE category in the left pane is displayed only when using a Chromium-based browser in a secure context (HTTPS).

---

**All files**: Use the drop-down list to filter files by browser or device.

The context menu options for a file are **Open** and **Delete**. In Device, the corresponding option for deleting a file is **Forget**. In this case, the file is not deleted, instead the reference to the file is removed.

**Drag and Drop Objects and Files into the Worksheet**

You can drag objects from the left pane and drop them into the worksheet editor in the right pane.

• If you drag and drop a table or view, you are prompted to select one of the following SQL statements: Insert, Update, Select, or Delete. For example, if you choose Select, a Select statement is constructed with all columns in the table or view. You can then edit the statement, for example, modifying the column list or adding a WHERE clause.
Figure 3-2 Insert Select Query

If you choose Object Name, the name of the object prefixed by the schema name is added to the worksheet.

- If you drag and drop a function or procedure, you can choose to insert the name or the PL/SQL code of the function or procedure in the worksheet. If you select the PL/SQL code, you can enter the parameters before inserting the code into the worksheet.

Executing SQL Statements in the Code Editor

The code editor in the SQL page enables you to run SQL statements, PL/SQL scripts, and JavaScript code. The main features include in-context code completion, syntax highlighting, and error debugging.

You can enter SQL and PL/SQL statements to specify actions such as creating a table, inserting data, selecting data, or deleting data from a table. SQL keywords are automatically highlighted. For multiple statements, you must terminate:

- Each non-PL/SQL statement with either a semicolon (;) or a slash (/) on a new line
- Each PL/SQL statement with a slash (/) on a new line

The **PL/SQL editor** is triggered when opening the following object types: Functions, Procedures, Packages and Types. This editor helps you detect errors in your PL/SQL code during compilation. The output includes details such as the specific line and column where the error is detected, along with a link to go to the relevant position in the code block.

The **JavaScript worksheet mode** supports the Multilingual Engine syntax in Oracle Database release 21c. For more details, see Support for Multilingual Engine.

For SQL*Plus and SQLcl statements supported in the worksheet, see Supported SQL*Plus and SQLcl Commands.

Use the Consumer Group drop-down list to select the consumer group to run your SQL or PL/SQL code. The values in the drop-down list match the database services available when connecting to the database. This feature is available if you have the following privileges:

- The **ALTER SESSION** privilege to change the consumer group
- The **SELECT privilege on V_$SERVICES**

---

**Chapter 3**

**The SQL Page**

---

3-6
• The **SELECT** privilege on DBA_RSRC_CONSUMER_GROUP_PRIVS
• The **EXECUTE** privilege on DBMS_SESSION

If you press **Ctrl+Space**, the editor provides you with a list of possible completions at the insertion point that you can use to autocomplete code that you are editing. This list is based on the code context at the insertion point.

The editor offers a comprehensive list of commands available through the Command Palette. To open the Command Palette, press **Ctrl+Shift+P**. For a list of keyboard shortcut keys, see **Keyboard Shortcuts**.

An error in the code is signified by a red dot in the left gutter and a squiggle line beneath the specific text. When you hover over it, you see a pop-up displaying possible fixes for resolving the error.
You can set code editor preferences using the Preferences option available in the top-right user drop-down list. Some of the code editor options that you can customize are Theme (Light, Dark and High contrast dark), Font size and family, Tab size, Word wrap, Ruler, Line numbers and so on.

The SQL toolbar contains icons for the following operations:

- **Worksheet** drop-down list
  - **Open**: Opens a file from the browser or device.
  - **Open Recent**: Displays the recently accessed files. If there are more than five files in the recent list, then a **More** link is displayed for viewing the additional files.
  - **Save As**: Saves a file to the browser or device.

- **New File**
  - Select **Worksheet** to create a worksheet.
  - Select **PL/SQL** to create a PL/SQL object type. The editor switches to a PL/SQL mode.
  - Select **JavaScript** to create a JavaScript file. The editor switches to a JavaScript mode and (**JS**) is added to the file name.

- **Open** opens a file from your browser or device. To open a file from your device, in the Open File slider, click **Open File** and browse to select the file, or drag and drop the file into the slider.

- **Run Statement** executes the selected statements or the statement at the mouse pointer in the worksheet editor. The SQL statements can include bind variables and substitution variables of type VARCHAR2 (although in most cases, VARCHAR2 is automatically converted internally to NUMBER if necessary). A dialog box is displayed for entering variable values.

- **Run Script**
- **Run as SQL Script** executes all statements in the worksheet editor using the Script Runner. The SQL statements can include bind variables (but not substitution variables) of type VARCHAR2 (although in most cases, VARCHAR2 is automatically converted internally to NUMBER if necessary). A dialog box is displayed for entering bind variable values.

- **Run as JavaScript** executes the code as a JavaScript file. This option is used only with JavaScript code. If the JavaScript code is added to a PL/SQL block, then select “Run as SQL Script” to execute the script.

  - **Compile** (for PL/SQL toolbar) performs a PL/SQL compilation of the subprogram.
  - **Create Chart** creates a chart for the corresponding SQL statement entered in the editor. In contrast, you can partially highlight a subquery and create a chart. A slider window is displayed for entering the chart parameters. For a description of the fields, see Creating or Editing a Chart.
  - If the SQL statement is syntactically incorrect or incomplete, an error/warning notification is displayed.
  - **Explain Plan** generates the execution plan for the statement (internally executing the EXPLAIN PLAN statement). The execution plan is automatically displayed in the Explain Plan tab in the worksheet output pane.
  - **Autotrace** runs the statement and collects run time statistics and the actual execution plan. The Autotrace output is displayed in the Autotrace tab in the worksheet output pane. Currently, there are no preferences available.
  - **Download Editor Content** downloads the content of the worksheet as a SQL file to the local system.
  - **Format** formats the SQL statement in the editor, such as capitalizing the names of statements, clauses, keywords, and adding line breaks and indentation.
  - **Clear** removes the statements from the editor.
  - **Tour** provides a guided tour of the worksheet highlighting salient features and providing information that is useful if you are new to the interface.
  - **Help** provides context-related help and provides a link to the help documentation.
  - **Open in Fullscreen** opens the editor in full screen mode.

**About Session State in Database Actions**

In Database Actions, a connection to the database is stateless.

In a stateless environment, each HTTPS request from a client maps to a new database session. Therefore, a session begins and ends with every SQL statement or script execution.

As the session state is not maintained, session attributes do not persist and commands such as ROLLBACK and COMMIT do not apply. If a SQL statement or script executes successfully, an implicit commit is performed. If it executes with an error, an implicit rollback is performed.

Therefore, when needed, include the ROLLBACK and COMMIT commands or session attributes in the PL/SQL code block that is sent to the database for a session.

**Keyboard Shortcuts**

This section lists the keyboard shortcuts for various commands in the SQL page.
### Table 3-1  Keyboard Shortcuts

<table>
<thead>
<tr>
<th>Windows</th>
<th>MacOS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ctrl + Enter</td>
<td>Cmd + Enter</td>
<td>Runs the code as query.</td>
</tr>
<tr>
<td>Alt + Tab</td>
<td>Option + Tab</td>
<td>Focus next element.</td>
</tr>
<tr>
<td>Ctrl + Esc / Escape</td>
<td>Cmd + Esc / Escape</td>
<td>Remove focus from editor.</td>
</tr>
<tr>
<td>Ctrl + Down Arrow</td>
<td>Cmd + Down Arrow</td>
<td>Moves to the next SQL code from history.</td>
</tr>
<tr>
<td>Ctrl + Up Arrow</td>
<td>Cmd + Up Arrow</td>
<td>Moves to the previous SQL code from history.</td>
</tr>
<tr>
<td>Ctrl + S</td>
<td>Cmd + S</td>
<td>Saves the current worksheet.</td>
</tr>
<tr>
<td>Ctrl + O</td>
<td>Cmd + O</td>
<td>Opens the worksheet browser dialog.</td>
</tr>
<tr>
<td>Ctrl + I</td>
<td>Cmd + I</td>
<td>Downloads the content of the editor.</td>
</tr>
<tr>
<td>F1</td>
<td>Fn + F1</td>
<td>Opens the help topic.</td>
</tr>
<tr>
<td>Shift + Esc</td>
<td>Shift + Esc</td>
<td>Focus previous element.</td>
</tr>
<tr>
<td>F5</td>
<td>Fn + F5</td>
<td>Runs code as script.</td>
</tr>
<tr>
<td>F6</td>
<td>Fn + F6</td>
<td>Shows Autotrace.</td>
</tr>
<tr>
<td>F10</td>
<td>Fn + F10</td>
<td>Shows Explain Plan.</td>
</tr>
<tr>
<td>F11</td>
<td>Fn + F11</td>
<td>Creates a chart.</td>
</tr>
<tr>
<td>Ctrl + B</td>
<td>Cmd + B</td>
<td>Opens the &quot;Convert Case&quot; drop-down list.</td>
</tr>
<tr>
<td>Ctrl + F7</td>
<td>Cmd + Fn + F7</td>
<td>Formats code in the editor.</td>
</tr>
<tr>
<td>Ctrl + Space</td>
<td>Ctrl + Space</td>
<td>Autocompletes code (shows hints).</td>
</tr>
</tbody>
</table>

The following table lists the keyboard shortcuts for commands in the Command Palette.

### Table 3-2  Command Palette Keyboard Shortcuts

<table>
<thead>
<tr>
<th>Windows</th>
<th>MacOS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ctrl + Alt + Up</td>
<td>Cmd + Option + Up</td>
<td>Add Cursor Above</td>
</tr>
<tr>
<td>Ctrl + Alt + Down</td>
<td>Cmd + Option + Down</td>
<td>Add Cursor Below</td>
</tr>
<tr>
<td>Shift + Alt + I</td>
<td>Shift + Option + I</td>
<td>Add Cursors to Line Ends</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + C</td>
<td>Cmd + K Cmd + C</td>
<td>Add Line Comment</td>
</tr>
<tr>
<td>Ctrl + D</td>
<td>Cmd + D</td>
<td>Add Selection To Next Find Match</td>
</tr>
<tr>
<td>Shift + Alt + Down</td>
<td>Shift + Option + Down</td>
<td>Copy Line Down</td>
</tr>
<tr>
<td>Shift + Alt + Up</td>
<td>Shift + Option + Up</td>
<td>Copy Line Up</td>
</tr>
<tr>
<td>Ctrl + U</td>
<td>Cmd + U</td>
<td>Cursor Undo</td>
</tr>
<tr>
<td></td>
<td>Cmd + Backspace</td>
<td>Delete All Left</td>
</tr>
<tr>
<td></td>
<td>Ctrl + K</td>
<td>Delete All Right</td>
</tr>
<tr>
<td>Ctrl + Shift + K</td>
<td>Shift + Cmd + K</td>
<td>Delete Line</td>
</tr>
<tr>
<td>Shift + Alt + Right</td>
<td>Shift + Ctrl + Cmd + Right</td>
<td>Expand Selection</td>
</tr>
<tr>
<td>Ctrl + F</td>
<td>Cmd + F</td>
<td>Find</td>
</tr>
</tbody>
</table>
### Table 3-2  (Cont.) Command Palette Keyboard Shortcuts

<table>
<thead>
<tr>
<th>Windows</th>
<th>MacOS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enter</td>
<td>Enter</td>
<td>Find Next</td>
</tr>
<tr>
<td>Ctrl + F3</td>
<td>Cmd + Fn + F3</td>
<td>Find Next Selection</td>
</tr>
<tr>
<td>Shift + Enter</td>
<td>Shift + Enter</td>
<td>Find Previous</td>
</tr>
<tr>
<td>Ctrl + Shift + F3</td>
<td>Shift + Cmd + Fn + F3</td>
<td>Find Previous Selection</td>
</tr>
<tr>
<td></td>
<td>Cmd + E</td>
<td>Find With Selection</td>
</tr>
<tr>
<td>Ctrl + Shift + [</td>
<td>Option + Cmd + [</td>
<td>Fold</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 0</td>
<td>Cmd + K Cmd + 0</td>
<td>Fold All</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + /</td>
<td>Cmd + K Cmd + /</td>
<td>Fold All Block Comments</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 8</td>
<td>Cmd + K Cmd + 8</td>
<td>Fold All Regions</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 1</td>
<td>Cmd + K Cmd + 1</td>
<td>Fold Level 1</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 2</td>
<td>Cmd + K Cmd + 2</td>
<td>Fold Level 2</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 3</td>
<td>Cmd + K Cmd + 3</td>
<td>Fold Level 3</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 4</td>
<td>Cmd + K Cmd + 4</td>
<td>Fold Level 4</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 5</td>
<td>Cmd + K Cmd + 5</td>
<td>Fold Level 5</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 6</td>
<td>Cmd + K Cmd + 6</td>
<td>Fold Level 6</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 7</td>
<td>Cmd + K Cmd + 7</td>
<td>Fold Level 7</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + [</td>
<td>Cmd + K Cmd + [</td>
<td>Fold Recursively</td>
</tr>
<tr>
<td>Ctrl + Shift + \</td>
<td>Shift + Cmd + \</td>
<td>Go to Bracket</td>
</tr>
<tr>
<td>Ctrl + G</td>
<td>Ctrl + G</td>
<td>Go to Line...</td>
</tr>
<tr>
<td>Alt + F8</td>
<td>Option + Fn + F8</td>
<td>Go to Next Problem (Error, Warning, Info)</td>
</tr>
<tr>
<td>F8</td>
<td>Fn + F8</td>
<td>Go to Next Problem in Files  (Error, Warning, Info)</td>
</tr>
<tr>
<td>F7</td>
<td>Fn + F7</td>
<td>Go to Next Symbol Highlight</td>
</tr>
<tr>
<td>Shift + Alt + F8</td>
<td>Shift + Option + Fn + F8</td>
<td>Go to Previous Problem (Error, Warning, Info)</td>
</tr>
<tr>
<td>Shift + F8</td>
<td>Shift + Fn + F8</td>
<td>Go to Previous Problem in Files (Error, Warning, Info)</td>
</tr>
<tr>
<td>Shift + F7</td>
<td>Shift + Fn + F7</td>
<td>Go to Previous Symbol Highlight</td>
</tr>
<tr>
<td>Ctrl + ]</td>
<td>Cmd + ]</td>
<td>Indent Line</td>
</tr>
<tr>
<td>Ctrl + Shift + Enter</td>
<td>Shift + Cmd + Enter</td>
<td>Insert Line Above</td>
</tr>
<tr>
<td></td>
<td>Ctrl + J</td>
<td>Join Lines</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + D</td>
<td>Cmd + K Cmd + D</td>
<td>Move Last Selection To Next</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Find Match</td>
</tr>
<tr>
<td>Alt + Down</td>
<td>Option + Down</td>
<td>Move Line Down</td>
</tr>
<tr>
<td>Alt + Up</td>
<td>Option + Up</td>
<td>Move Line up</td>
</tr>
<tr>
<td>F1 (All browsers)</td>
<td>Fn + F1</td>
<td>Open Command palette</td>
</tr>
<tr>
<td>Ctrl + Shift + P (Google Chrome only)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ctrl + [</td>
<td>Cmd + [</td>
<td>Outdent Line</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + U</td>
<td>Cmd + K Cmd + U</td>
<td>Remove Line Comment</td>
</tr>
<tr>
<td>Ctrl + H</td>
<td>Option + Cmd + F</td>
<td>Replace</td>
</tr>
<tr>
<td>Ctrl + Shift + .</td>
<td>Shift + Cmd + .</td>
<td>Replace with Next Value</td>
</tr>
<tr>
<td>Ctrl + Shift + .</td>
<td>Shift + Cmd + .</td>
<td>Replace with Previous Value</td>
</tr>
</tbody>
</table>
Table 3-2  (Cont.) Command Palette Keyboard Shortcuts

<table>
<thead>
<tr>
<th>Windows</th>
<th>MacOS</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ctrl + Shift + L</td>
<td>Shift + Cmd + L</td>
<td>Select All Occurrences of Find Match</td>
</tr>
<tr>
<td>Alt + F1</td>
<td>Option + Fn + F1</td>
<td>Show Accessibility Help</td>
</tr>
<tr>
<td>Shift + F10</td>
<td>Shift + Fn + F10</td>
<td>Show Editor Context Menu</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + L</td>
<td>Cmd + K Cmd + I</td>
<td>Show Hover</td>
</tr>
<tr>
<td>Shift + Alt + Left</td>
<td>Shift + Ctrl + Cmd + Left</td>
<td>Shrink Selection</td>
</tr>
<tr>
<td>Shift + Alt + A</td>
<td>Shift + Option + A</td>
<td>Toggle Block Comment</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + L</td>
<td>Cmd + K Cmd + L</td>
<td>Toggle Fold</td>
</tr>
<tr>
<td>Ctrl + /</td>
<td>Cmd + /</td>
<td>Toggle Line Comment</td>
</tr>
<tr>
<td>Ctrl + M</td>
<td>Shift + Ctrl + M</td>
<td>Toggle Tab Key Moves Focus</td>
</tr>
<tr>
<td>Ctrl + Space</td>
<td>Ctrl + Space</td>
<td>Trigger Suggest</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + X</td>
<td>Cmd + K Cmd + X</td>
<td>Trim Trailing Whitespace</td>
</tr>
<tr>
<td>Ctrl + Shift + ]</td>
<td>Option + Cmd + ]</td>
<td>Unfold</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + J</td>
<td>Cmd + K Cmd + J</td>
<td>Unfold All</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + 9</td>
<td>Cmd + K Cmd + 9</td>
<td>Unfold All regions</td>
</tr>
<tr>
<td>Ctrl + K Ctrl + ]</td>
<td>Cmd + K Cmd + ]</td>
<td>Unfold Recursively</td>
</tr>
</tbody>
</table>

Support for Multilingual Engine

Database Actions provides support for Multilingual Engine (MLE) by enabling you to run JavaScript code in the worksheet.

Prerequisites

For the availability of MLE features in Database Actions, you need the:

- DBMS_MLE package in Oracle Database Release 21c and later versions. For more information, see DBMS_MLE in Oracle Database PL/SQL Packages and Types Reference.
- EXECUTE DYNAMIC MLE and EXECUTE ON JAVASCRIPT privileges assigned to you.

You can work with JavaScript code in the worksheet in the following ways:

- Create a JavaScript worksheet
- Execute JavaScript code in a standard worksheet
- Execute JavaScript code as a PL/SQL block

Create a JavaScript Worksheet

You can open the worksheet in JavaScript mode. You will also see the toolbar icons and output tabs change to reflect the JavaScript mode.

To create and save a JavaScript worksheet:

1. In the toolbar, expand the New File icon and select JavaScript.
2. When you enter code in the worksheet, the JavaScript code is automatically highlighted.
3. Execute the JavaScript code in the worksheet using Run Script.

4. Click Save.
   In the Files pane, (JS) is added to the JavaScript file name. This enables you to quickly identify the JavaScript file.

---

Execute JavaScript code in a standard worksheet

In a standard worksheet (when the worksheet is not in JavaScript mode):

1. Select the JavaScript code to execute.
2. In the worksheet toolbar, expand the Run Script icon and select Run as JavaScript.

**Note:**

Selecting Run as SQL script results in an error.
Execute JavaScript code as a PL/SQL block

You can use a PL/SQL code block to execute JavaScript code.

From the worksheet toolbar, expand the Run Script icon and select Run as SQL script (F5).

Viewing the SQL Output

The lower right pane in SQL displays the output of the operation executed in the SQL editor.

The following figure shows the output pane in the SQL page.
The output pane has the following tabs:

- **Query Result**: Displays the results of the most recent Run Statement operation in a display table.
- **Script Output**: Displays the text output from your statements executed as a script using the script engine.
- **DBMS Output**: Displays the output of DBMS_OUTPUT package statements.
- **Explain Plan**: Displays the plan for your query using the Explain Plan command.
- **Autotrace**: Displays the session statistics and execution plan from `v$sql_plan when executing a SQL statement using the Autotrace feature. Displays the output if you clicked the Autotrace icon.
- **SQL History**: Displays the SQL statements and scripts that you have executed. To re-enter a previously executed query in the worksheet, double-click the query in the history list. You can search for specific statements by clicking the Search icon. The Search functionality is case-sensitive, retrieves all entries that contain the search text, and does not require wildcard characters.
- **Data Loading**: Displays a report of the total rows loaded and failed for all visible tables (including tables from other schemas).

The icons in this pane are:

- **Clear output**: Clears the output.
- **Show info**: Displays the SQL statement for which the output is displayed.
- **Open in new tab**: Opens the query result or explain plan in a new window.
- **Download**: This is applicable only for Query Result. Enables you to download the query result to your local computer in CSV, JSON, XML, or TEXT (.tsv) format.
In the Query Result tab, in the display table, the context menu (right-click) for the row header consists of the following:

- **Columns** enables you to select columns to hide.
- **Sort** displays a dialog box for selecting columns to sort by. For each column, you can specify ascending or descending order, and you can specify that null values be displayed first.

**Figure 3-4  Context Menu for Row Header**

The context menu for the rest of the display table consists of the following commands:

- **Count Rows** displays the number of rows in the result set for your query.
- **Single Record View** enables you to view data for a table or view, one record at a time.
- **Export** generates the file for download based on the format selected, which can be XML, CSV (comma-separated values including a header row for column identifiers), Insert, Delimited, Fixed, HTML, JSON, or TEXT.
  - **Format**: Select the format to export from the drop-down list.
  - **Line Terminator**: Identifies the terminator for each line. The line terminator is not included in the data exported. If the preview page shows the data in one single row, the correct terminator is not specified.
  - **Header**: Controls whether the first row is a header row or the first row of data.
  - **Left and Right Enclosure**: Enclosures are used for character data and are optional. Enclosures are not included in the data exported.

**Note:**

If a popup blocker is enabled, it will prevent the file from downloading.

- **Copy** copies data from a cell or a row or a range of rows.

**Figure 3-5  Context Menu**
Loading Data

In the SQL page, you can load data from a local file into an existing table or into a new table.

The file formats that you can load are CSV, XLS, XLSX, TSV, TXT, XML, JSON, and AVRO. For XML, JSON, and AVRO files, see Format Specifications for JSON, AVRO, and XML Files.

• Loading Data from a Local File to a New Table
• Loading Data from a Local File to an Existing Table

Loading Data from a Local File to a New Table

To load data from a local file to a new table:

1. You can start in one of the following ways:
   • In the Navigator tab, in the left pane, click Object submenu, select Data Loading, and then select Upload Data into New Table.

   ![Figure 3-6   Upload Data to New Table Option](image)

   • In the Navigator tab, drag and drop the local file into the left pane. When you drag a file into the pane, the following message is displayed: Drop the file here to start.

   • In the worksheet output pane, select the Data Loading tab and drag and drop the local file into the output pane.

   ![Figure 3-7   Data Loading Tab](image)
The Upload Data into New Table is displayed. A preview of the data is displayed in a grid format.

2. Click **Show/Hide options** to display options that you can modify for data preview:
   - **Column names**: Select **Get from file** to display column headers in the first row.
   - **Encoding**: An option to select the encoding type is visible when the loaded file is in plain text format (CSV, TSV, or TXT). The default encoding type is UTF-8.
   - **Text enclosure and Field delimiter**: These options are visible only when the selected file is in plain text format (CSV, TSV, or TXT). Select or enter the character used in the source file for text enclosure and field delimiter.
   - **Rows to skip**: Enter or use the up and down arrows to select the number of rows to skip.
   - **Preview size**: Enter or use the up and down arrows to select the number of rows to preview.
   - **Limit rows to upload**: If you select this option, you need to specify the rows to load. Use the up and down arrows to select the number of rows to load.

To remove the options selected and the data preview, click **Clear**.

After selecting the required options, click **Apply**, and then click **Next**.

3. In Table Definition, do the following:
   - In the **Table Name** field, enter a name for the target table.
   - Select the check box at the beginning of a row to add the column to the target table.
   - Select or enter values for column attributes such as Column Name, Column Type, Precision, Scale, Default, Primary Key and Nullable.
   - The Format Mask column appears for date, timestamp and numeric types of data. For date and timestamp types, you must select a value from the drop-down list or type the value in the Format Mask field. For numeric types, the format mask is optional.

   For a date and timestamp column, you need to supply a compatible format mask that describes the data being uploaded. For example, if the date data looks like `12-FEB-2021 12.21.30`, you need to supply a date mask of `DD-MON-YYYY HH.MM.SS`. The format mask is automatically determined based on the data in the file. You need to review the suggested format mask and if needed, modify it by entering the format directly into the target cell.
4. Review the generated DDL code based on the selections made in the previous screens. The mapping of the source to target columns are also displayed.

Click Finish. After the data is successfully loaded, the new table is displayed in the Navigator tab.

5. For a detailed report of the total rows loaded and failed, do one of the following:
   - Right-click the table in the Navigator tab, select Data Loading, and then select History. This displays the report for a specific table.
   - In the Navigator tab, select Object submenu **Object**, select Data Loading, and then select History. This displays the report for all tables in the schema that is selected in the Navigator tab.
   - In the worksheet output pane, select the Data Loading tab. This displays the report for all visible tables (including tables from other schemas).

A summary of the data loaded is displayed in the History dialog. If any data failed to load, you can view the number of failed rows in the Failed Rows column. Click the failed rows column to open a dialog showing the failed rows.

In the History dialog, you can also search for files loaded by schema name, table name, or file name. To remove the loaded files, click Remove all history.

### Loading Data from a Local File to an Existing Table

To load data from a local file to an existing table:

1. In the Navigator tab, in the left pane, right-click the table that you want to load data into, select Data Loading, and then select Upload Data.
Figure 3-9  Upload Data Option for an Existing Table

The Import data dialog is displayed.

2. Drag and drop the file from your system into the dialog, or click Select Files to browse for the file and open it.

A preview of the data is displayed in a grid format.

3. Click Show/Hide options to display options that you can modify for data preview:
   
   • **Column names**: Select Get from file to display column headers in the first row.
   
   • **Encoding**: An option to select the encoding type is visible when the loaded file is in plain text format (CSV, TSV, or TXT). The default encoding type is UTF-8.
   
   • **Text enclosure** and **Field delimiter**: These options are visible only when the selected file is in plain text format (CSV, TSV, or TXT). Select or enter the character used in the source file for text enclosure and field delimiter.
   
   • **Rows to skip**: Enter or use the up and down arrows to select the number of rows to skip.
   
   • **Rows to load**: Enter or use the up and down arrows to select the number of rows to load.
   
   • **Preview size**: Enter or use the up and down arrows to select the number of rows to preview.

To remove the options selected and the data preview, click Clear.

After selecting the required options, click Apply, and then click Next.
4. In Data mapping, match the data in the file to the appropriate columns in the target table. By default, the matching is done using column name.

**Figure 3-10  Data Mapping**

To modify, click **Show/Hide options**. In Match columns by:

- **Select Name** to match columns based on the name of the column in the target table.
- **Select Position** if you want to match columns based on the position of the column in the target table.
- **Select None** to remove the current selections and to select the target column for each source column from the drop-down list.

**Note:**

Based on the data in the file, attempts are made to automatically retrieve the correct format mask of date-based columns. If this is incorrect, you can change the suggested format by entering it directly into the target cell.

If there are any issues to resolve, you see a notification such as **1 pending actions** on the top right of the dialog.

Click **Next**.

5. A summary of the previous screens is displayed. Click **Finish**.

The data will start uploading to the target table. After it is completed, an entry is added to the Log with the status of the operation. To view the Log, click the timestamp notification at the bottom of the page. If the operation is successful, a **Data Import Completed** notification is displayed.
6. For a detailed summary of the upload process, right-click the table in the Navigator tab, select **Data Loading**, and then select **History**. A summary of the data loaded is displayed in the Data Loading History dialog.

If any data failed to load, you can view the number of rows in the Failed Rows column. Click the column to open a dialog showing the failed rows.

In the Data Loading History dialog, you can also search for files loaded by schema name, table name, or file name. To remove the loaded files, click Remove all history.

### Format Specifications for JSON, AVRO, and XML Files

Data has to be stored in a particular format for JSON, AVRO, and XML files to load them successfully into a table.

The format specifications are described in the following sections.

#### JSON and AVRO Files

For JSON and AVRO files, the conversion for primitive types to table columns is supported only for top-level data. Nested objects are saved as JSON strings such as VARCHAR2 (JSON) or CLOB (JSON).

---

**Note:**

JSON check constraints are available only for Oracle Database 12c and later releases.

---

Consider the following JSON file as an example:

```json
[
  {
    "ItemNumber": 1,
    "Description": "One Magic Christmas",
    "Part": {
      "UnitPrice": 19.95,
      "UPCCode": 13131092899
    },
    "Quantity": 9,
    "Total": 179.55
  },
  {
    "ItemNumber": 2,
    "Description": "Lethal Weapon",
    "Part": {
      "UnitPrice": 17.95,
      "UPCCode": 85391628927
    },
    "Quantity": 5,
    "Total": 89.75
  }
]"
The AVRO schema for this file:

```json
{
    "type": "array",
    "items": {
        "type": "record",
        "fields": [
            {
                "name": "ItemNumber",
                "type": "int"
            },
            {
                "name": "Description",
                "type": "string"
            },
            {
                "name": "Part",
                "type": {
                    "type": "record",
                    "fields": [
                        {
                            "name": "UnitPrice",
                            "type": "float"
                        },
                        {
                            "name": "UPCCode",
                            "type": "float"
                        }
                    ]
                }
            },
            {
                "name": "Quantity",
                "type": "int"
            },
            {
                "name": "Total",
                "type": "float"
            }
        ]
    }
}
```

Load the JSON file using "Upload Data" in the SQL page, and it is converted to the following table with two rows. `part` is a nested object that is assigned the column type CLOB (JSON) during data mapping.
XML Files

This section lists the specifications for loading XML files.

- Attributes will have their own columns

  If the XML data is structured as:

  ```xml
  <?xml version="1.0"?>
  <catalog>
    <book id="bk102">
      <author>Ralls, Kim</author>
      <title>Midnight Rain</title>
      <genre>Fantasy</genre>
      <publisher>John Doe</publisher>
    </book>
  </catalog>
  ```

  The generated columns in the table are **id, author, title, genre, and publisher**.

- Two or more levels of nesting are needed to parse the data

  In the following example, the data that needs to be parsed will not be located because it has only one level of nesting (catalog).

  ```xml
  <?xml version="1.0"?>
  <catalog>
    <author>Ralls, Kim</author>
    <title>Midnight Rain</title>
    <genre>Fantasy</genre>
    <publisher>John Doe</publisher>
  </catalog>
  ```

  However, the following examples will work:

  ```xml
  <?xml version="1.0"?>
  <catalog>
    <book id="bk102">
      <author>Ralls, Kim</author>
      <title>Midnight Rain</title>
      <genre>Fantasy</genre>
      <publisher>John Doe</publisher>
    </book>
  </catalog>
  ```
or

```xml
<?xml version="1.0"?>
<catalog>
  <bookstore>
    <book id="bk102">
      <author>Ralls, Kim</author>
      <title>Midnight Rain</title>
      <genre>Fantasy</genre>
      <publisher>John Doe</publisher>
    </book>
  </bookstore>
</catalog>
```

or

```xml
<?xml version="1.0"?>
<catalog>
  <bookstore>
    <shelf>
      <book id="bk102">
        <author>Ralls, Kim</author>
        <title>Midnight Rain</title>
        <genre>Fantasy</genre>
        <publisher>John Doe</publisher>
      </book>
    </shelf>
  </bookstore>
</catalog>
```

- Special characters such as hyphen (-) and period (.) in tag names are replaced by underscore (_) in the column name

XML tag names can contain hyphens and periods. Since the parser is converting XML to JSON, these characters become invalid object keys.

```xml
<?xml version="1.0"?>
<catalog>
  <book id="bk102">
    <author-name>Ralls, Kim</author-name>
    <title.1>Midnight Rain</title.1>
    <genre>Fantasy</genre>
    <publisher>John Doe</publisher>
  </book>
</catalog>
```

The generated columns are `id, author_name, title_1, genre, and publisher`.

- First-level only-text tags are ignored

```xml
<?xml version="1.0"?>
<catalog>
  <library> New Age Library </library>
  <book id="bk102">
```
The  tag is ignored and only the content of the  tag is taken into account. The generated columns are id, author, title, genre, and publisher.

- First-level repetitive data is interpreted as an array of values

```xml
<?xml version="1.0" encoding="UTF-8"?>
<items id="orders">
  <item_number>1</item_number>
  <description>One Magic Christmas</description>
  <part>
    <unit_price>19.95</unit_price>
    <upccode>13131092899</upccode>
  </part>
  <quantity>9</quantity>
  <total>179.55</total>
  <item_number>2</item_number>
  <description>Lethal Weapon</description>
  <part>
    <unit_price>17.95</unit_price>
    <upccode>85391628927</upccode>
  </part>
  <quantity>5</quantity>
  <total>89.75</total>
</items>
```

The generated columns include item_number, description, part, and each column will have only one row with the following values respectively ([1,2], ["One Magic Christmas","Lethal Weapon"], [{"unit_price":19.95,"upccode":13131092899}, {"unit_price":17.95,"upccode":85391628927}] and so on for the remaining columns.

- Tags containing values and attributes are converted to object

```xml
<?xml version="1.0"?>
<catalog>
  <book id="bk102">
    <author country="ca">Ralls, Kim</author>
    <title>Midnight Rain</title>
    <genre>Fantasy</genre>
    <publisher>John Doe</publisher>
  </book>
</catalog>
```
The <author> tag is converted to a column and will have an object as value that is structured as:

```json
{
   "_":"Ralls, Kim",
   "country":"ca"
}
```

Note that the value of the tag has underscore ("_") as key and the attributes as "attribute_name": "attribute_value".

The Data Modeler Page

The Data Modeler page provides an integrated version of Oracle SQL Developer Data Modeler with basic reporting features. You can create diagrams from existing schemas, retrieve data dictionary information, generate DDL statements, and export diagrams.

To navigate to the Data Modeler page, do either of the following:

- In the Launchpad page, click **Data Modeler**.
- Click **Selector** to display the navigation menu. Under Development, select **Data Modeler**.

The Data Modeler page consists of the left pane for navigating objects and diagrams, the editor pane for working with relational diagrams, and the right pane for viewing the properties of the object selected in the diagram. These panes are described in the following sections:

- Navigating Diagrams and Objects
- About the Data Modeler Editor

Related Topics

- **Oracle SQL Developer Data Modeler User’s Guide**

Navigating Diagrams and Objects

The **Diagrams** tab lists the Data Modeler diagrams that have been saved.

When you right-click a diagram, you have options to open, save, delete, and view properties. **Figure 3-11** shows a thumbnail preview that is displayed when you click the name of a diagram in the left pane.
Figure 3-11  Diagram Preview in Data Modeler

To create a new diagram, in the Diagrams tab, click New Diagram.
To open an existing diagram, in the Diagrams tab, right-click the diagram and click Open.

Note:

Diagrams are stored in the database, in the Database Actions user schema. When you use Data Modeler for the first time, a OSDDMW_DIAGRAMS table is created in your schema, to store the diagrams.

Oracle SQL Developer Data Modeler can import from or export to the OSDDMW_DIAGRAMS table if a connection is provided. For more information, see Sharing Diagrams with SQL Developer Data Modeler.

The Navigator tab lists the objects that are available for each schema. You can select the schema and object type from the drop-down lists. For a particular schema, if a tables object is selected, the left pane displays all the tables that belong to the particular schema. You can expand a table to view its columns. Figure 3-12 indicates the important elements in the Navigator tab.

In a schema, you can search for objects in the Navigator tab by entering a few characters. You can also search the contents of a saved diagram. The search
functionality is not case-sensitive, retrieves all matching entries, and does not require the use of wildcard characters.

**Figure 3-12   Navigator Tab**

The context menu for a table or view consists of:

**Add Object to Diagram:** Adds the selected object to the selected diagram. Alternatively, you can drag and drop an object into a selected diagram.

**Add Object with dependencies to Diagram:** Adds the selected object along with parent and child tables related to the object to the selected diagram.

**Add Object as Star Schema to Diagram:** Adds the selected object to the diagram and searches the data dictionary for foreign keys and implied foreign keys related to the object. The related tables or views are added to the diagram and the star schema layout is applied. See [Implied Foreign Keys](#).

**Edit, Use as Template:** Opens the Table Properties Dialog. Use to edit an existing object (such as a table or view) for a specific schema, or create a new object by using an existing one for the initial content. See [The Table Properties Dialog](#) and [The View Properties Dialog](#).

**Add/Edit Sequence:** Opens the Sequence Properties Dialog. Use to create or edit a sequence for a selected schema. See [The Sequence Properties Dialog](#).

**Add Index:** Opens the Index Properties Dialog. Use to create an index for a table. See [The Index Properties Dialog](#).

**Add Synonym:** Opens the Synonym Properties Dialog. Use to create a synonym for a table or view in a selected schema. See [The Synonym Properties Dialog](#).

**About the Data Modeler Editor**

You can create and work with relational diagrams in the editor pane.
When an object is selected in the diagram, you can inspect the properties of the object in the right pane. For a table, the properties displayed are Columns, Primary Key, Unique Constraints, Indexes, Foreign Keys, Comments, and Colors. The only properties that you can edit are background and border color. Click Reset to return to the default colors.

Figure 3-13 shows the properties of an object selected in the editor pane.

You can drag the thumbnail viewer in the upper right corner to display the appropriate area in the editor pane. This is useful when you are working with long diagrams in the editor pane.

Figure 3-13  Inspect Properties of Object Selected in Editor Pane

When creating diagrams, you can do the following:

- Select an object and then drag to move it around.
- Adjust or move objects with the relationships intact.
- Add elbows to relationship lines to avoid intersecting with lines from other objects. Right-click the relationship line and drag to create the elbow. Click Remove vertex to restore the original shape.
- Resize objects by dragging the handles that are positioned around the box.
- Select and then right-click an object for the following options:
  - **DDL Preview**: Shows the DDL statements for the object.
  - **Update**: Updates any actions performed on the object.
  - **Delete**: Deletes the object.
  - **Edit**: Opens the Table Properties Dialog for editing a table object or the View Properties Dialog for a view object.
  - **Implied Foreign Keys**: Opens the Implied Foreign Keys Dialog. Use this option to define implied foreign keys for the object. See Implied Foreign Keys
  - **View JSON Data Guides**: Opens the JSON data guide diagram for the table or view. See About Viewing JSON Data Guides
The icons on the toolbar are:

- **Save Diagram**: Saves the currently selected diagram. Diagrams are stored in a table that is created in the schema of the user.
- **Print Diagram**: Prints the selected diagram.
- **Save to SVG**: Saves the currently selected diagram to an image file in SVG format.
- **Add Objects to Diagram**: A dialog is displayed where you can select one or more objects of type tables or views from a specific schema into the selected diagram. Type `*` to list all the objects of an object type (tables or views) in the schema. You can also search by typing a few characters.
- **Add Note**: Adds notes to the selected diagram. Select the note to add text and to see the associated properties in the right pane, such as Font Size and Colors.
  - To enter text, expand TEXT and type information. Click **Apply**.
  - To select the background, border, or text color for the note, click the box and select the required color. To revert to the default color, click **Reset**.
- **Delete**: Deletes the selected object or objects from the diagram. To select multiple objects, press the CTRL key and select the objects.
- **Zoom In**: Displays more detail, and potentially fewer objects, in the currently selected diagram.
- **Zoom Out**: Displays less detail, and potentially more objects, in the currently selected diagram.
- **Fit Screen**: Makes all relevant objects fit in the window for the currently selected diagram, adjusting the sizes of shapes and text labels as needed.
- **Actual Size**: Adjusts the shapes and text labels in the currently selected diagram to the default sizes.
- **Layout: Auto Layout**: Rearranges the objects in the diagram to a layout that may be more meaningful and attractive. If you do not like the rearrangement, you can restore the previous layout by clicking **Undo**.
- **Layout: Star Layout**: Rearranges the objects in the diagram to a star schema layout, where the fact table is in the center and the associated dimension tables surround the fact table.
- **DDL Preview**: Shows the DDL statements for the object. You have the option to save or send the DDL statements to the worksheet. To choose what you want to display, click **Options**.
- **DDL Preview for Current Schema**: Shows the DDL statements for the current schema. You can send the DDL statements to the worksheet.
- **Diagram Report**: Generates a data dictionary report of everything in the diagram.
- **Schema Report**: Generates a data dictionary report of everything in the schema.
- **Help**: Displays the help for the Data Modeling editor.

**About Viewing JSON Data Guides**

A JSON data guide represents the JSON schema for documents in one column with JSON content. A table can have more than one column with JSON content.
The View JSON Data Guides option is available in the context menu for a table or view in a Data Modeler diagram.

This is applicable for tables, views and external tables that have columns with JSON content. Columns with JSON content are identified and data guides are retrieved for each such column. This process is quicker with the existence of a JSON search index.

The JSON schema (JSON Data Guide) is presented visually like an entity-relationship diagram. Arrays are presented as one-to-many relationships, contained objects as one-to-one relationships, and "oneOf" constructs as a box that surrounds possible choices. There is a column selector at the top right part of the page enabling you to select a column with JSON content for diagram presentation.

The following example is a JSON schema and its representation in a data guide diagram.

```json
{
  "type" : "object",
  "properties" : {
    "User" : {
      "type" : "string",
      "o:length" : 8,
      "o:preferred_column_name" : "DATA$User"
    },
    "PONumber" : {
      "type" : "number",
      "o:length" : 4,
      "o:preferred_column_name" : "DATA$PONumber"
    },
    "LineItems" : {
      "type" : "array",
      "o:length" : 1024,
      "o:preferred_column_name" : "DATA$LineItems"
    }
  }
}
```
"items":
{
  "properties":
  {
    "Part":
    {
      "type": "object",
      "o:length": 256,
      "o:preferred_column_name": "DATA$Part",
      "properties":
      {
        "UPCCode":
        {
          "type": "number",
          "o:length": 16,
          "o:preferred_column_name": "DATA$UPCCode"
        },
        "UnitPrice":
        {
          "type": "number",
          "o:length": 8,
          "o:preferred_column_name": "DATA$UnitPrice"
        },
        "Description":
        {
          "type": "string",
          "o:length": 128,
          "o:preferred_column_name": "DATA$Description"
        }
      }
    },
    "Quantity":
    {
      "type": "number",
      "o:length": 4,
      "o:preferred_column_name": "DATA$Quantity"
    },
    "ItemNumber":
    {
      "type": "number",
      "o:length": 1,
      "o:preferred_column_name": "DATA$ItemNumber"
    }
  }
},
"Reference":
{
  "type": "string",
  "o:length": 32,
  "o:preferred_column_name": "DATA$Reference"
},
"Requestor":
{
  "type": "string"}
"o:length" : 16,
"o:preferred_column_name" : "DATA$Requestor"
},
"CostCenter" : {
  "type" : "string",
  "o:length" : 4,
  "o:preferred_column_name" : "DATA$CostCenter"
},
"ShippingInstructions" : {
  "type" : "object",
  "o:length" : 256,
  "o:preferred_column_name" : "DATA$ShippingInstructions",
  "properties" : {
    "name" : {
      "type" : "string",
      "o:length" : 16,
      "o:preferred_column_name" : "DATA$name"
    },
    "Phone" : {
      "type" : "array",
      "o:length" : 64,
      "o:preferred_column_name" : "DATA$Phone",
      "items" : {
        "properties" : {
          "type" : {
            "type" : "string",
            "o:length" : 8,
            "o:preferred_column_name" : "DATA$type"
          },
          "number" : {
            "type" : "string",
            "o:length" : 16,
            "o:preferred_column_name" : "DATA$number"
          }
        }
      }
    },
    "Address" : {
      "oneOf" : [
        {
          "type" : "string",
          "o:length" : 1,
          "o:preferred_column_name" : "DATA$Address"
        }
      ]
    }
  }
}
{  
  "type": "object",
  "o:length": 128,
  "o:preferred_column_name": "DATA\$Address_1",
  "properties": {
    "city": {
      "type": "string",
      "o:length": 32,
      "o:preferred_column_name": "DATA\$city"
    },
    "state": {
      "type": "string",
      "o:length": 2,
      "o:preferred_column_name": "DATA\$state"
    },
    "street": {
      "type": "string",
      "o:length": 32,
      "o:preferred_column_name": "DATA\$street"
    },
    "country": {
      "type": "string",
      "o:length": 32,
      "o:preferred_column_name": "DATA\$country"
    },
    "zipCode": {
      "type": "number",
      "o:length": 8,
      "o:preferred_column_name": "DATA\$zipCode"
    }
  }
}
Creating and Editing Database Objects

You can create and edit objects using Create and Edit Object wizards available from the Navigator tab in the SQL and Data Modeler pages.

The wizards for creating and editing various object types are described in the following sections:

- The Table Properties Dialog
- The Index Properties Dialog
- The Sequence Properties Dialog
- The View Properties Dialog
- The Synonym Properties Dialog
- Implied Foreign Keys
- The Materialized View Log Properties Dialog

The Table Properties Dialog

The Table Properties Dialog is displayed when you create a table, edit an existing table, or create a table using an existing one as a template.

You can open the Table Properties dialog from the Navigator tab in SQL or Data Modeler.
To create a table for a specific schema, in the Navigator tab, select **Tables** from the object type drop-down list, click **Object submenu** ..., and select **Create Object**.

**Figure 3-15  Create Object**

To create a table from an existing one for a specific schema, right-click the table object in the Navigator tab, and select **Use as Template** as shown in **Figure 3-16**.

To edit a table for a specific schema, right-click a table object in the Navigator tab, and select **Edit**.

**Figure 3-16  Edit Option for Existing Table**

The table properties are grouped in several panes.
If you are editing an existing table, you can visit the panes in any order. If you click Create before you finish creating the table, right-click the table name, select Edit, and continue creating the table.

**Note:**
Editing a partitioned table is not recommended. To identify whether a table is partitioned or not, right-click the table name and select Edit. If the table is partitioned, a warning message will be displayed.

**Schema:** Database schema in which to create the table. By default, a new table is created in the existing schema or the schema that you are logged into.

**Name:** Name for the table.

The different panes in the dialog are described in the following sections:

- Columns Pane
- Primary Key Pane
- Unique Keys Pane
- Indexes Pane
- Foreign Keys Pane
- Table Constraints Pane
- Comments Pane
- Storage Pane
- External Table Properties Pane
- Materialized View Pane
- DDL Pane
- Output Pane

**Related Topics**
- *Oracle Database SQL Language Reference*

**Columns Pane**

Specifies properties for each column in the table.

**General tab**

Lists the columns available in the table.

To add a column, click **Add Column (+)**. A new row is added to the table below. Select the row and enter the details for the column.

To delete a column, select the row and click **Remove Column (-)**. To move a column up or down in the table, select it and use the up-arrow and down-arrow icons.
Figure 3-17   Add a Column

- **Name**: Name for the column.
- **Datatype**: Data type for the column.
- **Default**: If no value is specified, the default value inserted into the column when a row is inserted.
- **Default on NULL**: Applicable for Oracle Database 12c and later releases. If this option is selected, when a row is inserted into the table and the value specified for the column is NULL, the default value is inserted into the column.
- **Expression**: Expression for computing the value in the column.
- **Comments**: Optional descriptive comments about the column. Use this field to provide descriptions for the attributes.

In the table:
- **PK**: If this option is selected, the column becomes the primary key.
- **Identity Column**: If this option is selected, the column becomes an identity column. This is applicable only for Oracle Database 12c and later releases. For more details, see the Identity Column tab.

**Constraints tab**

Displays the Not Null and Check Constraints for a column. A check constraint requires values in a column to comply with a specified condition.

- **Not Null Constraint: Name**: Name for the Not Null constraint.
• **Not Null Constraint: Not Null**: If this option is selected, the column must contain data. You cannot specify no value or an explicit null value for this column when you insert a row. If this option is not checked, the column can contain either data or no data. A primary key column cannot be null.

• **Check Constraint: Name**: Name for the check constraint definition.

• **Check Constraint: Constraint**: Condition that must be met for a column to fulfill the check constraint. You can use any valid CHECK clause (without the CHECK keyword). For example, to indicate that the value in a numeric column named RATING must be from 1 to 10, you can specify: rating >=1 and rating <= 10.

• **Enabled**: If this option is selected, the constraint is checked when data is entered or updated in the column.

• **Deferrable**: If this option is selected, you can defer checking the validity of the constraint until the end of a transaction.

• **Initially Immediate**: If this option is selected, the constraint is checked whenever you add, update, or delete data from the column.

• **Validate**: If this option is selected, the existing data is checked to see if it conforms to the constraint.

### Identity Column tab

Applicable for Oracle Database 12c and later releases. The Identity Column tab lists the properties of the identity column. This tab becomes available only after the Identity Column checkbox is selected for the column in the General tab. An identity column is an autoincrement column that can be used to identify a table row. Only one identity column can be specified for a table.

• **Generate**: Always means that values cannot be explicitly included for the identity column in INSERT OR UPDATE statements, By Default means values for the identity column are generated automatically if no values are specified explicitly, By Default on Null means values are generated for the column only when a NULL value is supplied.

• **Start with**: Starting value of the sequence.

• **Increment**: Interval between successive numbers in a sequence.

• **Min value**: Lowest possible value for the sequence. The default is 1 for an ascending sequence and -(10^26) for a descending sequence.

• **Max value**: Highest possible value for the sequence. The default is 10^27 for an ascending sequence and -1 for a descending sequence.

• **Cache and Cache size**: Cache causes sequence values to be preallocated in cache, which can improve application performance; Cache size indicates the number of sequence values preallocated in cache. No Cache causes sequence values not to be preallocated in cache.

• **Cycle**: Indicates whether the sequence "wraps around" to reuse numbers after reaching its maximum value (for an ascending sequence) or its minimum value (for a descending sequence). If cycling of values is not enabled, the sequence cannot generate more values after reaching its maximum or minimum value.

• **Order**: Indicates whether sequence numbers are generated in the order in which they are requested. If No Order is specified, sequence numbers are not guaranteed to be in the order in which they were requested.
Primary Key Pane

Specifies the primary key for the table.

The primary key is the column, or set of columns, that uniquely identifies each row in the table. If the Primary Key checkbox is selected for a column in the General tab, the corresponding fields are automatically populated in the Primary Key pane. You can make changes to the properties as required.

An index is automatically created on the primary key.

- **Name**: Name of the constraint to be associated with the primary key definition.
- **Enabled**: If this option is checked, the primary key constraint is enforced: that is, the data in the primary key column (or set of columns) must be unique and not null.
- **Index**: Name of the index to which the primary key refers.
- **Tablespace**: Name of the tablespace associated with the index.
- **Available Columns**: Lists the columns that are available to be added to the primary key definition. You can select multiple attributes, if required, for the primary key.
- **Selected Columns**: Lists the columns that are included in the primary key definition.

To add a column to the primary key definition, select it in Available Columns and click the Add (>) icon; to remove a column from the primary key definition, select it in Selected Columns and click the Remove (<) icon. To move all columns from available to selected (or the reverse), use the Add All (>>) or Remove All (<<) icon. To move a column up or down in the primary key definition, select it in Selected Columns and use the arrow buttons.

Unique Keys Pane

Specifies one or more unique constraints for the table.

A unique constraint specifies a column, or set of columns, whose data values must be unique: each data value must not be null, and it must not be the same as any other value in the column.

To add a unique constraint, click the Add button; to delete a unique constraint, select it and click the Remove button.

- **Name**: Name of the unique constraint.
- **Enabled**: If this option is selected, the unique constraint is enforced.
- **Rely**: If this option is selected, the constraint in NOVALIDATE mode is taken into account during query rewrite.
- **Deferrable**: If this option is selected, in subsequent transactions, constraint checking can be deferred until the end of the transaction using the SET CONSTRAINT(S) statement.
- **Initially Immediate**: If this option is selected, the constraint is checked at the end of each subsequent SQL statement.
- **Validate**: If the option is selected, the existing data is checked to see if it conforms to the constraint.
- **Index**: Name of the index to which the unique key refers.
- **Tablespace**: Name of the tablespace associated with the index.
• **Available Columns**: Lists the columns that are available to be added to the unique constraint definition.

• **Selected Columns**: Lists the columns that are included in the unique constraint definition.

To add a column to the unique constraint definition, select it in Available Columns and click the Add (>) icon; to remove a column from the unique constraint definition, select it in Selected Columns and click the Remove (<) icon. To move all columns from available to selected (or the reverse), use the Add All (>>) or Remove All (<<) icon. To move a column up or down in the unique constraint definition, select it in Selected Columns and use the arrow buttons.

### Indexes Pane

Lists the indexes defined for the table.

To add an index, click Add Index (+); to delete an index, select it and click Remove Index (-).

• **Name**: Name of the index.

• **Type**: The type of Oracle index. Non-unique means that the index can contain multiple identical values; Unique means that no duplicate values are permitted; Bitmap stores rowids associated with a key value as a bitmap.

• **Tablespace**: Name of the tablespace for the index.

• **Expression**: A column expression is an expression built from columns, constants, SQL functions, and user-defined functions. When you specify a column expression, you create a function-based index.

• **Available Columns** and **Selected Columns**: Columns selected for the index. To select a column, click the column in the Available Columns box, and then click the Add Selected Columns icon to move it to the Selected Columns box.

### Foreign Keys Pane

Specifies one or more foreign keys for the table.

A foreign key specifies a column ("local column"), whose data values match values in the primary key or unique constraint of another table.

• **Name**: Name of the foreign key definition.

• **Enabled**: If this option is checked, the foreign key is enforced.

• **Rely, Deferrable, Initially Immediate, Validate**: See the description of these fields in the Unique Keys pane.

• **Referenced Constraint: Schema**: Name of the schema containing the table with the primary key or unique constraint to which this foreign key refers.

• **Referenced Constraint: Table**: Name of the table with the primary key or unique constraint to which this foreign key refers.

• **Referenced Constraint: Constraint**: Name of the primary key or unique constraint to which this foreign key refers.

• **Referenced Constraint: On Delete**: Action to take automatically when a row in the referenced table is deleted and rows with that value exist in the table containing this foreign key: NO ACTION (shown by a crossing line in diagrams)
performs no action on these rows; CASCADE (shown by an "X") deletes these rows; SET NULL (shown by a small circle) sets null all columns in those rows that can be set to a null value.

• **Associations: Local Column**: Lists the column in the currently selected (local) table that is included in the foreign key definition. For each referenced column in the foreign key definition, select the name of a column in the edited table.

• **Associations: Referenced Column**: For each local column, identifies the column in the other (foreign) table that must have a value matching the value in the local column.

### Table Constraints Pane

Specifies one or more check constraints for the table.

A check constraint specifies a condition that must be met when a row is inserted into the table or when an existing row is modified.

• **Name**: Name of the check constraint definition.

• **Check Condition**: Condition that must be met for a row to fulfil the check constraint. You can use any valid CHECK clause (without the CHECK keyword). For example, to indicate that the value in a numeric column named RATING must be from 1 to 10, you can specify rating >=1 and rating <= 10.

• **Enabled**: If this option is checked, the check constraint is enforced.

### Comments Pane

Enter descriptive comments in this pane. This is optional.

### Storage Pane

Enables you to specify storage options for the table.

When you create or edit a table or an index, you can override the default storage options.

• **Organization**: Specifies that the table is stored and organized with (Index) or without an index (Heap) or as an external table (External).

• **Tablespace**: Name of the tablespace for the table or index.

• **Logging**: ON means that the table creation and any subsequent INSERT operations against the table are logged in the redo log file. OFF means that these operations are not logged in the redo log file.

• **Row Archival**: YES enables in-database archiving, which allows you to archive rows within the table by marking them as invisible.

### External Table Properties Pane

Specifies options for an external table.

An external table is a read-only table whose metadata is stored in the database but whose data in stored outside the database.

**External Table**

• **Access Driver Type**: Specifies the type of external table.
– ORACLE_LOADER: Extracts data from text data files. This is the default access driver, which loads data from external tables to internal tables.

– ORACLE_DATAPUMP: Extracts data from binary dump files. This access driver can perform both loads and unloads.

– ORACLE_BIGDATA: Extracts data from Oracle Big Data Appliance.

– ORACLE_HDFS: Extracts data stored in a Hadoop Distributed File System (HDFS).

– ORACLE_HIVE: Extracts data stored in Apache HIVE.

• Default Directory: Specifies the default directory to use for all input and output files that do not explicitly name a directory object. The location is specified with a directory object, not a directory path.

• Access Params: Assigns values to the parameters of the specific access driver for the external table. Access parameters are optional.

– OPAQUE_FORMAT_SPEC: The opaque_format_spec specifies all access parameters for the ORACLE_LOADER, ORACLE_DATAPUMP, ORACLE_HDFS, and ORACLE_HIVE access drivers. For descriptions of the access parameters, see Oracle Database Utilities. Field names specified in the opaque_format_spec must match columns in the table definition, else Oracle Database ignores them.

– USING CLOB: Enables you to derive the parameters and their values through a subquery. The subquery cannot contain any set operators or an ORDER BY clause. It must return one row containing a single item of data type CLOB.

• Reject Limit: The number of conversion errors that can occur during a query of the external data before an Oracle Database error is returned and the query is aborted.

• Project Column: Determines how the access driver validates the rows of an external table in subsequent queries.

– ALL: Processes all column values, regardless of which columns are selected, and validates only those rows with fully valid column entries. If any column value raises an error, such as a data type conversion error, the row is rejected even if that column was not referenced in the select list of the query.

– REFERENCED: Processes only those columns in the select list of the query. The ALL setting guarantees consistent result sets. The REFERENCED setting can result in different numbers of rows returned, depending on the columns referenced in subsequent queries, but is faster than the ALL setting. If a subsequent query selects all columns of the external table, then the settings behave identically.

• Location: Specifies the data files for the external table. Use the Add (+) icon to add each location specification.

– For ORACLE_LOADER and ORACLE_DATAPUMP, the files are named in the form directory:file. The directory portion is optional. If it is missing, then the default directory is used as the directory for the file. If you are using the ORACLE_LOADER access driver, then you can use wildcards in the file name. An asterisk (*) signifies multiple characters and a question mark (?) signifies a single character.

– For ORACLE_HDFS, LOCATION is a list of Uniform Resource Identifiers (URIs) for a directory or for a file. There is no directory object associated with a URI.
For ORACLE_HIVE, LOCATION is not used. Instead, the Hadoop HCatalog table is read to obtain information about the location of the data source (which could be a file or another database).

**Opaque Format Spec**

Specifies all access parameters for the ORACLE_LOADER, ORACLE_DATAPUMP, ORACLE_HDFS, and ORACLE_HIVE access drivers.

For example:

```sql
RECORDS DELIMITED BY NEWLINE
CHARACTERSET US7ASCII
TERRITORY AMERICAN
BADFILE log_file_dir:'ext_1v3.bad'
LOGFILE log_file_dir:'ext_1v3.log'
FIELDS TERMINATED BY "|" OPTIONALLY ENCLOSED BY '"' LDRTRIM
( PROD_ID,
  CUST_ID,
  TIME_ID DATE(10) "YYYY-MM-DD",
  CHANNEL_ID,
  PROMO_ID,
  QUANTITY_SOLD,
  AMOUNT_SOLD,
  UNIT_COST,
  UNIT_PRICE
)
```

and the full statement:

```sql
CREATE TABLE SH.SALES_TRANSACTIONS_EXT
( PROD_ID NUMBER,
  CUST_ID NUMBER,
  TIME_ID DATE,
  CHANNEL_ID NUMBER,
  PROMO_ID NUMBER,
  QUANTITY_SOLD NUMBER,
  AMOUNT_SOLD NUMBER (10,2),
  UNIT_COST NUMBER (10,2),
  UNIT_PRICE NUMBER (10,2)
)
ORGANIZATION EXTERNAL
(
  TYPE ORACLE_LOADER
  DEFAULT DIRECTORY DATA_FILE_DIR
  ACCESS PARAMETERS
  ( RECORDS DELIMITED BY NEWLINE
    CHARACTERSET US7ASCII
    TERRITORY AMERICAN
    BADFILE log_file_dir:'ext_1v3.bad'
    LOGFILE log_file_dir:'ext_1v3.log'
    FIELDS TERMINATED BY "|" OPTIONALLY ENCLOSED BY '"' LDRTRIM
    ( PROD_ID,
      CUST_ID,
      TIME_ID DATE(10) "YYYY-MM-DD",
```
CHANNEL_ID,
PROMO_ID,
QUANTITY_SOLD,
AMOUNT_SOLD,
UNIT_COST,
UNIT_PRICE
)
)
LOCATION ( "DATA_FILE_DIR":'salesv3.dat')
)
REJECT LIMIT 100
;

CLOB Subquery

Type or copy and paste the query.

Note:

For more information about the external table fields, see Oracle Database Utilities and Oracle Database SQL Language Reference

Materialized View Pane

Specifies options for a materialized view.

Query: Contains the SQL code for the query part of the view definition. Type or copy and paste the query.

General

• **On Pre-built Table:** If Yes, an existing table is registered as a preinitialized materialized view. This option is particularly useful for registering large materialized views in a data warehousing environment. The table must have the same name and be in the same schema as the resulting materialized view, and the table should reflect the materialization of a subquery.

• **Reduced Precision:** Yes authorizes the loss of precision that will result if the precision of the table or materialized view columns do not exactly match the precision returned by the subquery. If No, the precision of the table or materialized view columns must exactly match the precision returned by the subquery, or the create operation will fail.

• **For Update:** Select Yes to allow a subquery, primary key, object, or rowid materialized view to be updated. When used in conjunction with Advanced Replication, these updates will be propagated to the master.

• **Real Time MV:** Select Yes to create a real-time materialized view or a regular view. A real-time materialized view provides fresh data to user queries even when the materialized view is not in sync with its base tables due to data changes. Instead of modifying the materialized view, the optimizer writes a query that combines the existing rows in the materialized view with changes recorded in log files (either materialized view logs or the direct loader logs). This is called on-query computation.
• **Query Rewrite**: If **Enable**, the materialized view is enabled for query rewrite, which transforms a user request written in terms of master tables into a semantically equivalent request that includes one or more materialized views.

• **Build**: Specifies when to populate the materialized view. **Immediate** indicates that the materialized view is to be populated immediately. **Deferred** indicates that the materialized view is to be populated by the next refresh operation. If you specify Deferred, the first (deferred) refresh must always be a complete refresh; until then, the materialized view has a staleness value of unusable, so it cannot be used for query rewrite.

• **Use Index**: If **Yes**, a default index is created and used to speed up incremental (fast) refresh of the materialized view. If **No**, this default index is not created. (For example, you might choose to suppress the index creation now and to create such an index explicitly later.)

• **Index Tablespace**: Specifies the tablespace in which the materialized view is to be created. If a tablespace is not selected, the materialized view is created in the default tablespace of the schema containing the materialized view.

• **Cache**: If **Yes**, the blocks retrieved for this table are placed at the most recently used end of the least recently used (LRU) list in the buffer cache when a full table scan is performed. This setting is useful for small lookup tables. If **No**, the blocks are placed at the least recently used end of the LRU list.

**Refresh Clause**

• **Refresh**: Select **Yes** to enable refresh operations.

• **Refresh Type**: The method of refresh operation to be performed:
  – **Complete Refresh**: Executes the defining query of the materialized view, even if a fast refresh is possible.
  – **Fast Refresh**: Uses the incremental refresh method, which performs the refresh according to the changes that have occurred to the master tables. The changes for conventional DML changes are stored in the materialized view log associated with the master table. The changes for direct-path INSERT operations are stored in the direct loader log.
  – **Force Refresh**: Performs a fast refresh if one is possible; otherwise, performs a complete refresh.

• **Action**: The type of refresh operation to be performed:
  – **On Demand**: Performs a refresh when one of the DBMS_MVIEW refresh procedures are called.
  – **On Commit**: Performs a fast refresh whenever the database commits a transaction that operates on a master table of the materialized view. This may increase the time taken to complete the commit, because the database performs the refresh operation as part of the commit process.
  – **Specify**: Performs refresh operations according to what you specify in the **Start on** and **Next** fields.

• **Start Date**: Starting date and time for the first automatic refresh operation. Must be in the future.

• **Next Date**: Time for the next automatic refresh operation. The interval between the **Start on** and **Next** times establishes the interval for subsequent automatic refresh operations. If you do not specify a value, the refresh operation is performed only once at the time specified for **Start on**.
• **With**: Refresh type, which determines the type of materialized view:
  – Primary Key: Creates a primary key materialized view, which allows materialized view master tables to be reorganized without affecting the eligibility of the materialized view for fast refresh.
  – Row ID: Creates a rowid materialized view, which is useful if the materialized view does not include all primary key columns of the master tables.

• **Default Storage**: If Yes, DEFAULT specifies that Oracle Database will choose automatically which rollback segment to use. If you specify DEFAULT, you cannot specify the rollback_segment. DEFAULT is most useful when modifying, rather than creating, a materialized view.

• **Storage Type**: MASTER specifies the remote rollback segment to be used at the remote master site for the individual materialized view. LOCAL specifies the remote rollback segment to be used for the local refresh group that contains the materialized view. This is the default.

• **Rollback Segment**: Enter the name of the rollback segment.

• **Using Constraint**: If this option is checked, more rewrite alternatives can be used during the refresh operation, resulting in more efficient refresh execution. The behavior of this option is affected by whether you select Enforced or Trusted.
  – Enforced: Causes only enforced constraints to be used during the refresh operation.
  – Trusted: Enables the use of dimension and constraint information that has been declared trustworthy by the database administrator but that has not been validated by the database. If the dimension and constraint information is valid, performance may improve. However, if this information is invalid, then the refresh procedure may corrupt the materialized view even though it returns a success status.

### DDL Pane

You can review and save the SQL statements that are generated when creating or editing the object. If you want to make any changes, go back to the relevant panes and make the changes there.

• For a new table, click **CREATE** to view the generated DDL statements.

• When you edit table properties, click **UPDATE** to view the generated ALTER statements. For a new table, the UPDATE tab will not be available.

When you are finished, click **Apply**.

### Output Pane

Displays the results of the DDL commands. If there are any errors, go to the appropriate pane, fix the errors, and run the commands again. You can save to a text file or clear the output.

### The Index Properties Dialog

The Index Properties dialog box is displayed when you create or edit an index.
To create an index for a selected schema, in SQL, in the Navigator tab, select **Indexes** from the object type drop-down list, click Object submenu **...**, and select **Create Object**.

To edit an index for a selected schema, right-click a table object in the Navigator tab, and select **Edit**.

**Definition pane**

- **Schema**: Database schema that owns the table associated with the index.
- **Table**: Name of the table associated with the index.
- **Schema**: Database in which to create the index.
- **Tablespace**: Tablespace for the index.
- **Name**: Name of the index.
- **Type**: The type of Oracle index.
  - **Non-unique** means that the index can contain multiple identical values.
  - **Unique** means that no duplicate values are permitted.
  - **Bitmap** stores rowids associated with a key value as a bitmap.
- **Expression**: A column name or column expression. A column expression is an expression built from columns, constants, SQL functions, and user-defined functions. When you specify a column expression, you create a function-based index.
- **Available Columns**: Columns available in the table.
- **Selected Columns**: Columns selected for the index. Click Add Selected Columns > to move columns from the Available Columns list.
- **Order**: **ASC** for an ascending index (index values sorted in ascending order); **DESC** for a descending index (index values sorted in descending order).

**DDL pane**

You can review and save the SQL statements that are generated when creating or editing the index. If you want to make any changes, go back to the Definition pane and make the changes there.

- For a new index, click **CREATE** to view the generated DDL statements.
- When you edit index properties, click **UPDATE** to view the generated ALTER statements. For a new index, the UPDATE tab will not be available.

When you are finished, click **Apply**.

**Output pane**

Displays the results of the DDL commands. If there are any errors, go to the Definition pane, fix the errors, and run the commands again. You can save to a text file or clear the output.

**The Sequence Properties Dialog**

The Sequence Properties Dialog is displayed when you create or edit a sequence.

To create a sequence for a selected schema, in SQL, in the Navigator tab, select **Sequences** from the object type drop-down list, click Object submenu **...**, and select **Create Object**.

To edit a sequence for a selected schema, right-click a sequence object in the Navigator tab and select **Edit**.
A sequence is an object from which multiple users may generate unique integers. You can use sequences to automatically generate primary key values.

Properties pane

- **Schema**: Database schema in which to create the sequence.
- **Name**: Name of the sequence.
- **Start with**: Starting value of the sequence.
- **Increment**: Interval between successive numbers in a sequence.
- **Min value**: Lowest possible value for the sequence. The default is 1 for an ascending sequence and -(10^26) for a descending sequence.
- **Max value**: Highest possible value for the sequence. The default is 10^27 for an ascending sequence and -1 for a descending sequence.
- **Cache and Cache size**: Cache causes sequence values to be preallocated in cache, which can improve application performance; Cache size indicates the number of sequence values preallocated in cache. No Cache causes sequence values not to be preallocated in cache.
- **Cycle**: Indicates whether the sequence "wraps around" to reuse numbers after reaching its maximum value (for an ascending sequence) or its minimum value (for a descending sequence). If cycling of values is not enabled, the sequence cannot generate more values after reaching its maximum or minimum value.
- **Order**: Indicates whether sequence numbers are generated in the order in which they are requested. If No Order is specified, sequence numbers are not guaranteed to be in the order in which they were requested.

DDL pane

You can review and save the SQL statements that are generated when creating or editing the sequence. If you want to make any changes, go back to the Properties pane and make the changes there.

- For a new sequence, click CREATE to view the generated DDL statements.
- When you edit a sequence, click UPDATE to view the generated ALTER statements. For a new sequence, the UPDATE tab will not be available.

When you are finished, click Apply.

Output pane

Displays the results of the DDL commands. If there are any errors, go to the Properties pane, fix the errors, and run the commands again. You have save to a text file or clear the output.

The View Properties Dialog

The View Properties Dialog is displayed when you create or edit a view.

You can open the View Properties Dialog from the Navigator tab in SQL or Data Modeler.

To create a view for a selected schema, in SQL, in the Navigator tab, select Views from the object type drop-down list, click Object submenu ***, and select Create Object.
To create a view from an existing template for a selected schema, in the Navigator tab, select the view to create from, right-click and select **Use as Template**.

To edit a view for a selected schema, right-click a view object in the Navigator pane, and select **Edit**.

**Schema**: Database schema in which to create the view.

**Name**: Name of the view.

The different panes in the dialog are described in the following sections:

**SQL Query pane**

Enter or copy and paste the SQL query for the view, using the SELECT and FROM keywords along with the syntax needed to retrieve the desired information. A semicolon is not required after the query.

**Columns pane**

Click **Refresh Columns** to automatically populate the columns in this pane. You can edit the columns by selecting the required row and making changes in the Header Alias and Comments fields.

**Storage pane**

- **Force on Create**: Select **Yes** to create the view regardless of whether the base tables of the view or the referenced object types exist or the owner of the schema containing the view has privileges on them. These conditions must be true before any SELECT, INSERT, UPDATE, or DELETE statements can be issued against the view. If the view definition contains any constraints, CREATE VIEW ... FORCE fails if the base table does not exist or the referenced object type does not exist. CREATE VIEW ... FORCE also fails if the view definition names a constraint that does not exist.

- **Query Restriction**: **Read Only** prevents the view from being used to add, delete, or change data in the underlying table. **Check Option** prohibits any changes to the underlying table that would produce rows that are not included in this view.

Use the Primary Key, Unique Keys, Foreign Keys, and Comments panes to add or edit properties as required.

**DDL pane**

Based on the inputs provided, the DDL statements are generated. You can review and save the SQL statements. If you want to make any changes, go back to the relevant pane and make the changes there.

- For a new view, click **CREATE** to view the generated DDL statements.
- When you edit a view, click **UPDATE** to view the generated ALTER statements. For a new view, the UPDATE tab will not be available.

When you are finished, click **Apply**.

**Output pane**

Displays the results of the DDL commands. If there are any errors, go to the respective pane, fix the errors, and run the commands again. You can save to a text file or clear the output.
The Synonym Properties Dialog

The Synonym Properties Dialog is displayed when you create a synonym.

There are two ways of creating a synonym for a selected schema:

- In SQL, in the Navigator tab, right-click the object for which you want to create the synonym, and select Add Synonym. In this case, the only fields that you can edit in the Properties pane are Public and Synonym Name. The values of the remaining fields are predetermined by the object selected.

- In SQL, in the Navigator tab, select the object type as Synonyms or Public Synonyms from the drop-down list. Click Object submenu, and select Create Object. All the fields in the Properties dialog are available for edit.

The different panes in the dialog are described in the following sections:

Properties pane

- **Public**: If this option is checked, the synonym is accessible to all users. However, each user must have appropriate privileges on the underlying object to use the synonym. If this option is not checked, the synonym is a private synonym, and is accessible only within its schema.

- **Synonym Schema**: Database schema in which to create the synonym.

- **Synonym Name**: Name of the synonym. A private synonym must be unique within its schema; a public synonym must be unique within the database.

- **Object Type**: Specify the type of object to which this synonym refers.

- **Object Schema**: Schema containing the object or name to which this synonym refers.

- **DB Filter**: After selecting the Object Type and Object Schema, the list of objects of the selected type may be very long. To filter the object names, enter the search entry and click Refresh. The Object Name field is auto-filled with appropriate object names in the drop-down list.

- **Object Name**: Select the name of the object to which this synonym refers.

- **DB Link**: Enter a complete or partial database link to create a synonym for a schema object on a remote database where the object is located. If you specify DB Link and omit schema, then the synonym refers to an object in the schema specified by the database link. Oracle recommends that you specify the schema containing the object in the remote database. If you omit DB Link, then Oracle Database assumes the object is located on the local database.

DDL pane

Based on the inputs provided, the DDL statements are generated. You can review and save the SQL statements. If you want to make any changes, go back to the relevant pane and make the changes there.

- For a new view, click CREATE to view the generated DDL statements.

- When you edit a view, click UPDATE to view the generated ALTER statements. For a new view, the UPDATE tab will not be available.

When you are finished, click Apply.
Output pane

Displays the results of the DDL commands. If there are any errors, go to the respective pane, fix the errors, and run the commands again. You can save to a text file or clear the output.

The Materialized View Log Properties Dialog

The Materialized View Log Properties dialog is displayed when you create or edit a materialized view log, which is a table associated with the master table of a materialized view.

To create a materialized view log for a selected schema, in SQL, in the Navigator tab, select Materialized View Logs from the object type drop-down list, click Object submenu, and select Create Object.

To edit, right-click a materialized view log object in the Navigator pane and select Edit.

Schema: Database schema in which to create the materialized view log.

Table: Name of the master table of the materialized view to be associated with this materialized view log.

Properties tab

- **Row ID Logged**: Yes indicates that the rowid of all rows changed should be recorded in the materialized view log. No indicates that the rowid of all rows changed should not be recorded in the materialized view log.

- **PK Logged**: Yes indicates that the primary key of all rows changed should be recorded in the materialized view log; No indicates that the primary key of all rows changed should not be recorded in the materialized view log.

- **New values**: Yes saves both old and new values for update DML operations in the materialized view log; No disables the recording of new values in the materialized view log. If this log is for a table on which you have a single-table materialized aggregate view, and if you want the materialized view to be eligible for fast refresh, you must specify Yes.

- **Object ID Logged**: For a log on an object table only: Yes indicates that the system-generated or user-defined object identifier of every modified row should be recorded in the materialized view log. No indicates that the system-generated or user-defined object identifier of every modified row should not be recorded in the materialized view log.

- **Cache**: For data that will be accessed frequently, CACHE specifies that the blocks retrieved for this log are placed at the most recently used end of the least recently used list in the buffer cache when a full table scan is performed. This attribute is useful for small lookup tables. NOCACHE specifies that the blocks are placed at the least recently used end of the LRU list.

- **Parallel**: If YES, parallel operations will be supported for the materialized view log.

- **Sequence Logged**: Yes indicates that a sequence value providing additional ordering information should be recorded in the materialized view log. No indicates that a sequence value providing additional ordering information should not be recorded in the materialized view log. Sequence numbers (that is, Yes for this option) are necessary to support fast refresh after some update scenarios.

- **Commit SCN**: If this option is enabled, the database is instructed to use commit SCN data rather than timestamps.

- **Available Columns and Selected Columns**: Additional columns, which are non-primary-key columns referenced by subquery materialized views, to be recorded in the
materialized view log. To select one or more filter columns, use the arrow buttons to move columns from Available to Selected.

**Storage tab**

- **Tablespace**: Tablespace in which the materialized view log is to be created
- **Logging**: YES or NO, to establish the logging characteristics for the materialized view log.
- **Buffer Mode**: Select KEEP to put blocks from the segment into the KEEP buffer pool. Select RECYCLE to put blocks from the segment into the RECYCLE pool. Select DEFAULT to indicate the default buffer pool.
- **Percent Free**: Specify a whole number representing the percentage of space in each data block of the database object reserved for future updates to rows of the object. The value of PCTFREE must be a value from 0 to 99.
- **Percent Used**: Specify a whole number representing the minimum percentage of used space that Oracle maintains for each data block of the database object. PCTUSED is specified as a positive integer from 0 to 99 and defaults to 40.
- **Initrans**: Specify the initial number of concurrent transaction entries allocated within each data block allocated to the database object. This value can range from 1 to 255 and defaults to 1.
- **Freelists**: In tablespaces with manual segment-space management, for objects other than tablespaces and rollback segments, specify the number of free lists for each of the free list groups for the table, partition, cluster, or index. The default and minimum value for this parameter is 1, meaning that each free list group contains one free list.
- **Freelist Groups**: In tablespaces with manual segment-space management, specify the number of groups of free lists for the database object you are creating.
- **Initial Extent**: Specify the size of the first extent of the object.
- **Next Extent**: Specify in bytes the size of the next extent to be allocated to the object.
- **Percent Increase**: In locally managed tablespaces, Oracle Database uses the value of PCTINCREASE during segment creation to determine the initial segment size and ignores this parameter during subsequent space allocation.
- **Min Extent**: In locally managed tablespaces, Oracle Database uses the value of MINEXTENTS in conjunction with PCTINCREASE, INITIAL and NEXT to determine the initial segment size.
- **Max Extent**: This storage parameter is valid only for objects in dictionary-managed tablespaces. Specify the total number of extents, including the first, that Oracle can allocate for the object.
- **Unlimited**: Select this option if you want extents to be allocated automatically as needed. Oracle recommends this setting as a way to minimize fragmentation.

**Purge tab**

- **Type**: In IMMEDIATE SYNCHRONOUS, the materialized view log is purged immediately after refresh. This is the default. In IMMEDIATE ASYNCHRONOUS, the materialized view log is purged in a separate Oracle Scheduler job after the refresh operation.
Deferred, Start With, Next, Repeat Interval: Sets up a scheduled purge that is independent of the materialized view refresh and is initiated during CREATE or ALTER MATERIALIZED VIEW LOG statement.

Refresh tab

- **Type: Synchronous Refresh** creates a staging log that can be used for synchronous refresh. Specify the name of the staging log to be created. The staging log will be created in the schema in which the master table resides. **Fast Refresh** creates a materialized view log that can be used for fast refresh. The materialized view log will be created in the same schema in which the master table resides. This is the default.

DDL pane

Based on the inputs provided, the DDL statements are generated. You can review and save the SQL statements. If you want to make any changes, go back to the relevant pane and make the changes there.

- For a new materialized view log, click **CREATE** to view the generated DDL statements.
- When you edit a materialized view log, click **UPDATE** to view the generated ALTER statements. For a new materialized view log, the UPDATE tab will not be available.

When you are finished, click **Apply**.

Output pane

Displays the results of the DDL commands. If there are any errors, go to the respective pane, fix the errors, and run the commands again. You can save to a text file or clear the output.

Implied Foreign Keys

Implied foreign keys are dependencies that exist between tables but are not defined in the database. In a data warehouse environment, it is a common practice not to create foreign keys. However, it becomes necessary to show these dependencies for presentation or reporting purposes.

You can display implied foreign keys for objects in a star schema by defining them or by discovering them in the data dictionary. It is possible to have more than one source for implied foreign keys.

Define Implied Foreign Keys

You can defined implied foreign keys in two ways:

- Using the Implied Foreign Keys dialog
- By dragging the arrow to the referenced object in the diagram

Using the Implied Foreign Keys Dialog

You can define implied foreign keys using the Implied Foreign Keys dialog in Data Modeler.

1. In a Data Modeler diagram, right-click an object (table or view) and select **Implied Foreign Keys**.
In the Implied Foreign Keys dialog, click + to add an entry in the grid.

3. Select the entry in the grid to enable and enter values in the following fields:
   - **Referenced Object**: Object in the diagram that has a dependency to the source object.
   - **Local Column**: Name of the column in the source object.
   - **Referenced Column**: Name of the column in the targeted object.
   - **Discovery Sources**: Automatically prefilled, displays whether the implied foreign keys have been defined or were discovered in the data dictionary.

4. Click OK. The implied foreign key dependency is displayed with a dotted line on the diagram.

By Dragging the Arrow to the Referenced Object in the Diagram

You can also define an implied foreign key in the following way:

1. Select the source object on the diagram.

2. Click and drag the small curved arrow with a blue indicator to the referenced object. The dependency will be displayed with a dotted line on the diagram.
3. Right-click the source object and enter the column names in the Implied Foreign Keys dialog.

Discover Implied Foreign Keys in the Data Dictionary

A star schema can be discovered by searching for several types of definitions in the data dictionary.

In the Navigator tab, right-click an object (table or view) and select **Add Object as Star Schema to Diagram**. The object must be a fact table. The data dictionary is then searched for joins and dependencies related to the object, such as:

- Foreign keys defined for the selected table to other tables. If implied foreign keys are later discovered for the same columns, they will not be displayed on the diagram.
- Joins used in the definitions of bitmap join indexes, materialized views with aggregates, and attribute clustering.
- Dependencies based on dimension definitions and column name matching in fact table.
- Fact and dimension definitions for Analytic views and OLAP cube and dimensions.
The REST Pages

The REST pages enable you to develop and protect ORDS (Oracle REST Data Services) based RESTful web services for your database.

To navigate to the REST pages, do either of the following:

• In the Launchpad page, click REST.

• Click Selector to display the navigation menu. Under Development, select REST.

Topics

• About RESTful Web Services
• RESTful Services Terminology
• About the Overview Page
• About the AutoREST Page
• About the REST Search Toolbar
• Creating RESTful Web Services
• Securing RESTful Web Services

About RESTful Web Services

Representational State Transfer (REST) is a style of software architecture for distributed hypermedia systems such as the World Wide Web. A service is described as RESTful when it conforms to the tenets of REST.

A RESTful service has the following characteristics:

• Data is modeled as a set of resources. Resources are identified by URIs.
• A small, uniform set of operations are used to manipulate resources (for example, GET, POST, PUT, DELETE).
• A resource can have multiple representations (for example, a blog might have an HTML representation and a RSS representation).
• Services are stateless and because it is likely that the client will want to access related resources, these should be identified in the representation returned, typically by providing hypertext links.

RESTful Services Terminology

The following are some major terms related to RESTful services.

• RESTful service: An HTTP web service that conforms to the tenets of the RESTful architectural style.
- **Resource module**: An organizational unit that is used to group related resource templates.

- **Resource template**: An individual RESTful service that is able to service requests for some set of URIs (Universal Resource Identifiers). The set of URIs is defined by the URI pattern of the Resource Template.

- **URI pattern**: A pattern for the resource template. Can be either a route pattern or a URI template, although you are encouraged to use route patterns.
  - **Route pattern**: A pattern that focuses on decomposing the path portion of a URI into its component parts. For example, a pattern of /objects/:object/:id? will match /objects/emp/101 (matches a request for the emp resource with id of 101) and will also match /objects/emp/ (matches a request for the emp resource, because the :id parameter is annotated with the ? modifier, which indicates that the id parameter is optional).
  - **URI template**: A simple grammar that defines the specific patterns of URIs that a given resource template can handle. For example, the pattern employees/{id} will match any URI whose path begins with employees/, such as employees/2560.

- **Resource handler**: Provides the logic required to service a specific HTTP method for a specific resource template. For example, the logic of the GET HTTP method for the preceding resource template might be:

  ```sql
  select empno, ename, dept from emp where empno = :id
  ```

- **HTTP operation**: HTTP (HyperText Transport Protocol) defines the standard methods that are supported in Oracle REST Data Services for building RESTful services: GET (retrieve the resource contents), POST (store a new resource), PUT (update an existing resource), and DELETE (remove a resource).

---

### See Also:

- Structure of RESTful Web Services Video
- Getting Started with RESTful Services

---

### About the Overview Page

The Overview page provides an overview of the activity in the REST pages.

To navigate to the Overview page, do either of the following:

- In the Launchpad page, click **REST**.

- Click **Selector** to display the navigation menu. Under Development, select **REST**.

You can navigate to pages within REST by using the menu in the header: Overview, Modules, AutoREST and Security.

The Overview page consists of three sections.
Objects
Displays the total number of resource modules, schema objects with REST access, roles, privileges, and OAuth clients created. Click the respective card to navigate to the corresponding pages for modules, roles, privileges and OAuth clients.

In the REST pages, the term "object" is used to refer to a module, template, handler, role, privilege, or OAuth Client.

Security
Metadata Catalog: Displays whether access to the ORDS metadata catalog of the schema is protected. The metadata catalog lists all the services in the schema. The user authorization for the metadata catalog is provided at the time of enabling user access.

Modules: Displays the number of modules that are published out of the total modules created.

Module Security: Displays the number of modules that are protected by a privilege out of the total modules created. A partially secured privilege is where the module is protected both by a privilege and a pattern.

Recent Objects
Displays recently updated or created objects. Each entry has a context menu available at the end of the row. Click Actions to access options such as Edit and Delete.

See Also:
Accessing Database Actions

About the AutoREST Page
Use the Automatic Enabling of Schema Objects for REST Access (AutoREST) feature to enable REST access for database objects in the current schema.

Enabling REST access for a schema object such as table, view, function, procedure and package allows it to be accessed through RESTful services. See AutoREST in Oracle REST Data Services Developer’s Guide

To navigate to the AutoREST page, in the REST Overview page, click AutoREST or select AutoREST from the menu in the header.

The AutoREST page consists of three parts:

- **Objects**: Displays the number of REST-enabled schema objects that are protected and require authorization out of the total number of schema objects enabled in the schema.
- **AutoREST**: Displays the number of tables and views, packages, procedures and functions that are REST enabled in the schema. Click a card to see the corresponding objects listed below.
- The bottom part of the page lists schema objects based on the selection made in AutoREST. You can view these objects in card view or grid view. The actions available in the context menu for an object are:
Enabling REST Access for a Database Object

You can enable REST access for a table, view, materialized view, function, procedure or package.

To REST enable a schema object:

1. In the Navigator tab in the SQL page, right-click an object and select REST, then select Enable.

2. Enter the following fields:
   - **Object Alias**: Enter another name for the object. For security purposes, it is recommended to use a different name than the actual object name.
   - **Require Authentication**: Select this option to secure the REST endpoint.
   - **Authorization Role** and **Authorization Privilege**: These fields are automatically pre-filled.

3. Click Enable.

A REST-enabled object is indicated by a **REST Enabled** icon 🔄 in the Navigator tab.

Disabling REST Access for a Database Object

This section describes how to disable REST access for a database object.

1. In the SQL page, in the Navigator tab, right-click the database object that is REST enabled and select Disable.
2. You see a prompt to confirm. Click OK.

Generating cURL Requests for a REST-Enabled Database Object

For a REST-enabled database object, you can generate sample cURL calls for GET, POST, PUT and DELETE requests for the selected object. Use the Copy to Clipboard icon to copy the cURL code.

To view the cURL call for a specific method for the selected object:

1. In the SQL page, in the Navigator tab, right-click the REST-enabled object, select REST and then select cURL command.
2. Select the HTTP method from the left pane, enter values if required and click Next to see the cURL command.

For more information about each HTTP method, see Examples in Oracle REST Data Services.

About the REST Search Toolbar

This section describes the various options available on the search toolbar in the REST pages. The options selected are saved for your session.

Search: Performs a search based on the text entered in the field. This field is not case sensitive. When you start typing in the Search field, the icon appears. If you click the icon, the text that you typed is deleted and you can re-enter text again.

Filter by: Performs a search based on the filters selected. In the Filter by drop down, first select the column to apply the filter on, and then select the values to search for in that column.

Sort By: Sorts the data in ascending or descending based on the columns selected. The columns can be turned on or off. Reset restores the columns to the original state. The sort criteria selected is shown in the upper right corner as Sort By.

Page Size, First Page, Previous page, Next Page: Only displayed in the Card View. Enables you to select page size, and move between pages.

Reload: Refreshes the information on the page.

There are two views available:

Open Card View: This is the default display view. Information is displayed for each object in a card format. Each card displays associated attributes for the object along with the context menu, which list the actions available for the object.

Open Grid View: Displays the information for each object in a single row of a table. The last column in each row contains the context menu icon, which shows the actions available for the object.
Creating RESTful Web Services

Create RESTful web services using the Modules, Templates and Handlers pages.

To create a RESTful web service, you need to:

• Define a resource module
• Define a resource template
• Define one or more resource handlers such as GET, PUT, POST or DELETE. Optionally, define parameters that you need to pass to the resource handler.

The following sections provide information on how to create resource modules, resource templates and resource handlers.

• Managing Resource Modules
• Managing Resource Templates
• Managing Resource Handlers
• Example: Inserting a Record using a POST Handler
• Viewing Resource Handler Details and Managing Parameters

See Also:

• REST Workshop Overview Video
• Creating a RESTful Service in Oracle REST Data Services Quick Start Guide

Managing Resource Modules

You can define, edit and delete resource modules in the Modules page.

To navigate to the Modules page, in the REST Overview page, click Modules in Objects, or from the menu in the header, select Modules.

The module attributes displayed by default in card view are shown in the following figure.
The module status can be published or unpublished. If the status is unpublished, the icon is displayed in a red color.

Click the module name in card view to go to the Templates page for that module. You can also navigate to the Templates page through the context menu.

The actions available in the context menu are:

- Creating a Resource Module
- Importing a Resource Module
- Editing a Resource Module
- Deleting a Resource Module
- Publishing/Unpublishing a Resource Module
- Exporting a Resource Module
- Viewing the Module in OpenAPI View
- Navigating to the Templates page for a module

See Also:

About the REST Search Toolbar

Creating a Resource Module

This section describes how to create a resource module.

1. In the Modules page, click Create Module.
2. Enter the following fields. The fields with an asterisk (*) are mandatory:

   **Module Definition tab**
   - **Module Name**: Enter the name of the module. This field is case sensitive.
   - **Base Path**: Enter the base of the URI for accessing the RESTful service. For example: hr/ means that all URIs starting with hr/ will be serviced by this resource module. Note the change in the Preview URL as you enter the base path.
   - **Is Published**: Enable to make the RESTful service available for use.
   - **Pagination Size**: Enter or select the number of results to return on each page based on a database query. The default value is 25.
   - **Protected By Privilege**: Select a privilege to protect the module from the drop-down list.
   - **Comments**: Enter descriptive comments.
   - **Go to Module after creation**: Select this option to go to the Templates page after the module is created.

   **Origins Allowed tab**
   - Enter origins that are allowed to access the resource templates. Click Add (+) to add each origin. For example:
     http://example1.org
Show code: Select this option to view the PL/SQL code equivalent of the Create Module slider. You can copy and execute this PL/SQL code in the worksheet to perform the same action that occurs when you click Create in the Create Module slider.

3. Click Create.

The new module is displayed on the Modules page. If you cannot see it, locate the module using the Search field.

Importing a Resource Module

This section describes how to import a resource module.

1. In the Modules page, at the top right, click Import Module.
2. Browse and select the JSON file to import. Click Open.

The Import Module slider is displayed with Base Path and Preview URL fields prefilled. Edit the Base Path as needed. Editing the base path field will also update the base path in the JSON format that is displayed in the slider.

3. Click Import.

When the module is imported, a privilege configuration is created and set to protect the module. The name of the privilege is generated by using the base path and adding “_priv” at the end of the base path string and also replacing every slash “/” in the base path to underscore “_”.

4. Your module is ready for editing.

After your module is imported, the templates and handlers are created, but the module is not published. You need to supply SQL and PL/SQL blocks for each handler, and publish your module when it is ready for testing or production.

Editing a Resource Module

This section describes how to edit a resource module.

1. In the Modules page, for the specific module, click and select Edit.
2. Edit the required fields and click Save. For a description of the fields, see Creating a Resource Module.

Deleting a Resource Module

This section describes how to delete a resource module.

1. In the Modules page, for the specific module, click and select Delete.
A prompt appears asking you to confirm.

2. Click **OK** to delete.

Publishing/Unpublishing a Resource Module

This section describes how to make a resource module available or not.

1. In the Modules page, for the specific module, click \[ \] and then select **Publish** or **Unpublish**.

   You see a prompt to confirm.

2. Click **OK**.

Exporting a Resource Module

You can export PL/SQL source code or Open API (Swagger) JSON code for a resource module or RESTful service. If the module has templates, handlers, parameters, roles, and so on, these objects are also displayed and exported.

1. In the Modules page, for a specific module, click \[ \] and select **Export Module**.

2. Select **PL/SQL** to copy or export the PL/SQL source code for the module. Select **OpenAPI** to copy or export the Open API (Swagger) JSON code for the module.

   **Note:**

   A module exported using Open API is always imported with an unpublished status.

3. Click **Copy to Clipboard** to copy the code or click **Download** to download to your local computer.

Viewing the Module in OpenAPI View

This section describes how to open the module in Open API view.

The Open API view option is available for modules that are published and not protected.

1. In the Modules page, for the specific module, click \[ \] and select **OpenAPI View**.

2. The module is displayed as a Swagger UI implementation.

   You can view and execute the handlers, pass parameters to the handlers, and copy or download the responses.

   To learn more about Swagger UI implementation, refer to the official Swagger UI website.

Managing Resource Templates

You can define, edit and delete resource templates for a module in the Templates page.

To navigate to this page, in the Modules page, click \[ \] for a module and select **Templates**, or click the name of the module in card view.

At the top of the page, the module card is available with context menu options such as Edit, Delete, Publish and Export.
The templates attributes displayed by default in card view are shown in the following figure.

Click the template name in card view to go to the Handlers page. You can also navigate to the Handlers page through the context menu.

The actions available in the context menu are:

- Creating a Resource Template
- Editing a Resource Template
- Deleting a Resource Template
- Navigating to the Handlers page for the template

Creating a Resource Template

This section describes how to create a resource template.

1. In the Templates page for a module, click Create Template.
2. Enter the following fields. The fields with an asterisk (*) indicate that they are mandatory:
   - **URI Template**: Enter the URI pattern for the resource template.
     For example, a pattern of /objects/:object/:id? will match /objects/emp/101 (matches a request for the item in the emp resource with id of 101) and will also match /objects/emp/ (matches a request for the emp resource, because the :id parameter is annotated with the ? modifier, which indicates that the id parameter is optional).
     Note that the Preview URL changes as you type the URI template.
   - **Priority**: Enter or choose the priority for how the resource template should be evaluated (1 is low priority and 9 is high priority).
   - **HTTP Entity Tag Type**: Select the type of entity tag to be used by the resource template. An entity tag is an HTTP Header that acts as a version identifier for a resource.
     Options include:
     - **Secure HASH (default)**: The contents of the returned resource representation are hashed using a secure digest function to provide a unique fingerprint for a given resource version.
     - **Query**: Define a query manually that uniquely identifies a resource version. A manually defined query can often generate an entity tag more efficiently than hashing the entire resource representation.
– None: Do not generate an entity tag.

• Comments: Enter descriptive comments if any.

• Go to Handlers after creation: Select this option to go to the Handlers page after the template is created.

Show code: Select this option to view the PL/SQL code equivalent of the Create Template slider. You can copy and execute this PL/SQL code in the worksheet to perform the same action that occurs when you click Create in the Create Template slider.

3. Click Create.

The new template for the module is added to the Templates page.

If "Go to Handlers after creation" is selected, you can use the module name in the breadcrumbs at the top of the page to see the list of templates for the module.

Editing a Resource Template

This section describes how to edit a resource template.

1. In the Templates page, for a specific template, click and then select Edit.

2. Edit the required fields and click Save.

For a description of the fields, see Creating a Resource Template.

Deleting a Resource Template

This section describes how to delete a resource template.

1. In the Templates page, for a specific template, click and then select Delete.

You are prompted to confirm.

2. Click OK to delete.

Managing Resource Handlers

You can create, edit and delete resource handlers for a template in the Handlers page.

To navigate to this page, in the Templates page, click for a template and select Handlers, or click the name of the template in card view.

At the top of the page, the template card is available with context menu options such as Edit and Delete.

The handler attributes displayed by default in card view are shown in the following figure.
The HTTP Method can be one of the following: GET, PUT, POST and DELETE.

Click the handler name to go to the HTTP method page or you can also navigate to the HTTP method page using the context menu.

The actions available in the context menu are:

- Creating a Resource Handler
- Editing a Resource Handler
- Deleting a Resource Handler
- Navigating to the Handler HTTP method page

Creating a Resource Handler

This section describes how to create a resource handler.

1. In the Handlers page, click Create Handler.
2. Specify the properties of the resource handler. The specific options depend on the method type.

**Handler Definition Tab**

- **Method**: Enter the HTTP request method for this handler: GET (retrieves a representation of a resource), POST (creates a new resource or adds a resource to a collection), PUT (updates an existing resource), or DELETE (deletes an existing resource). Only one handler for each HTTP method is permitted.

- **Items Per Page**: Enter or choose the default pagination size, or the number of rows to return for a database query. This option is only available for GET handlers. If this value is not defined, the number of items per page is the one defined in the module.

- **Source Type**: Select the source implementation for the selected HTTP method. The default is collection query.
  - **Collection Query**: Executes a SQL query and transforms the result set into a JSON representation. Available when the HTTP method is GET.
  - **Collection Item**: Executes a SQL query returning one row of data into a JSON representation. Available when the HTTP method is GET.
  - **Media**: Executes a SQL Query conforming to a specific format and turns the result set into a binary representation with an accompanying HTTP Content-Type header identifying the Internet media type of the representation.
  - **PL/SQL**: Executes an anonymous PL/SQL block and transforms any OUT or IN/OUT parameters into a JSON representation.

**Note:**

This option is not available if the template has all four handlers created (GET, POST, PUT, DELETE).
• **Source**: Enter the SQL query or the PL/SQL block for the HTTP method. In the handler details page, you can test the SQL or PL/SQL handler code in the handler editor.

• **Go to Handler after creation**: Select this option to go to the details page after the handler is created.

**MIMEs Allowed tab**

Identifies the type of information included in the HTTP Request Body. For example, if the POST Handler is expecting JSON in the request, add "Application/JSON" as the MIME type.

**Show code**: Select this option to view the PL/SQL code equivalent of the Create Handler slider. You can copy and execute this PL/SQL code in the worksheet to perform the same action that occurs when you click **Create** in the Create Handler slider.

3. Click **Create**.

The handler is displayed on the Handlers page for the specific template.

You can test the REST service endpoint using a REST client or a command-line tool such as cURL.

---

**About cURL and Testing RESTful Services in Oracle REST Data Services Developer’s Guide**

---

**Editing a Resource Handler**

This section describes how to edit a resource handler.

1. In the Handlers page, for a specific handler, click ![Go to Details](image) and then click **Details**.

2. Edit the required fields and click **Save**.

   For a description of the fields, see **Creating a Resource Handler**.

** Deleting a Resource Handler**

This section describes how to delete a resource handler.

1. In the Handlers page, for the specific handler, click ![Go to Details](image) and then click **Delete**.

   You are prompted to confirm.

2. Click **OK** to delete.

**Example: Inserting a Record using a POST Handler**

The following example illustrates how to insert a record in the DEPT table.

**Prerequisites**
• Using the worksheet in the SQL page, create a DEPT table.

```sql
CREATE TABLE DEPT(
    DEPTNO number(2,0),
    DNAME varchar2(14),
    LOC varchar2(13),
    CONSTRAINT PK_DEPT PRIMARY KEY (DEPTNO)
)
```

• Create a module named example. See Creating a Resource Module
• Create a template named emp/ for the module. See Creating a Resource Template

To insert a record:

1. In the Handlers page, click Create Handler.
2. Enter the following details:
   • In the Method field, select POST.
   • In the Source Type field, select PL/SQL.
   • In the Source field, enter the following PL/SQL block:
     ```sql
     DECLARE
         id number;
     BEGIN
         INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (:DEPTNO, :DEPTNAME, :DEPTLOC) RETURNING DEPTNO INTO id;
         :status_code := 201;
         :forward_location := id;
     END;
     ```
     where
     - RETURNING DEPTNO INTO id returns the value assigned to DEPTNO into the variable id.
     - :status_code is an implicit parameter that is assigned 201 to indicate that the resource is created.
     - :forward_location is an implicit parameter that is assigned the id that contains the value of DEPTNO. This parameter specifies the location to forward a GET request to produce the response for the POST request.
   • Ensure Go to Handler page after creation is selected.
   • In the MIMEs Allowed Tab, select application/json from the drop-down list and click Add +.
   • Click Create.

The Handler POST page appears.
You can test the source code by clicking **Execute** in the Source editor.

3. Create a GET resource handler to produce a response for the POST request. 
   In the Example module, create a template named **emp/:id**.
   Create a GET resource handler where **Source Type** is **Collection Item** and **Source** contains the following query:
   
   ```sql
   SELECT * FROM DEPT WHERE DEPTNO = :id
   ```

4. Test the RESTful service endpoint using the following command in cURL. You can copy the URL (http://xyz.us.comp.com:1234/ords/pdbdba/example/emp) using the Copy to Clipboard icon in the POST handler card at the top.
   
   ```bash
   curl --location --request POST --header "Content-Type: application/json" 
   --data '{"DEPTNO": 54, "DEPTNAME": "HR", "DEPTLOC": "America" }' 'http://xyz.us.comp.com:1234/ords/pdbdba/example/emp/
   ```
The output is:

{"deptno":54,"dname":"HR","loc":"America","links":
[{"rel":"collection","href":"https://xyz.us.comp.com:1234/ords/pdbdba/example/emp/"}]

5. In the SQL page, check the DEPT table to see if the new record has been inserted by using the following statement:

SELECT * FROM DEPT;

Viewing Resource Handler Details and Managing Parameters

You can view resource handler details and manage parameters in the HTTP method page.

To navigate to this page, for a specific handler, click \(\text{Details}\) and then select Details or click the name of the handler in card view.

At the top of the page, the handler card is available with context menu options such as Edit and Delete.

The Handler HTTP method page has two major sections:

- A code editor for executing the SQL or PL/SQL source code that was defined in the resource handler. You can execute the code, save, undo, redo, download and clear output. For more information about the editor, see Executing SQL Statements in the Code Editor.

  Implicit parameters used in REST service handlers are displayed in a scrolling list towards the right side of the editor. See Implicit Parameters

- A section for managing parameters required for running the HTTP method.

By default, the parameters are displayed in grid view. The attributes displayed on a parameter in card view are shown in the following figure.

The actions available in the context menu are:

- Creating a Parameter
- Editing a Parameter
- Deleting a Parameter
Creating a Parameter

This section describes how to create a parameter for a resource handler.

1. In the handler HTTP method page, click **Create Parameter**.
2. Enter the following fields. The fields with an asterisk (*) are mandatory:
   - **Parameter Name**: Enter the name of the parameter, as it is named in the URI Template or HTTP Header. The name defines how the parameter is identified in the incoming request or how the parameter is named in the Response.
   - **Bind Variable Name**: Enter the name of the parameter, as it will be referred to in SQL. If NULL is specified, then the parameter is unbound.
   - **Source Type**: Select the type that is identified if the parameter originates in the URI Template or HTTP Header.
   - **Parameter Type**: Select the native type of the parameter.
   - **Access Method**: Select the parameter access method. Indicates if the parameter is an input value, output value, or both.

**Show code**: Select this option to view the PL/SQL code equivalent of the Create Parameter slider. You can copy and execute this PL/SQL code in the worksheet to perform the same action that occurs when you click **Create** in the Create Parameter slider.

3. Click **Create**.
   The parameter is added in the Parameters section.

Editing a Parameter

This section describes how to edit a parameter.

1. In the Parameters section, for the specific parameter, click and then select **Edit**.
2. Edit the required fields and click **Save**.
   For a description of the fields, see **Creating a Parameter**.

Deleting a Parameter

This section describes how to delete a parameter.

1. In the Parameters section, for the specific parameter, click and then select **Delete**.
   You are prompted to confirm.
2. Click **Yes** to delete.
Implicit Parameters

This section describes the implicit parameters in the Resource Handler Details page. These parameters are automatically added to the resource handlers.

Table 4-1  Implicit Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Access Mode</th>
<th>HTTP Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:body</td>
<td>BLOB</td>
<td>IN</td>
<td>N/A</td>
<td>Specifies the body of the request as a temporary BLOB.</td>
</tr>
<tr>
<td>:body_text</td>
<td>CLOB</td>
<td>IN</td>
<td>N/A</td>
<td>Specifies the body of the request as a temporary CLOB.</td>
</tr>
<tr>
<td>:content_type</td>
<td>VARCHAR</td>
<td>IN</td>
<td>Content-</td>
<td>Specifies the MIME type of the request body, as indicated by the Content-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Type</td>
<td>Type request header.</td>
</tr>
<tr>
<td>:current_user</td>
<td>VARCHAR</td>
<td>IN</td>
<td>N/A</td>
<td>Specifies the authenticated user for the request. If no user is authentic</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>ated, then the value is set to null.</td>
</tr>
</tbody>
</table>
### Table 4-1  (Cont.) Implicit Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Access Mode</th>
<th>HTTP Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:forward_location</td>
<td>VARCHAR</td>
<td>OUT</td>
<td>X-ORDS-FORWARD-LOCATION</td>
<td>Specifies the location where Oracle REST Data Services must forward a GET request to produce the response for this request.</td>
</tr>
<tr>
<td>:fetch_offset</td>
<td>NUMBER</td>
<td>IN</td>
<td>N/A</td>
<td>Specifies the zero-based offset of the first row to be displayed on a page.</td>
</tr>
<tr>
<td>:fetch_size</td>
<td>NUMBER</td>
<td>IN</td>
<td>N/A</td>
<td>Specifies the maximum number of rows to be retrieved on a page.</td>
</tr>
</tbody>
</table>
### Table 4-1  (Cont.) Implicit Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Access Mode</th>
<th>HTTP Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:page_offset</td>
<td>NUMBER</td>
<td>IN</td>
<td>N/A</td>
<td>Specifies the zero based page offset in a paginated request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Note: The :page_offset parameter is deprecated. Use :row_offset parameter instead.</td>
</tr>
<tr>
<td>:page_size</td>
<td>NUMBER</td>
<td>IN</td>
<td>N/A</td>
<td>Specifies the maximum number of rows to be retrieved on a page.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>The :page_size parameter is deprecated. Use :fetch_size parameter instead.</td>
</tr>
</tbody>
</table>
Table 4-1  (Cont.) Implicit Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Access Mode</th>
<th>HTTP Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>:row_of</td>
<td>NUMBER</td>
<td>IN</td>
<td>N/A</td>
<td>Specifies the one-based index of the first row to be displayed in a paginated request.</td>
</tr>
<tr>
<td>fset</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>:row_count</td>
<td>NUMBER</td>
<td>IN</td>
<td>N/A</td>
<td>Specifies the one-based index of the last row to be displayed in a paginated request.</td>
</tr>
<tr>
<td>:status_code</td>
<td>NUMBER</td>
<td>OUT</td>
<td>X-ORDS-STATUS-CODE</td>
<td>Specifies the HTTP status code for the request.</td>
</tr>
</tbody>
</table>

For more information about the implicit parameters, see Oracle REST Data Services Developer’s Guide.

Securing RESTful Web Services

Define roles, privileges and OAuth Clients to ensure authentication and authorization are required for accessing RESTful web services.

To protect a RESTful web service, you need to:

- Create a role
- Create a privilege selecting the role and modules or resources to protect

To enable access to a protected RESTful service using the OAUTH2 Workflow, create an OAuth client using the role and privilege created for protecting the RESTful service.

The following sections provide information on how to create roles, privileges and OAuth clients:

- Managing Roles
- Managing Privileges
Managing Roles

You can create, edit and delete roles for RESTful services in the Roles page.

To navigate to the Roles page, from the REST Overview page, click Roles in Objects, or from the menu in the header, select Security and then select Roles.

The actions available in the context menu are:

- Creating a Role
- Editing a Role
- Deleting a Role
- Viewing Assigned Privileges

Creating a Role

Create a role with a specific name. After the role is created, you can associate it with a privilege.

1. In the Roles page, click Create Role.
2. In the Role Name field, enter the name of the role to create.
   
   Show code: Select this option to view the PL/SQL code equivalent of the Create Role slider. You can copy and execute this PL/SQL code in the worksheet to perform the same action that occurs when you click Create in the Create Role slider.
3. Click Create.

The new assigned role is displayed on the Roles page.

Editing a Role

This section describes how to edit a role.

1. In the Roles page, for the specific role, click Actions and select Edit.
2. Enter the changes required and click Save.
For a description of the fields, see Creating a Role.

Deleting a Role

This section describes how to delete a role.

1. In the Roles page, for the specific role, click Actions and select Delete. You are prompted to confirm.
2. Click Yes to delete.

Viewing Assigned Privileges

This section describes how to view the privileges associated with a role.

In the Roles page, for the specific role, click Actions and select Details. The privileges assigned to the role are displayed.

Managing Privileges

You can create, edit and delete privileges for RESTful services in the Privileges page.

A privilege defines the set of roles, at least one of which an authenticated user must possess to access a RESTful service protected by a privilege.

To navigate to the Privileges page, from the REST Overview page, click Privileges in Objects, or from the menu in the header, select Security and then select Privileges.

The privilege attributes displayed by default in card view are shown in the following figure.

The actions available in the context menu are:

• Creating a Privilege
• Editing a Privilege
• Deleting a Privilege

Creating a Privilege

This section describes how to create a privilege.

1. In the Privileges page, click Create Privilege.
2. Enter the following fields. The fields with an asterisk (*) are mandatory:
   - **Label**: Enter the name of the privilege in a way that it is easy to understand.
   - **Name**: Enter a unique name for the privilege.
   - **Description**: Enter a brief description of the purpose of the privilege.
   - **Comments**: Enter comments.
   - **Roles**: Enter one or more roles assigned to the privilege.
   - **Protected Modules**: Select the modules to protect.
   - **Protected Resources**: Instead of Protected Modules, use the Protect Resources tab to apply security based on a URI pattern.

   **Show code**: Select this option to view the PL/SQL code equivalent of the Create Privilege slider. You can copy and execute this PL/SQL code in the worksheet to perform the same action that occurs when you click **Create** in the Create Privilege slider.

3. Click **Create**.
   The new privilege is displayed on the Privileges page.

### Editing a Privilege

This section describes how to edit a privilege.

1. In the Privileges page, for the specific privilege, click **Actions** and select **Edit**.
2. Make the required changes and click **Save**.
   For a description of the fields, see **Creating a Privilege**.

### Deleting a Privilege

This section describes how to delete a privilege.

1. In the Privileges page, for the specific privilege, click **Actions** and select **Delete**.
2. You are prompted to confirm. Click **Yes**.

### Managing OAuth Clients

Using OAuth 2.0-based authentication, you can ensure that your RESTful web services are accessed only by specific users or clients.

OAuth 2.0 is a standard Internet protocol that defines flows to provide conditional and limited access to a RESTful API. For more information, see OAuth-Based Authentication.

You can create, edit and delete OAuth Clients in the OAuth Clients page.

To navigate to the OAuth Clients page, from the REST Overview page, click **Clients** in Objects, or from the menu in the header, select **Security** and then select **OAuth Client**.

The OAuth Client attributes displayed by default in card view are shown in the following figure.
The actions available in the context menu are:

- **Creating an OAuth Client**
- **Editing an OAuth Client**
- **Deleting an OAuth Client**
- **Get Bearer Token**: Provides the access token to call the RESTful service.

### See Also:

Tutorial for Protecting and Accessing Resources

---

**Creating an OAuth Client**

Creates the OAuth Client and grants the required roles and privileges.

1. In the OAuth Clients page, select **Create OAuth Client**.
2. Enter the following fields. The fields with an asterisk (*) are mandatory:
   - **Name**: Name of the client.
   - **Description**: Description of the purpose of the client.
   - **Support URI**: Enter the URI where end users can contact the client for support. Example: http://www.myclientdomain.com/support/
   - **Support Email**: Enter the email where end users can contact the client for support.
   - **Roles**: Select roles to be granted.
   - **Allowed Origins**: Add the list of URL prefixes.
   - **Privileges**: Select privileges that the client wants to access.

   **Show code**: Select this option to view the PL/SQL code equivalent of the Create OAuth Client slider. You can copy and execute this PL/SQL code in the worksheet to perform the same action that occurs when you click **Create** in the Create OAuth Client slider.

3. Click **Create**.

The OAuth Client registered is displayed on the OAuth Clients page.
The Client ID and Client Secret values represent the secret credentials for the
OAuth client. Click **Show/Hide** to see the values.

Test the secured REST service endpoint using a REST client or the cURL
command line tool.

### See Also:

Tutorial for Protecting and Accessing Resources in *Oracle REST Data
Services Developer's Guide*

---

**Editing an OAuth Client**

This section describes how to edit an OAuth Client.

1. In the OAuth Clients page, for the specific client, click **Actions** and select **Edit**.
2. Edit the required fields and click **Save**.

For a description of the fields, see **Creating an OAuth Client**.

**Deleting an OAuth Client**

This section describes how to delete an OAuth Client.

1. In the OAuth Clients page, for the specific client, click **Actions** and select **Delete**.
2. You are prompted to confirm. Click **Yes**.

**Example: Creating an OAuth Client**

This section describes how to create an OAuth Client using an example.

Create an OAuth Client for the created module “example” in **Example: Inserting a
Record using a POST Handler**. The endpoint for the RESTful service is http://
xyz.us.comp.com:1234/ords/pdbdba/example/emp/.

**Prerequisites**

Create a role named HR Admin. See **Creating a Role**

Create a privilege named Example.HR. See **Creating a Privilege**

1. In the OAuth Clients page, click **Create OAuth Client**.
2. Enter the following fields:
   * Enter Name, Description, Support URI and Support Email for your OAuth
     Client.
   * In the Roles tab, add the created role (HR Admin).
   * In the Privileges tab, add the created privilege (Example.HR).
   * Click **Create**.

The new OAuth Client card appears on the OAuth Clients page.
3. Using cURL, test the service endpoint without OAuth credentials.

    curl --location --request POST --header "Content-Type: application/json"
    --data '{"DEPTNO": 55, "DEPTNAME": "Sales", "DEPTLOC": "Australia" }
    'http://xyz.us.comp.com:1234/ords/pdbdba/example/emp/

You see an error.

4. Test the endpoint with OAuth credentials:
   
   a. To get an access token, select **Get Bearer Token** from the context menu in the OAuth Client card.

      ![OAuth Client card](image)

      The OAuth Token dialog appears. Use **Copy to Clipboard** to copy the token.

      ![OAuth Token dialog](image)

   b. In cURL, use the bearer access token in the following statement to request the resource:

    curl -H "Content-Type: application/json" -H "Authorization: Bearer
    AMccw1nt99gm9kxDs_w1DA" --location --request POST --data
    '{"DEPTNO": 55, "DEPTNAME": "Sales", "DEPTLOC": "Australia"} 
    'http://xyz.us.comp.com:1234/ords/pdbdba/example/emp/
5. In the SQL page, you can verify the new record that has been inserted in the DEPT table by using the following statement:

```sql
SELECT * FROM DEPT;
```
The JSON Page

Note:
Available for Oracle Database release 19c and later releases and only if you signed in as a database user with the SODA_APP role.

Use the JSON page in Database Actions to view and manage JSON collections, search for collection items using Query-by-Example (QBE), create JSON search indexes and display data guide diagrams for collections.

To navigate to the JSON page, do either of the following:
• In the Launchpad page, click JSON.
• Click Selector to display the navigation menu. Under Development, select JSON.

Topics
• About the JSON User Interface
• Managing JSON Collections
• About Querying Documents in a Collection
• Creating Indexes for JSON Collections
• Creating Relational Views of JSON Documents
• Viewing the JSON Data Guide Diagram for a Collection

See Also:
SODA Overview in Oracle Database Introduction to Simple Oracle Document Access (SODA)

About the JSON User Interface

The JSON user interface consists of the left pane for listing and searching saved collections and the right pane for viewing and managing documents in a collection.

Listing and Searching JSON Collections

The following figure shows the main items in the left pane of the JSON page.
Select the appropriate option in the drop-down list to display saved JSON collections or recently accessed JSON collections.

The search functionality is not case-sensitive, retrieves all matching entries, and does not require the use of wildcard characters.

Click the Create Collection icon to create a new collection. See Creating a Collection

Right-click a collection name to open the context menu. The available options are:

- **Indexes**: Enables you to view existing JSON indexes and create search, functional or spatial indexes.
- **Drop**: Removes the collection from the database completely.

**Viewing the Contents of a Collection**

When a specific collection is selected in the left pane, the JSON documents that belong to the collection are displayed in the lower part of the right pane. The top part of the right pane contains the JSON editor where you can run queries for filtering or sorting the documents.
The two icons in the right corner are:

- **Tour**: Starts the JSON tour, which provides information about the features available.
- **Go to SQL**: Navigates to the SQL page.

The icons in the toolbar are:

- **New JSON Document**: Adds a new document to the collection. See About Adding or Editing a JSON Document
- **Import Document(s)**: Imports one or more existing JSON files from your local computer into the collection.
- **Delete All Documents in the list**: Deletes all JSON documents in the collection that match the current QBE search string. If the current string is {}, then all the documents in the collection are deleted.
- **Collection Details**: Enables you to view collection properties, JSON data guide (if created) and related statistics if they are gathered, size of search index, and page for managing JSON indexes.
- **Diagram**: Displays the the JSON data guide as a diagram in a hierarchical format. See Viewing the JSON Data Guide Diagram for a Collection
- **New Collection View**: Creates relational views of documents in a collection.
• **Run Query**: Filters documents using the QBE condition entered in the JSON editor. See About Querying Documents in a Collection

• **View Collection SQL**: Displays the JSON collection in SQL format. You can execute the SQL code, download it or copy to clipboard.

Each JSON document has the following icons:

• **Edit Document**: Edits the JSON document. See About Adding or Editing a JSON Document

• **Clone Document**: Creates a clone of the document.

• **Copy Document**: Copies the document to the clipboard.

• **Export Document**: Downloads the document as a .JSON file.

• **Delete Document**: Deletes the document.

### Managing JSON Collections

You can add, view or drop collections, or browse, add, edit and delete JSON documents in a collection.

• **Create a Collection**: See Creating a Collection

• **Add or Edit Documents in a Collection**: See About Adding or Editing a JSON Document

• **View Documents in a Collection**: Select the collection name in the left pane. The documents in the collection are displayed in the right pane.

• **View Collection Details**: Select the specific collection in the left pane, and then click Collection Details in the right pane toolbar to view collection properties, JSON data guide (if created) and related statistics if gathered, the size of search index, and the page for managing JSON indexes.

• **Drop a Collection**: Right-click a collection name in the left pane to open the context menu. Select Drop to remove the collection from the database completely.

### Creating a Collection

This section describes how to create a collection.

1. When you navigate to the JSON page:
   • If there are no collections created, you see the JSON home page. Click Create Collection.
   • If there are existing collections available, then in the left pane, click Create Collection.

2. Enter the collection name, which is case sensitive.

3. If the MongoDB Compatible option is selected, then the document IDs will be Embedded IDs.
   • Document IDs are part of the JSON document (field name is "_id"). If an _id field is missing, it is generated and added to the JSON document.
This collection is MongoDB compatible. MongoDB drivers connecting to the MongoDB API for the Oracle Database can read and write to this collection.

To use these collections from SODA, you need a SODA driver that supports the EMBEDDED_OID key assignment method.

This method is documented as EMBEDDED_OID.

The name of the content column (the column where JSON documents are stored) in the underlying table is DATA.

Out of line IDs:

- SODA always generates the ID for a new document.
- The ID is not embedded in the JSON document but stored separately.
- MongoDB drivers connecting to the MongoDB API for the Oracle Database can only read from this collection.
- This method is documented as UUID.
- The name of the content column in the underlying table is JSON_DOCUMENT.

4. Click **Create** to create the collection.

   Click **Cancel** or press the **Esc** key to cancel the collection.

---

### About Adding or Editing a JSON Document

You can add and edit JSON documents using the JSON editor.

To add a document, click **New JSON Document** in the right pane toolbar for a specific collection. In the JSON editor, you can copy and paste the JSON document or use the Import icon to import the JSON document.

To edit a document, click **Edit Document** in the document card.

The following figure shows an open document in the JSON editor.
About Database Differences in JSON Documents

You can find differences in a JSON document that has been edited from two different connections.

At the time of saving an edited document, you are notified that there are differences between your edited document and the instance of the document in the database. The differences are also highlighted enabling a quick review.

Click Overwrite if you prefer to overwrite the changes to the document instance in the database.
You can search for one or more documents in a collection by using a filter specification or Query-by-Example (QBE).

A QBE is a pattern expressed in JSON. You use it to select, from a collection, the JSON documents whose content matches it, meaning that the condition expressed by the pattern evaluates to true for the content of only those documents. For more information about QBEs and how to use them, see Overview of SODA QBEs in Oracle Database Introduction to Simple Oracle Document Access (SODA).

For a specific collection, enter the QBE string in the JSON editor. For example, to select documents where the name is Mary, enter \{"name":"Mary"\} and then click Run Query, as shown in Figure 5-2. The results of the query are displayed in the lower right pane.

The editor offers a comprehensive list of commands available through the Command Palette. To open the Command Palette, press Ctrl+Shift+P. For a list of keyboard shortcut keys, see Keyboard Shortcuts.

An error in the query is signified by a red dot in the left gutter and a squiggle line beneath the specific text. When you hover over it, you see a pop-up displaying possible fixes for resolving the error.

You can set editor preferences using the Preferences option in the top-right user drop-down list on the header. Some of the available options are Theme (Light, Dark and High contrast dark), Font size and family, Tab size, Word wrap, Ruler, Line numbers and so on.
The icons above the editor are:

- **Format JSON data** enables indentation and line feeds for the QBE string.
- **Add Clause**: Adds a formatted template of the $orderby or $patch clause to the QBE string.
- **Clear**: Clears the current QBE string.
- **History**: Retrieves previous QBE search strings.

**Figure 5-2  QBE String in JSON Editor**

The QBE expression must be a valid JSON object and can contain $query and $orderby or $patch clauses. The QBE expression is treated as a $query clause if there are no clauses defined. Starting with the $query clause, add the $orderby or $patch clause later using the **Add Clause** list. The content is transformed and a template for the $orderby and $patch clause is provided. You need to set correct values in the templates. See Using the In-Context Autocomplete Feature in the JSON Editor

For example, this is a simple filtering query:
After the $orderby clause is added using Add Clause, you see the following entry:

```
1 {  
2   "site_admin": true
3 }
```

After the $patch clause is added, the query changes to:

```
1 {  
2   "$query": {  
3     "site_admin": true
4   },  
5   "$orderby": {  
6     "$fields": [
7       {  
8         "path": ":",  
9         "datatype": "varchar2",  
10        "order": "asc"
11       },  
12       "$scalarRequired": false,
13       "$lax": false
14     }
15  },  
16  "$patch": [
17     {  
18       "op": "test"
19     },  
20     {  
21       "path": "/",
22       "value": ""  
23     }
24   ],  
25   "$orderby": {  
26     "$fields": [
27       {  
28         "path": ":",  
29         "datatype": "varchar2",  
30         "order": "asc"
31       },  
32       "$scalarRequired": false,
33       "$lax": false
34     }
35  }
```
Using the In-Context Autocomplete Feature in the JSON Editor

If you press **Ctrl+Space**, the editor provides you with a context-aware list of options from which you can select and autocomplete at the insertion point.

The following types of information appear in the list:

- **Filter comparison clauses**
  
  A template is available for each clause and it is inserted at the cursor position. For example, when you select the `$between` clause from the autocomplete list, you see the following entry in the editor:

  ```
  "age": 49, 70
  ```

  where `age` is the property name and 49 and 70 are values. These are parameters that you can edit and they appear highlighted. You can type or use the autocomplete help to edit the property name. Press the **Tab** key to move to the next parameter.

- **Property names from JSON documents in the collection**

  There are two sources for property names from JSON documents:

  - If a search index is created with support for data guide, then the JSON data guide from the database dictionary is scanned for property names.
  - Property names that are collected from viewed or edited documents are presented. It is possible that the property names are a subset from the whole namespace.

  Based on the type of property, the related template is inserted and the cursor is positioned at the expected place for insertion. The following is a list of property types, their templates and the corresponding cursor position for each:
Creating Indexes for JSON Collections

You can create indexes for JSON collections in the JSON page.

See Also:

Indexes for JSON Data in Oracle Database JSON Developer's Guide

Open the Indexes Pane

In the JSON left pane, right-click the collection, and select Indexes.

The Indexes pane lists the existing indexes for the collection. Select the index row to display more information.
The icons at the top are **Add JSON Index**, **Edit JSON Index**, and **Delete JSON Index**.

The properties of the selected index appear in JSON format below the listed indexes. Select **JSON** from the TABLE - JSON option to view all indexes in JSON presentation.

**Create an Index**

1. Click the **+ New JSON Index** icon. The New Index pane appears.
2. Enter the following fields to create an index:

   - **Name**: Enter a name for the index.
   - **Type**: Select the index type from the drop-down list. The different options are Functional, Spatial and Search. Based on the index type selected, the corresponding options appear.
• For a functional type index, the fields to enter are:
  – **Unique**: Select this option to make all indexed values unique.
  – **Index Nulls**: Select this option to use the index in Order By queries.
  – **Path Required**: Select this option if the path must select a scalar value, even a JSON null value.
  – **Properties**: Type the property that you want to index on, or Type * to display all available document properties in the collection. To select a property, select the checkbox in the respective row.

<table>
<thead>
<tr>
<th>Note:</th>
</tr>
</thead>
<tbody>
<tr>
<td>You cannot index properties in arrays.</td>
</tr>
</tbody>
</table>

  – **Composite Index**: Select this option if you want to use more than one property.
  – **Advanced**: Select this option to change the storage properties of the indexed property. For each property, you can change the type (varchar2, number, date or timestamp), maximum length for indexing (for character properties), and sort order.

• For search index, the options are:
  – **Dataguide off-on**: Select on to create JSON data guide for collection.
  – **Text Search off-on**: Select on to index all properties in documents to support full-text search based on string equality (every property is treated as string)
  – **Range Search off-on**: Select on to support range search when string-range search or temporal search (equality or range) is required.

• Spatial index is used to index GeoJSON geographic data. The selected property should be of GeoJSON type. See Using GeoJSON Geographic Data

  For spatial index, the options are:
  – **Path Required**: Select this option if the path must select a value, even if it is a JSON null value.
  – **Lax**: Select this option if the targeted field does not need to be present or does not have a GeoJSON geometry object as its value.

<table>
<thead>
<tr>
<th>Note:</th>
</tr>
</thead>
<tbody>
<tr>
<td>You cannot enable Path Required and Lax at the same time.</td>
</tr>
</tbody>
</table>

3. Click **Create**. A notification is displayed indicating that the index is created and the Indexes pane is populated.

### Creating Relational Views of JSON Documents

You can create relational views of JSON documents.

1. In the JSON page, click **New Collection View** in the right pane.
2. Enter the following fields:
   - **View Name**: Enter a name for the view.
   - **Select columns to add**: Click in this field to view the list of available properties and to select the required properties, adding them one at a time. Alternatively, select Add All to add all objects and arrays with all their properties. Select one or several properties and move them up or down to position them in the resulting view. Objects and arrays are moved or deleted together with all their properties.

   You can edit the Column Name attribute for each column, and for string types, you can also edit the Length attribute. Precision and scale can be set for numeric properties, for example, (10,2).

3. Click **Test Query** to review the DDL statements before creating the view.

   Click **Create** to create the view, or click **Definition** to return to the previous screen and make changes to the corresponding attributes.

   The created view is now available in the SQL page.

**Viewing the JSON Data Guide Diagram for a Collection**

The diagram view displays the JSON data guide for a collection as a hierarchical structure using a format similar to entity-relationship diagrams.

The JSON data guide represents the JSON schema for documents that have a column with JSON content.

In the diagram, arrays are presented as one-to-many relationships, contained objects as one-to-one relationships, and “oneOf” constructs as a box that surrounds possible choices.

Click **Diagram** in the right pane toolbar to display the JSON data guide diagram for a specific collection.
The icons in the toolbar are Print Diagram, Save to SVG format. Zoom In, Zoom Out, Fit Screen and Actual Size.

See Also:

JSON Data Guide in Oracle Database JSON Developer's Guide
The Charts and Dashboards Page

Charts enable you to create charts from the database. The chart is constructed using the input SQL command. Dashboards enable you to group charts together to create reports.

To navigate to the Charts and Dashboards page, do either of the following:

• In the Launchpad page, click Charts.
• Click Selector to display the navigation menu. Under Development, select Charts.

Topics

• Overview
• Creating or Editing a Chart
• Example: Creating and Editing a Chart
• Creating or Editing a Dashboard

Overview

This section describes the layout of the main Charts and Dashboards page.

To navigate to the Charts and Dashboards page, do either of the following:

The upper right corner has two icons:

• Tour: Provides a guided tour of charts and dashboards highlighting salient features and providing information that is useful if you are new to the interface.
• Create: Creates a new chart or dashboard. See Creating or Editing a Chart and Creating or Editing a Dashboard

This page consists of two sections:

• Charts and Dashboards: Displays the total number of charts and dashboards created. Click the respective card to view the charts or dashboards created in the section below.
• Displays the charts or dashboards created in the default card format.

The actions available for a chart or dashboard are:

• View Chart/Dashboard: To view the chart or dashboard on a different page.
• Edit: To make changes to the chart or dashboard.
• Unpublish: To make the chart or dashboard unavailable for use.
• Go to module definition: To show the module definition in the REST Modules page.
• Delete: To delete the chart or dashboard.
Creating or Editing a Chart

You can create or edit a chart to visualize data using SQL commands.

To create a new chart, click **Create Chart**.
To edit a chart, click the context menu icon and select **Edit**.

Enter the following details:

1. **Name**: Enter a name for the chart.
2. **Description**: Enter a description for the chart.
3. **Protected by Privilege**: Select an available privilege from the drop-down list to enable only those with this privilege to securely access the chart.
4. **Published**: When enabled, you can share and access the chart externally.

**Note:**

You can also protect a chart by protecting the corresponding module using the REST pages. For more information, see *Editing a Resource Module*. When a chart is protected, a user must authenticate before viewing the chart.

5. Click **Next**.
6. Enter the SQL statement in the command editor on the left. Ensure that you do not end the SQL statement with a semicolon.

The chart settings are automatically updated in the **Chart Definition** section on the right.

7. Edit/Update the chart settings in the **Chart Definition** section. The fields in the mapping section varies according to the **Chart Type** selected.

When the **Chart Type** selected is one of the following: *Area Chart, Bar Chart, Line Chart, or Line with Area Chart*, the mappings displayed are as follows:

- **Orientation**: Choose between horizontal and vertical orientation types from the drop-down list.
- **X axis label and Y axis label**: Optionally enter labels for X axis and Y axis.
- **Label**: The label is auto-generated from the SQL statement which represents the X axis.
- **Value**: The Value is auto-generated from the SQL statement which represents the Y axis (the values to be plotted on the chart).
- **Refresh Columns**: Refresh the data of all the columns selected in the SQL statement.
- **Coordinate System**: Choose between Cartesian and Polar coordinate systems from the drop-down list.
- **Sorting**: Choose a sorting type (ascending, descending, off) for plotting the values. If selected, the Sorting value overrides the order specified by the **ORDER BY** clause in the SQL statement. For a large amount of data, Oracle recommends using the **ORDER BY** clause to sort data.
When the **Chart Type** selected is one of the following: *Pie Chart, Pyramid Chart, or Funnel Chart*, the mappings displayed are as follows:

- **Style**: Select from either 2D or 3D styles.
- **Series ID**: The Series ID is auto-generated from the SQL statement which represents the category to be plotted.
- **Value**: The Value is auto-generated from the SQL statement which represents the values to be plotted for each category.

**Preview/Auto-preview**: Click preview to view the changes made to the chart. If Auto-preview is checked, the changes to the chart are refreshed automatically.

8. The **Chart** tab at the bottom displays the chart.
9. The **Data** tab at the bottom displays the data resulting from the SQL command.
10. Click **Create** to create the chart, or click **Save** when editing the chart.

---

**Note:**

Due to browser resource restrictions, the preview of charts is limited to 3000 rows from your SQL query results. However, once you create a chart, all the data is rendered in the standalone view.

---

**Example: Creating and Editing a Chart**

This section provides an example of creating and editing a chart.

**Creating a Chart**

The following example shows how to create a chart to understand the cumulative salary earned by each job category.

1. Navigate to the **Charts** page from Database Actions homepage.
2. Click **Create Chart**.
3. Enter the details of the chart as follows:
   - **Name**: Total Salary by Job Category
   - **Description**: The cumulative salary earned by each job category.
   - **Chart Definition**:
     - **SQL Statement**:

```sql
SELECT SUM(SALARY), JOB_ID
FROM EMPLOYEES
GROUP BY JOB_ID
```

- **Chart Type**: Area Chart
- **Orientation**: Vertical
- **X axis label**: Job Category
- **Y axis label**: Total Salary
- **Label**: JOB_ID (VARCHAR2)
- **Value**: SUM(SALARY) (NUMBER)
- **Coordinate system**: Cartesian
- **Sorting**: Off

- Click **Preview** in the Mapping section to preview the chart below.

- Change the **Chart Type** to **Bar Chart**, **Sorting** to **Descending** and click **Preview** to view the updated chart.

4. Click **Create** to create the chart. A confirmation notification is displayed and the chart card is visible in the Charts page.
5. Click the Selector and select **View Chart** to view the chart in a new tab.

![Total Salary by Job Category chart](chart.png)

**Editing a Chart**

The following example edits the chart created in the previous section.

1. Click the Selector and select **Edit** to edit the chart.

![Charts interface](charts_interface.png)

2. In the Mapping section, change the **Chart Type** to **Pie Chart** and **Style** to **3D**.
3. Click Preview to view the updated chart.

4. Click Save.

5. Click the Selector and select View Chart to view the updated chart in a new tab.

Creating or Editing a Dashboard

This section describes the steps to create or edit a dashboard.

To create a dashboard, in the Charts and Dashboard page, at the top right, click Create and then select New Dashboard.

To edit a dashboard, click the Actions icon and select Edit.

1. In the Create a Dashboard pane, enter or edit the following fields:
• Name: Enter a name for the dashboard.
• Published: Select this option to make the dashboard available for use.
• Description: Enter a description for the dashboard.
• Protected by Privilege: Select an available privilege from the drop-down list to enable only those with this privilege to securely access the dashboard.
• Go to Dashboard details after creation: Select this option to go to the main Charts and Dashboards page after creating the dashboard.

2. Click Create or click Save, if editing. The Layout page is displayed.

3. In the Layout page, you can add charts and design the layout for the dashboard. Click Add Chart.
   • In the Select Chart pane, for the Selection tab, select the chart to add from the drop-down list.
     You can select the Custom option and enter a URL for the chart from your schema or from a different schema. The URL must point to the REST Module root, for example, http://xxxx.oracle.com:1234/ords/hr/sdw/charts/Chart01/. Also, you need to have the associated privileges to view the chart.
     A preview of the selected chart is displayed. You can change the refresh rate of the chart, display the chart in Chart or tabular format, or share the chart by copying the URL.
   • In the Settings tab, select Chart and Data Grid to enable both views for the chart.
     By default, the Chart view is displayed in Preview.

4. Click Create.
   The chart is added to the Layout section.

5. Add more charts as needed to the dashboard. The maximum is 3 or 4 based on the number of columns that you select from the Columns drop-down list.
   • You can reorder the charts by dragging them horizontally or vertically to move them to the required position.
• Use the Edit icon on a chart to edit the chart if needed. You can also change the chart by selecting from the drop-down list.
• Drag the double-sided arrow at the right corner to resize the chart.

6. Click **Save and Preview** to save the layout and display a preview of the dashboard.

The dashboard is displayed on a new page. For each chart, you can change or customize the refresh rate, display the chart in Chart or tabular format, or share the chart by copying the URL.
The Scheduling Pages

Note:
Available for Oracle Database 12c release 2 and later releases.

Database Actions provides a graphical interface for using the DBMS_SCHEDULER PL/SQL package to work with Oracle Scheduler objects such as Jobs, Chains, Programs and Schedules. To use the scheduling features, you must first understand the concepts and essential tasks for job scheduling, which are explained in the chapters about Oracle Scheduler concepts and scheduling jobs in the Oracle Database Administrator's Guide.

In the initial release, the Scheduling functionality is focused on exploring Oracle Database Scheduler artifacts and provides details for jobs, chains, chain steps and rules, job execution history, programs and schedules. The job execution forecast wizard is useful for planning job schedules. Depending on the access rights of the logged-in user, information is taken from ALL_* or DBA_* views.

To navigate to the Scheduling pages, do either of the following:

• In the Launchpad page, click Scheduling.

• Click Selector to display the navigation menu. Under Development, select Scheduling.

Topics
• Overview
• Jobs
• Chains
• Programs
• Schedules
• Objects

Overview

The Overview page provides an overview of the activity in the Scheduling pages. It consists of two parts: Objects and Problem Jobs.

The Objects section provides an overview of the total number of running jobs, running chains, programs and schedules. Click an object tile to go to the corresponding page.
Click **Scheduler Summary** at the top right to view details such as Running Jobs, Current open window, Log history retain, Default timezone, Event expiry time, File watcher count and repeat interval.

This Problem Jobs section provides information about jobs with problems in their execution. Listed in tabular form are jobs having one of the following status: Failed, Broken, Chain_Stalled, Blocked, Retry Scheduled, Resource_Unavailable. The **Actions** context menu provides links to the history and job details for each problem job.

---

**Note:**

If there are no jobs with problems, the Jobs Summary page appears as the initial page for the Scheduling functionality.

---

**Jobs**

There are five options available in the Jobs menu: Summary, Running, Forecast, History, Notifications. Each option is described below.

**Summary**

At the top, the Summary page provides an overview of the total number of Failed, Blocked, Chain_stalled, Resource_unavailable, Broken, and Retry Scheduled jobs. Click a job tile and the corresponding list of jobs are displayed in tabular format below. Remove the filters to display all scheduled jobs.

To create a job, see **Create or Edit Job**.

The Actions icon is available at the end of each job row. Click **Actions** to view the following list of options:

- **Job**
  - **Run**: Runs the specific job.
  - **Edit**: See **Create or Edit Job**.
– **Drop**: Drops the specific job.

– **Job-Enable/Disable**: If enabled is selected, the job is picked up by the Scheduler for processing. The status of the job (enabled or not) is seen in the Job Details page, where the property “enabled” displays TRUE or FALSE.

- **History**
  - **Report**: Provides a history of the job runs in a report format, including log details, status of the run, run duration, errors if any, and so on.
  - **Chart**: Provides a history of the job runs in a visual bar chart representation. Run duration and used CPU time are presented for each execution. Jobs in chains are presented with aggregated information for the whole job. There is an overview scrollbar that enables zoom and scroll by changing the time frame. You can display all details from logs for that time frame for a selected job or for all jobs.

- **Job Forecast**: You can execute the job forecast functionality for a single job (available in the Action menu) or for a set of jobs (available on the toolbar for the table). For a set of jobs, filtering can be used to narrow the set of jobs. Use Ctrl key+click to select some jobs and then Job Forecast will run for the selected jobs only. If there are no selected jobs, then forecast will run for all listed jobs. Not every job included in the set is included in the final forecast. Only jobs with a defined calendar (repeat interval) are included. You can define the calendar inline or using schedule, window or window group. After the forecast is done, you can select different zoom levels and filter the results by schema.

- **Job Details**: Displays job attributes, such as action, job class, type, schedule, and so on. Select JSON to view the attributes in JSON format. Depending on the job’s definition, details about used objects are provided such as used program, PL/SQL code, procedure, procedure dependencies, program arguments, job arguments, job class, schedule, window, window group together with windows in the group, file watcher, history chart presenting one week of history from last start date.

**Running**

The Running page displays the list of currently running jobs.
History

The History page displays log run details for all Scheduler jobs available to the user. You can use the History window and set filters to limit the amount of data. You can filter using delayed jobs by providing a delay interval. Ordering is supported on the grid by clicking the column header.

Forecast

The Forecast page enables the job execution forecast for all available jobs. This operation takes time to finish depending on the number of jobs and forecast interval.

Notifications

You can create notifications only if an email server is set for the scheduler.

The Notifications page enables you to view, create, edit and delete notifications related to job events. Also, you can see email server details by clicking the notifications Email Server icon at the top right of the page.

For each message, you can specify job, recipient email addresses and sender (or no sender), and you can modify the subject and body of the message and set the filter condition. If multiple recipients and events are provided, then Oracle Scheduler creates a separate notification record for each combination <recipient,event>. You can edit the content of a set of notifications created in this way by using Edit Aggregated, which appears in the context menu of each notification. Edit edits the content of a single notification.
Notifications can be filtered by job name, job owners, recipients and events. You can select and remove some notifications by clicking the **Remove Notifications** icon. If there are no selected notifications (use **Ctrl+click** to deselect a single notification), then the Remove Notifications dialog appears enabling removal of all notifications for the selected job, or job and recipients, or job and events, or job and events and recipients.

![Remove Notifications dialog](image)

### Create or Edit Job

This section describes how to create a new Oracle Scheduler job or edit an existing job.

To create a job, Database Actions internally uses the `DBMS_SCHEDULER.CREATE_JOB` procedure, which is documented in *Oracle Database PL/SQL Packages and Types Reference*.

1. In the Jobs page, at the top right, click **Create Job**.
2. In Job Properties, enter the following fields:
   - **Enabled**: If this option is specified, validity checks are made and the job is created enabled if all the checks are successful. If this option is not specified, the job is not created enabled.
   - **Name**: Name of the job.
   - **Description**: Optional text string that can be used to describe the job.
• **Type**: Type of object to be executed by the job: PL/SQL Block, Chain, Stored Procedure, Named Program, or Script. Additional controls appear for Chain, Stored Procedure and Named Program enabling you to select related objects.

For Stored Procedure, only procedures with IN parameters are listed, procedures with IN OUT or OUT parameters are not permitted.

Schema level and package level procedures are listed for the selected schema but you can directly type the procedure name (or package_name.procedure_name) in the field.

• **Class**: Name of the job class to which this job belongs.

**Execution Mode**

• **Mode**: When to execute the job: **Immediate** (immediately on creation, and once only), **Once** (once, at a specified time), **Repeating**, **Queue**, **File Watcher**, **Schedule** (using a named schedule object), **Window** and **Window group**. If you specify anything other than Immediate, you are prompted for additional information.

For Repeating, you can manually define the repeat interval or click the pencil icon to select the date, frequency, weekday, and interval values. Some clauses of Oracle calendar syntax (include, exclude, intersect, periods and by period) are not supported and a warning is displayed when the edit icon is clicked.

**Destination**

• **Local** (local system), **Remote** (the database destination for a remote database job, or external destination for a remote external job), or **Multiple** (the job runs on all destinations associated with the provided destination group). Depending on what destination you selected for the job, select the local credential, the remote credential and destination, or the destination group.

**Properties**

• **Auto Drop**: Determines whether the job is to be automatically dropped after it has completed or has been automatically disabled.

• **Restart on failure**: Determines whether the job can be restarted in case of failure.

• **Restart on recovery**: Determines whether to restart the job in case of database failure.

• **Store Output**: If enabled, then for job runs that are logged, all job output and error messages are stored in the *_JOB_RUNDETAILS views. If disabled, then the output and messages are not stored.

• **Follow Default Time Zone**: Determines whether if the job start date is null, then when the default time zone scheduler attribute is changed, the Scheduler recomputes the next run date and time for this job so that it is in accordance with the new time zone.

• **Allow Runs in Restricted Mode**: If enabled, the job is permitted to run when the database is in restricted mode, provided that the job owner is permitted to log in during this mode.

• **Stop on Window Close**: If the schedule of a job is a window or a window group, enabling this option causes the job to stop once the associated window is closed, and disabling causes the job to continue after the window closes. (Note that if the job is allowed to continue, its resource allocation will probably...
change because closing a window generally also implies a change in resource plans.)

- **Instance Stickiness**: This attribute should only be used for a database running in an Oracle Real Application Clusters (Oracle RAC) environment. By default, it is enabled. Jobs start running on the instance with the lightest load and the Scheduler thereafter attempts to run on the instance that it last ran on. If that instance is either down or so overloaded that it does not start new jobs for a significant period of time, another instance runs the job. If the interval between runs is large, instance_stickiness is ignored and the job is handled as if it were a non-sticky job. If instance_stickiness is disabled, each instance of the job runs on the first instance available.

- **Parallel Instances**: For an event-based job, determines what happens if an event is raised and the event-based job that processes that event is already running. If disabled, it causes the new event to be ignored. If enabled, it causes an instance of the job to be started for every instance of the event, and each job instance is a lightweight job so multiple instances of the same event-based job can run in parallel.

- **Job Style**: Style of the job being created: REGULAR (regular job) or LIGHTWEIGHT (lightweight job). A lightweight must reference a program object. Use lightweight jobs when you have many short-duration jobs that run frequently. Under certain circumstances, using lightweight jobs can deliver a small performance gain.

- **Job Priority**: The priority of this job relative to other jobs in the same class as this job. If multiple jobs within a class are scheduled to be executed at the same time, the job priority determines the order in which jobs from that class are picked up for execution by the job coordinator. It can be a value from 1 through 5, with 1 being the first to be picked up for job execution.

- **Logging Level**: Determines how much information is logged: DBMS_SCHEDULER.LOGGING_OFF (no logging), DBMS_SCHEDULER.LOGGING_FAILED_RUNS (only jobs that failed, with the reason for failure), DBMS_SCHEDULER.LOGGING_RUNS (all runs of each job in this class), or DBMS_SCHEDULER.LOGGING_FULL (all operations performed on all jobs).

  However, if the job class has a higher (more detailed) logging level than the level specified for the job, the job class logging level is used.

- **Instance ID**: In an Oracle Real Application Clusters environment., the instance ID of the instance that the job must run on.

- **Max Runs**: The maximum number of consecutive scheduled runs of the job.

- **Max Failures**: The number of times a job can fail on consecutive scheduled runs before it is automatically disabled.

- **Raise Events**: Determines at what stages of the job execution to raise events.

- **Max Run Duration**: Maximum amount of time that the job should be allowed to run. Its data type is INTERVAL DAY TO SECOND. If this attribute is set to a nonzero and non-null value, and job duration exceeds this value, the Scheduler raises an event of type JOB_OVER_MAX_DUR. It is then up to your event handler to decide whether or not to allow the job to continue.

- **Schedule Limit**: Maximum delay time between scheduled and actual job start before a program run is canceled.

- **Reset to Defaults**: Resets all properties to their default values.
Enables you to set the NLS-related property required for this job. To enter a value, type it in to the related field for the Value column.

For a new job, parameters are taken from the database session.

3. In the **DDL** pane, you can review and save the SQL statements that are generated when creating or editing the job.
   - For a new job, click **CREATE** to view the generated DDL statements.
   - When you edit a job, click **UPDATE** to view the generated ALTER statements.

When you are finished, click **Apply**.

The **Output** pane displays the results of the DDL commands. If there are any errors, go to the corresponding pane, fix the errors, and run the commands again. You can save the content to a file.

---

**Chains**

There are four options available in the Chains menu: Summary, Running Chains, Steps, Rules. Each option is described below.

**Summary**

The Summary page displays all the chains owned by the current user. The details are presented in a tabular format. At the end of each row is the Actions icon 

Click **Actions** to view the list of options for the chain. Click a chain in the table and details about steps and rules appear below the Chains table. There is a **Drop Step** or **Drop Rule** option available in the context menu for a step or rule.

The actions available for a chain are:

- **Run**: Runs the specific chain.
- **Add Step**: Creates a new step for the selected chain.
- **Add Rule**: Creates a new rule for the selected chain.
- **Edit**: Edits the chain properties.
- **Drop**: Drops the chain
- **Chain-Enable/Disable**: Enables or disables the chain.
- **Show Diagram**: Displays the job steps of a chain in a visual diagram format.
Double-click a step or rule in the diagram to see more details. You can directly create steps and rules on the diagram using the **Add Step** and **Add Rule** icons.

You can also create a rule by connecting two steps:

- Click on a step and the link icon appears.
- Click the link icon and drag to the next step. The Rule dialog appears where you can provide details.

**Analyze Chain**: Displays output from the `DBMS_SCHEDULER.ANALYZE_CHAIN` procedure in text format. Lists the type of steps, dependencies and transitions between them based on defined rules.

**Running Chains**

The Running Chains page displays the execution of the steps of the currently running chains.

**Steps**

The Steps page provides details of each step in all chains.

**Rules**

The Rules page provides details of each rule in all chains.
Create or Edit Chain

This section describes how to create or edit a chain.

To create a chain, Database Actions internally uses the DBMS_SCHEDULER.CREATE_CHAIN procedure.

1. In the Chains page, at the top right, click Create Chain.

2. In Properties, enter the following fields:
   - **Name**: Name to assign to the chain.
   - **Evaluation Interval**: NULL reevaluates of the rules of a running chain only when the job starts and when a step completes. A non-NULL value causes rule evaluations to also occur periodically at the specified interval.
   - **Enabled**: Enables the chain. (Causes the DBMS_SCHEDULER.ENABLE procedure to be called after the chain is created.)

3. Click Create to create the chain.
   Click Create and Open to create the chain and open in Diagram mode.

Add or Edit Step

This section describes how to add a step for a chain.

1. In the context menu for a chain, select Add Step.

2. In the Create Step slider, enter the following fields:
   - **Step Name**: Name of the step.
   - **Type**: Select Program, Chain, Event or Queue to run during the step. Additionally, select the name of the program, chain, event or queue in the corresponding field.
   - **Pause**: Select this option to pause the step.
   - **Skip**: Select this option to skip the step.
   - **Restart on Failure**: Select this option to restart the step after database failure.
   - **Restart on Recovery**: Select this option to restart the step after database recovery.

3. Click Create.
   The new step is displayed for the chain in the main Chains page.

Add or Edit Rule

This section describes how to add a rule for a chain.

1. In the context menu for a chain, select Add Rule.

2. In the Create Rule slider, enter the following fields:
   - **Rule Name**: Name of the rule being created.
   - **Condition**: Enter the condition to evaluate.
• **Action**: Action to perform if condition is met.
• **Comments**: Optional comments describing the rule.

See Also:
**DEFINE_CHAIN_RULE Procedure** in *Oracle Database PL/SQL Packages and Types Reference*

---

**Programs**

The Programs page displays the list of created programs.

The Actions menu for a program consists of the following options: Enable or disable the program and View program details, including used procedure and procedure dependencies.

---

**Create or Edit Program**

This section describes how to create or edit an Oracle Scheduler program.

To create a program, Database Actions internally uses the **DBMS_SCHEDULER.CREATE_PROGRAM** procedure.

1. In the Programs page, at the top right, click **Create Program**.
2. In Program Properties, enter the following fields:

   **Details** tab
   - **Name**: Name of the program. The name has to be unique in the SQL namespace. For example, a program cannot have the same name as a table in a schema.
   - **Enabled**: If this option is specified, validity checks will be made and the program will be created enabled if all the checks are successful. If this option is not specified, the program is not created enabled.
   - **Description**: Optional text string that can be used to describe the program.
• **Type**
  
  – **PL/SQL Block**: The program is a PL/SQL block. Job or program arguments are not supported when the job or program type is PLSQL_BLOCK. In this case, the number of arguments must be 0. Enter or paste in the complete PL/SQL code, or edit the existing code.
  
  – **Stored Procedure**: The program is a PL/SQL or Java stored procedure, or an external C subprogram. Only procedures, not functions with return values, are supported. PL/SQL procedures with IN OUT or OUT arguments are not supported.
    
    **Schema**: Schema of the stored procedure. If not specified, the schema of the job is assumed.
    
    **Procedure**: Name of the stored procedure.
    
    **Arguments**: For each argument, name, data type, default value, and whether it is an input, output, or input/output argument.
  
  – **Script**: The program is a SQL Script (SQL*Plus statements), Backup Script (RMAN commands), or External Script (operating system commands). Enter or paste the script text in the box.

**Properties** tab

Enables you to set program properties. For most properties the default is null, but you can check the box to specify a value.

• **Detached**: Enabled if the program is a detached job. Use a detached job to start a script or application that runs in a separate process, independently and asynchronously to the Scheduler. A detached job typically starts another process and then exits. Upon exit (when the job action is completed) a detached job remains in the running state. The running state indicates that the asynchronous process that the job started is still active. When the asynchronous process finishes its work, it must connect to the database and call DBMS_SCHEDULER.END_DETACHED_JOB_RUN, which ends the job.

• **Max Runs**: Maximum number of runs before the program is marked as completed.

• **Max Failures**: Maximum number of failures tolerated before the program is marked as broken.

• **Max Run Duration**: Maximum run duration of the program.

• **Schedule Limit**: Maximum delay time between scheduled and actual job start before a program run is canceled.

3. In the **DDL** pane, you can review and save the SQL statements that are generated.

   • For a new program, click **CREATE** to view the generated DDL statements.
   
   • When you edit a program, click **UPDATE** to view the generated ALTER statements.

   When you are finished, click **Apply**.

4. The **Output** pane displays the results of the DDL commands. If there are any errors, go to the corresponding pane, fix the errors, and run the commands again. You can save to a text file or clear the output.
Schedules

The Schedules page displays the list of available schedules.

The Actions menu for a schedule has the following options: Enabling or disabling the schedule, Viewing schedule details, Editing or Dropping the schedule.

Create or Edit Schedule

This section describes how to create or edit Oracle Scheduler schedule.

To create a schedule, Database Actions internally uses the DBMS_SCHEDULER.CREATE_SCHEDULE procedure.

1. In the Schedules page, at the top right, click Create Schedule.
2. In Program Properties, enter the following fields:
   - **Properties tab**
     - **Name**: Name of the schedule. The name has to be unique in the SQL namespace. For example, a schedule cannot have the same name as a table in a schema.
     - **Description**: Optional text string that can be used to describe the schedule.
     - **Mode**: Specify when jobs that use this schedule are to run: Repeating (specify the repeat interval, start date, and end date), Queue (specify the queue name, agent, condition, start date, and end date), or File Watcher (specify the file watcher object name, condition, start date, and end date).
3. In the DDL pane, you can review and save the SQL statements that are generated.
   - For a new schedule, click CREATE to view the generated DDL statements.
   - When you edit a schedule, click UPDATE to view the generated ALTER statements. When you are finished, click Apply.
4. The Output pane displays the results of the DDL commands. If there are any errors, go to the corresponding pane, fix the errors, and run the commands again. You can save to a text file or clear the output.

Objects
The Objects menu in Scheduling provides links to the following four pages: Job Classes, File Watchers, Windows and Window Groups

**Job Classes**
A job class is an Oracle Scheduler object that enables the Scheduler administrator to group jobs for logical purposes, such as to assign the same set of attribute values to member jobs, to set service affinity for member jobs, to set resource allocation for member jobs, or to group jobs for prioritization.

The Job Classes page displays the job classes created in a tabular format. You can perform the following actions:

- **Create**: To create a job class, see [Create Job Class](#).
- **Edit**: To edit a job class, select **Edit** from the Actions menu at the end of a row. For a description of the fields, see [Create Job Class](#).
- **Drop**: To drop a job class, select **Drop** from the Actions menu.
- **Show DDL**: Displays the Data Definition Language statements for the job class. To show DDL, select this option from the Actions menu.

**File Watchers**
A file watcher is an Oracle Scheduler object that defines the location, name, and other properties of a file whose arrival on a system causes the Scheduler to start a job. You create a file watcher and then create any number of event-based jobs or event schedules that reference the file watcher. When the file watcher detects the arrival of the designated file, it raises a file arrival event. The job started by the file arrival event can retrieve the event message to learn about the newly arrived file.

The File Watchers page displays the file watchers created in a tabular format. You can perform the following actions:

- **Create**: To create a file watcher, see [Create File Watcher](#).
- **Edit**: To edit a file watcher, select **Edit** from the Actions menu at the end of a row. For a description of the fields, see [Create File Watcher](#).
- **Drop**: To drop a file watcher, select **Drop** from the Actions menu.
- **Show DDL**: Displays the Data Definition Language statements for the file watcher. To show DDL, select this option from the Actions menu.

**Windows**
A window is an Oracle Scheduler object that can be used to automatically start jobs or to change resource allocation among jobs during various time periods of the day, week, and so on.

The Windows page displays windows created in a tabular format. You can perform the following actions:

- **Create**: To create a window, see [Create Window](#).
- **Edit**: To edit a window, select **Edit** from the Actions menu at the end of a row. For a description of the fields, see [Create Window](#).
- **Drop**: To drop a window, select **Drop** from the Actions menu.
- **Show DDL**: Displays the Data Definition Language statements for the window. To show DDL, select this option from the Actions menu.

**Window Groups**
A window group is an Oracle Scheduler object that is a list of Oracle Scheduler Windows. Scheduler jobs that are scheduled to be run in a window group will be activated in that time span and using that resource plan for all windows in the group.

The Window Groups page displays windows created in a tabular format. You can perform the following actions:

- **Create**: To create a window group, see Create Window Group.
- **Edit**: To edit a window group, select Edit from the Actions menu at the end of a row. For a description of the fields, see Create Window Group.
- **Drop**: To drop a window group, select Drop from the Actions menu.
- **Show DDL**: Displays the Data Definition Language statements for the window group. To show DDL, select this option from the Actions menu.

## Create Job Class

To create a job class:

1. In the Job Class page, click Create Job Class.
2. Enter the following fields:
   - **Name**: Name of the job class.
   - **Description**: Optional text string that can be used to describe the job class.
   - **Logging Level**: Specifies how much information is written to the job log:
     - **RUNS**: Detailed information for all runs of each job in this class.
     - **FULL**: RUNS plus all operations performed on all jobs in this class.
     - **OFF**: No logging.
   - **Log Retention Period (days)**: Number of days that job log entries for jobs in this class are retained. The range of valid values is 0 through 999. If set to 0, no history is kept. If NULL (the default), retention days are set by the log_history Scheduler attribute.
   - **Service Name**: The database service that the jobs in this class will have affinity to. If no service is specified, the job class will belong to the default service, which means it will have no service affinity and any one of the database instances within the cluster might run the job.
   - **Resource Consumer Group**: Resource consumer group this class is associated with. If no resource consumer group is specified, the job class is associated with the default resource consumer group.
3. Click Create.

## Create File Watcher

To create a file watcher:

1. In the File Watchers page, click Create File Watcher.
2. Enter the following fields:
• **Enabled**: Enables the file watcher. (Causes the DBMS_SCHEDULER.ENABLE procedure to be called after the file watcher is created.)

• **Name**: Name of the file watcher.

• **Description**: Optional descriptive text.

• **Destination**: Name of an external destination. You create an external destination by registering a remote Scheduler agent with the database. The view ALL_SCHEDULER_EXTERNAL_DESTS lists valid external destination names. If this parameter is null, the file watcher is created on the local host.

• **Credential Name**: Name of a valid credential object. The file watcher uses the credential to authenticate itself with the host operating system to access the watched-for file. The file watcher owner must have EXECUTE privileges on the credential.

• **Directory Path**: Directory in which the file is expected to arrive. The single wildcard '?' at the beginning of the path denotes the Oracle home path. For example, '?/rdbms/log' denotes the rdbms/log subdirectory of the Oracle home directory.

• **File Name**: Name of the file to look for. Two wildcards are permitted anywhere in the file name: '?' denotes any single character, and '*' denotes zero or more characters.

• **Min File Size**: Minimum size in bytes that the file must be before the file watcher considers the file found.

• **Steady State Duration**: Minimum time interval that the file must remain unchanged before the file watcher considers the file found. Cannot exceed one hour. If null, an internal value is used. The minimum value is 10 seconds. Oracle recommends similar steady state duration values for all file watchers for efficient file watcher job operation. Also, the repeat interval of the file watcher schedule must be equal or greater than the steady state duration value.

3. Click **Create**.

### Create Window

To create a window:

1. In the Windows page, click **Create Window**.

2. Enter the following fields:
   - **Name**: Name of the window.
   - **Description**: Optional descriptive text.
   - **Enabled**: Enables the window. (Causes the DBMS_SCHEDULER.ENABLE procedure to be called after the window is created.)
   - **Resource Plan**: The resource plan that automatically activates when the window opens. When the window closes, the system switches to the appropriate resource plan, which is usually the plan that was in effect before the window opened, but can also be the plan of a different window. Only one resource plan can be associated with a window. It null, the resource plan in effect when the window opens stays in effect for the duration of the window. If
an empty string, the resource manager is disabled for the duration of the window. If the window is open and the resource plan is dropped, then the resource allocation for the duration of the window is not affected.

- **Duration**: The length of time that the window stays open. Can range from one minute to 99 days.
- **Priority**: Relevant when two windows overlap. Because only one window can be in effect at one time, the window priority determines which window opens. The two possible values for this attribute are **HIGH** and **LOW**. A high priority window has precedence over a low priority window, therefore, the low priority window does not open if it overlaps a high priority window.
- **Execution Mode**: Repeating or Schedule:
  - For **Repeating**, specify the **Repeat Interval** and optionally the **Start Date** and **End Date**.
  - For **Schedule**, select the name of the **Schedule** to be used.

3. Click **Create**.

### Create Window Group

To create a window group:

1. In the Window Groups page, click **Create Window Group**.
2. Enter the following fields:
   - **Name**: Name of the window group.
   - **Description**: Optional comments.
   - **Enabled**: Enables the window group. (Causes the DBMS_SCHEDULER.ENABLE procedure to be called after the window group is created.)
   - **Available Windows**: Lists all Scheduler windows.
   - **Selected Windows**: List windows to be added to the window group. Use the arrow icons to move selected windows or all windows from one list to the other.
3. Click **Create**.
The Liquibase Page

The Liquibase page displays information of all database deployments made in the current schema. A deployment consists of changesets, which is a list of sequential changes to be applied to the database.

To navigate to the Liquibase page, do either of the following:

• In the Launchpad page, click Liquibase.

• Click Selector to display the navigation menu. Under Development, select Liquibase.

If you have not made any deployments into the schema and you access the Liquibase page, no data appears.

Topics:

• Generate a Deployment

• About the Liquibase User Interface

Generate a Deployment

First, you have to make changes in the current schema such as creating a table, procedure or any database object.

Then, create a folder that will contain the changesets for the deployment. It is a good practice to create a different folder for each deployment. This ensures that you have information of all the deployments, in case you want to roll back to a previous version.

In the following example, Oracle SQLcl is used to deploy the changes.

1. Open the terminal and navigate to the path of the folder that you will use to save the changes made to the database (changelog files).

2. After you navigate to the correct path, log in to the schema you want to capture with the following command:

```
sqlcl <username>/<password>@<host>:<port>/<servicename>
```

3. You can do one of the following:

• Deploy the entire schema with the following command:

```
lb genschema
```

This command creates a controller.xml file that includes all the changesets for the schema.
• Deploy a specific object using the following command:

```
lb genobject -type <object_type> -name <object_name>
```

This command generates the changelogs (which are XML files) that contain the changes made to the database in the current folder.

4. Log in to the schema where you want to add these changes (in this example, XYZ) with the following command:

```
sqlcl xyz/<password>@<host>:<port>/<servicename>
```

5. Run the following command:

```
lb update -changelog controller.xml (or the file name of your changelog)
```

6. Log in to Database Actions on the schema where you deployed the changes and the Liquibase page displays information about the deployment.

### About the Liquibase User Interface

In the Liquibase page, the card at the top displays the Liquibase status. The deployments are displayed as cards below, as shown in the following figure.

**Liquibase Status**

Liquibase runs only one deployment at a time. When there is no deployment running, the top Liquibase card indicates that Liquibase is currently unlocked and displays the ID of the last deployment.

If there is a deployment running at the time, the card indicates that Liquibase is locked, and displays the name of the resource blocking it and the time at which the deployment started.
Deployments

The card for each deployment provides details such as the schema, deployment id, start and complete deployment date and time, and number of changesets run in the deployment. To open a specific deployment page, click the title of the card, or click Open Deployment Changes in the context menu.

Changesets

In a deployment page, the changesets that have been run for that specific deployment is displayed below. The Current Deployment card at the top shows the deployment to which the changesets belong.

Each changeset card displays the changeset name, the date and time when the changeset started deploying, and the schema where the change was deployed.

The context menu for a changeset consists of the following options:
- **Previous Object State XML**: Displays an XML showing the state of the object (table, procedure) before the changeset was applied. If the object did not exist previously on the database (example, a table that was created in the deployment), the following message is displayed: `xml code not available....`

- **Show Executed SQL**: Displays the SQL code that was executed when the changeset was deployed.
The Data Pump Page

Note:
This feature is only available for Oracle Database 12.2 and later releases.

The Data Pump page enables you to monitor Data Pump jobs that were initiated through the available Database API endpoints, the `DBMS_DATAPUMP` package, or the SQL Developer Data Pump Export and Import wizards.

The section at the top displays the total number of executing jobs, stopped jobs, and completed jobs. Click a tile (example, STOPPED) to filter and view the corresponding list of STOPPED jobs in the default card format.

You can filter or sort the jobs and set the time period by which to refresh the data.

A job card displays the following details: Job name, import or export operation, percentage of completion and time elapsed, and links to dump files and logs. The status of the job is indicated by the colour of the icon on the left side of the card. Green indicates successful jobs, yellow indicates that the jobs need to be reviewed, and blue indicates that the jobs are in progress.

In a job card, you can:

- Use Download to access dump files for completed jobs.
- Use Log to access the log files.
The Data Load Page

Use the Data Load page to make more data available to your Oracle Autonomous Database. You can load data from files or databases, from links to external databases or cloud storage files, or from a live feed of data from cloud storage.

From the Data Load page, you can also explore the data in your autonomous database and manage your cloud storage locations.

To reach the Data Load page, click Data Load in the Database Actions page, or click the Selector icon and select Data Load from the Data Tools menu in the navigation pane.

To load or create links to data or create live table feeds, on the Data Load page, select a combination of an operation and a data source location. The following table lists the operations and the source locations that support those operations.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Source Location</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load data</td>
<td>Local file</td>
<td>Load data from files on your local device, from remote databases, or from cloud storage into tables in your Oracle Autonomous Database.</td>
</tr>
<tr>
<td></td>
<td>Database</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Cloud storage</td>
<td></td>
</tr>
<tr>
<td>Link data</td>
<td>Database</td>
<td>Create external tables or views in your Oracle Autonomous Database that link to data in cloud storage or remote databases. Changes to the source data automatically appear in the target objects.</td>
</tr>
<tr>
<td></td>
<td>Cloud storage</td>
<td></td>
</tr>
<tr>
<td>Feed data</td>
<td>Cloud storage</td>
<td>Set up a feed of data from a cloud storage bucket into a table. Changes to the source data load into the target table as scheduled or on demand.</td>
</tr>
</tbody>
</table>

The following topics describe these actions.

- Exploring Data
- Managing Cloud Storage Connections
- Loading Data
- Linking Data
- Feeding Data

Checking Data Load Jobs

You can check an existing data load job and retrieve the data load job later when required.
The Data Load Job page displays the data load history in your Oracle Autonomous Database. On the **Data Load** Page, in the **Explore and Connect** section, select **Data Load Jobs**.

The Data Load Job page contains the **Search for Data Load Jobs** field, a list of Data load job cards. You can enter the data load job you are looking for in the field or click one of the data load jobs from the list.

The Data Load Jobs page consists of:

1. **Search for Data Load Jobs** field
   You can click the field and type or search for the name of the data load job you are looking for.

2. **Toolbar**
   The toolbar consists of the following buttons:
   
   - **Sort By**
     To select sorting values, click the **Sort By** button to open the list of options. Then click the **Ascending** or **Descending** icon next to one or more of the sorting values. For example, if you select the **Ascending** icon next to **Entity name** and the **Descending** icon next to **Entity type**, the entities will be sorted in alphabetical order by entity name and then in reverse alphabetical order by entity type.
     
     Click **Reset** in the list to clear the choices in the list.
     
     The sorting values you choose are listed next to the **Sort by** label beneath the toolbar. Click the **X** icon on a sorting value to remove it.
   
   - **Page size**
     By default, up to 25 entities are displayed on the page. If you want more entities on a page, select a number from this list.
   
   - **Previous** and **Next**
     If the search results are displayed on multiple pages, click these buttons to navigate through the pages.
   
   - **Refresh**
Click to refresh the data load jobs shown on the page, based on the current search field.

3. **Filters** panel

   Select one or more filter values to limit the data load jobs shown on the page. Only those entities that match the filter values are shown. The filter criteria are based on the schemas and how data is loaded or linked. That is, the items returned by a search are filtered by these filter settings. Selecting all or none of the options shows all entities.

4. **Display Area**

   The area beneath the **Search for Data Load Jobs** field displays the data load job carts returned by a search and that match the filter criteria set in the **Filters** panel. It displays list of carts which represents a list of previously run data load jobs.

You can view details about the job, re-run the Data Load job, Rename the Data Load Job and Delete the Data Load Job.

**View Details** about the Data Load Job

To view details about the existing data load job, click **Action** and select **View Details** in the card for the load job. Selecting **View Details** displays details like **Lineage**, **Impact** and **Log details** of the data load job.

For details on Lineage, Impact and Log details, see Viewing Entity Details.

**Rerun Data Load Job**

After viewing details about the selected data load or data link job whose sources are from cloud storage, you can re-run the selected previously run data load job. The previous files and folders processed in that job will be loaded in the cart with all your settings unchanged from the last run. On the left side of the page is a navigator pane, where you choose the cloud store connection and the folders or files containing the data that were previously run. The previously used cloud storage details are already present in the navigator pane. Select additional files or folders from the navigator pane and drop them in the Data Load Cart area. To change the cloud storage connection and the folders and files containing the data, see Managing Cloud Storage Connections.

Once you have added the data sources to the data load cart, click the **Start** icon in the data load cart menu bar. When the data load job completes, the Load Cloud Object page displays the results of the job. At the top of the page, a status message shows the number of items for which the load has completed over the number of items in the job and the total time elapsed for the job. If any previously loaded or linked files and folders are no longer present in the cloud location, an error message reports the problem. You can view the list of unavailable files and folders, which will not be loading into the cart for processing.

Click the **Cancel** icon to cancel re-running the selected job.

---

**Note:**

The **Rerun Data Load Job** option is visible for data jobs with Load and Link data options for Object storage.

---

**Rename Data Load Job**

To rename existing name of data load job cart, select the **Rename Data Load Job** option. The data load job names are system generated. This option enables you to rename the job to
be more descriptive. This makes it easier to search the next time you want to re-run
the data load job.

**Delete Data Load Job**

Select the **Delete Data Load Job** option to remove the previously run job from the
Data load Job page. This will also remove the log files and the error logs generated
when attempting to run the data load job.

**Exploring Data**

To view the data in the tables and views in your Oracle Autonomous Database, on the
Data Load page, in the Explore and Connect section, select **Explore**.

The navigator on the Explore page lists the tables and views in your Oracle
Autonomous Database. To filter the lists of tables and views displayed, enter a value in
the search field and press Enter. To refresh the list of tables and views, click the
Refresh icon.

To view details about a table or view, select the table or view in the navigator. The
Source Preview pane displays the columns and data of the table or view. For a table,
the pane includes a Statistics tab that displays the size and the number of rows and
columns of the table, the data type of the columns, the number of distinct values, and
other information. Below those details is a bar graph that displays the top unique
values for the selected column.

To close the Settings pane, click **Close**.

To return to the Data Load page, click **Data Load** at the top of the page.

**Managing Cloud Storage Connections**

A cloud storage link is a connection to a bucket in a cloud store. To view the existing
cloud storage links and to add new ones, on the Data Load page, select **CLOUD
LOCATIONS** and click **Next**.

The Manage Cloud page displays the existing cloud storage links, which are
CLOUD_STORAGE_LINK entities. You can edit or delete an link. You can also create
new cloud storage links.

You can filter and sort the display of links. By default, the links are filtered to display
those in the current user's schema, as indicated by the **Schemas** list below the search
field. The links are also sorted by entity name, as indicated by the **Sort by** list.

To include cloud storage links from other schemas, click the Display or hide filter panel
(funnel) icon. In the Filters section, select one or more schemas. If no schema is
selected as a filter, then the display includes cloud storage links from all available
schemas.

In the Sort by section, you can choose to sort the links by name, creation date, or both.

To remove a schema from the selected filters or to remove a sorting value, deselect
the schema or sorting value in the filter panel, or click the Remove filter (X) icon for the
schema or sorting value on the Manage Cloud page. To close the filter panel, click the
Hide filter panel (X) icon in the panel.

To search for available cloud storage links in the selected schemas, enter a value in
the search field and press Enter. The display then includes only the entities whose
names contain the characters in the search field. To clear the search field, click the Clear search results (X) icon in the search field.

To refresh the display of cloud storage links, click the Refresh icon.

In the Add Cloud Storage pane, fields with an asterisk (*) are mandatory. To cancel the creation of a link, click Cancel.

The procedure for creating a credential varies depending on the cloud storage provider. If your source files reside in a cloud store provided by one of the following, see the example for that provider.

- **Oracle Cloud Infrastructure (OCI)**, see Create an OCI Cloud Storage Link.
- **Amazon S3**, or you are calling an AWS API, see Create an Amazon S3 Cloud Storage Link.
- **Microsoft Azure Blob Storage** or you are calling an Azure API, see Create an Microsoft Azure Cloud Storage Link.
- **Google Cloud Storage**, see Create a Google Cloud Storage Link.
- **Other (Swift compatible)** cloud storage, see Create an Other (Swift Compatible) Cloud Storage Link.

Create an OCI Cloud Storage Link

1. On the Manage Cloud page, click Add Cloud Storage.
2. On the Add Cloud Storage pane, in the Name field enter a name for the cloud storage link. For example:

   My_Cloud_Store

3. (Optional) In the Description field, enter a description for the link. For example:

   My cloud storage link.

4. From the Cloud Store drop-down list, select Oracle.
5. In the URI + Bucket field, enter the URI and bucket for your OCI instance bucket.
   a. To get the URI and bucket, go to the bucket in the Object Storage compartment in your Oracle Cloud Instance.
   b. In the Objects group, click the Actions (three vertical dots) icon for a file in the bucket, then click View Object Details.
   c. Copy all of the URL Path (URI) except for the file name. Be sure to include the trailing slash. For example, for the file https://objectstorage.us-phoenix-1.oraclecloud.com/n/myoci/b/my_bucket/o/MyFile.csv, select the following:

   https://objectstorage.us-phoenix-1.oraclecloud.com/n/myoci/b/ my_bucket/o/
   
   d. Paste the string into the URI + Bucket field.
6. Select a credential option.
   - If you select No Credential, then proceed to the next step.
• If you select **Select Credential**, then select a credential from the drop-down list.

• If you select **Create Credential**, then do the following:
  
a. Enter a name in the **Credential Name** field. The name must conform to Oracle object naming conventions, which do not allow spaces or hyphens. For example:

    my_credential

b. For an OCI cloud store, in the **Oracle Cloud Infrastructure User Name** field, enter your OCI user name. For example:

    myUsername

c. For an OCI cloud store, in the **Auth Token** field, enter your auth token. For example:

    LPB>Kt.k{1M1SD+a]r

    For information on getting an auth token, see Working with Auth Tokens in Managing User Credentials

7. Click **Test**.
8. Click **Create**.

**Create an Amazon S3 Cloud Storage Link**

1. On the Manage Cloud page, click **Add Cloud Storage**.
2. On the Add Cloud Storage pane, in the **Name** field enter a name for the cloud storage link.

    My_Cloud_Store

3. (Optional) In the **Description** field, enter a description for the link. For example:

    My cloud storage link.

4. From the **Cloud Store** drop-down list, select **Amazon S3**.
5. In the **URI + Bucket** field, enter the URI and bucket for your Amazon S3 bucket. For example:

    https://s3-us-west-2.amazonaws.com/adwc/my_bucket

6. Select a credential option.
   • If you select **No Credential**, then proceed to the next step.
   • If you select **Select Credential**, then select a credential from the drop-down list.
   • If you select **Create Credential**, then do the following:
1. Enter a name in the **Credential Name** field. The name must conform to Oracle object naming conventions, which do not allow spaces or hyphens. For example:

   myCredential

2. In the **AWS Access Key ID** field, enter your AWS access key ID. For example:

   myAccessKeyId

3. In the **AWS Secret Access Key** field, enter your AWS secret access key. For information on AWS access keys, see *Managing access keys for IAM users*.

7. Click **Test**.

8. Click **Create**.

Create an Microsoft Azure Cloud Storage Link

1. On the Manage Cloud page, click **Add Cloud Storage**.

2. On the Add Cloud Storage pane, in the **Name** field enter a name for the cloud storage link. For example:

   My_Cloud_Store

3. (Optional) In the **Description** field, enter a description for the link. For example:

   My cloud storage link.

4. From the **Cloud Store** drop-down list, select **Microsoft Azure**.

5. In the **URI + Bucket** field, enter the URI and bucket for your Microsoft Azure bucket. For example:

   https://objectstore.microsoft.com/my_bucket

6. Select a credential option.

   • If you select **No Credential**, then proceed to the next step.
   
   • If you select **Select Credential**, then select a credential from the drop-down list.
   
   • If you select **Create Credential**, then do the following:

     a. Enter a name in the **Credential Name** field. The name must conform to Oracle object naming conventions, which do not allow spaces or hyphens. For example:

        my_credential

     b. In the **Azure Storage Account Name** field, enter the name of your Azure storage account. For example:

        AZURE_KEY123...

     c. In the **Azure Storage Account Access Key** field, enter your Azure access key. For information on Azure storage accounts, see *Create a storage account*.

7. Click **Test**.
8. Click **Create**.

**Create a Google Cloud Storage Link**

1. On the Manage Cloud page, click **Add Cloud Storage**.
2. On the Add Cloud Storage pane, in the **Name** field enter a name for the cloud storage link. For example:

   My_Cloud_Store

3. (Optional) In the **Description** field, enter a description for the link. For example:

   My cloud storage link.

4. From the **Cloud Store** drop-down list, select **Google**.
5. In the **URI + Bucket** field, enter the bucket and URI for your Google bucket. For example:

   https://my_bucket.storage.googleapis.com

6. Select a credential option.
   - If you select **No Credential**, then proceed to the next step.
   - If you select **Select Credential**, then select a credential from the drop-down list.
   - If you select **Create Credential**, then do the following:
     a. Enter a name in the **Credential Name** field. The name must conform to Oracle object naming conventions, which do not allow spaces or hyphens. For example:

        my_credential

     b. In the **HMAC Access Key** field, enter your HMAC access ID. For example:

        GOOGTS1C3LBP3KTSDKMB2BFD

     c. In the **HMAC Access Secret** field, enter your HMAC secret. For information on HMAC keys, see **HMAC Keys**.

7. Click **Test**.
8. Click **Create**.

**Create an Other (Swift Compatible) Cloud Storage Link**

1. On the Manage Cloud page, click **Add Cloud Storage**.
2. On the Add Cloud Storage pane, in the **Name** field enter a name for the cloud storage link. For example:

   My_Cloud_Store
3. (Optional) In the **Description** field, enter a description for the link. For example:

   My cloud storage link.

4. From the **Cloud Store** drop-down list, select **Other (Swift Compatible)**.

5. In the **URI + Bucket** field, enter the URI and bucket for your cloud store bucket. For example:

   https://someswiftcompatibleprovider.com/my_bucket

6. Select a credential option.
   - If you select **No Credential**, then proceed to the next step.
   - If you select **Select Credential**, then select a credential from the drop-down list.
   - If you select **Create Credential**, then do the following:
     a. Enter a name in the **Credential Name** field. The name must conform to Oracle object naming conventions, which do not allow spaces or hyphens. For example:

        my_credential

     b. In the **Access User Name** field, enter your access user name. For example:

        OTHER_KEY123...

     c. In the **Access Key** field, enter your access key.

7. Click **Test**.

8. Click **Create**.

## Loading Data

You can load data from files on your local device, from remote databases, or from cloud storage buckets.

The following topics describe the interfaces for these actions.

- Loading Data From Local Files
- Loading Data from Other Databases
- Loading Data from Cloud Storage

### Loading Data From Local Files

To load data from local files into your Oracle Autonomous Database, on the Data Load page, select **LOAD DATA** and **LOCAL FILE**, then click **Next**. Drag one or more files from your local file system navigator and drop them in the Data Load Cart. You can also click **Select Files** or the Select Files icon, select one or more files from the file system navigator, and then click **Open**.

You can add files in these file formats: AVRO, CSV, JSON, TSV, delimited TXT, XLS, XLSX, XML. For information on supported file formats, see Format Specifications for JSON, AVRO, and XML Files.
An item for each file appears in the cart. For an XLS or XLSX spreadsheet, the worksheets of the spreadsheet appear as individual items. The item shows the name of the source file or worksheet and its size, and the name of the table that is the target for the data load.

You can add more files to the cart by clicking the Select Files icon. You can add any number of files to the cart and load data from all of them in a single data load job.

To remove a source file from the Data Load Cart, click the Remove (trash can) icon for the source item. To remove all source files from the cart, click the Remove All (trash can) icon in the Data Load Cart menu bar.

To return to the Data Load page, click Data Load above the Data Load Cart menu bar.

Specify Processing Options

To specify settings for the data load job or preview the data in the source or the target, click the Settings (pencil) icon for the item in the Data Load Cart.

In the settings pane, on the Settings tab, you can view the name and size of the file in the title of the Load Data dialog box.

The Name field specifies the name of the target table. The value in the field varies depending on the selection in the Options field. If the option is Create Table, then the default target value is the name of source file or worksheet. To specify a different name for the target table, enter it in the Name field. For the other target table choices in the Options field, the default value is <None>. Expand the drop-down list and select an existing table as the target.

In the Options field select Create Table, Insert into Table, Replace Data, Drop Table and Create New Table, or Merge into Table. Point to the question mark icon to see a brief description of the selected option.

The Source column name option specifies whether to get the source and target column names from the file or to specify the column names manually. Getting the column names from the header of the source file is the default. If you select the Get from file header option, then the first row in the file is processed as column names. If you deselect the option, then the first row is processed as data. To specify column names manually, enter a name for each target column in the Mapping section. You can also select a data type for the column.

The Rows to skip field specifies how many rows to skip when loading the source data into the target. If you have selected the Get from file header option, and if you enter a number greater than 0 in the Rows to skip field, then that number of rows after the first row are not loaded into the target. If you have deselected the Get from file header option, and if you enter a number greater than 0 in the Rows to skip field, that number of rows including the first row are not loaded into the target.

To change the character set encoding for the contents of the file, select a value from the Encoding drop-down list.

To specify the characters that enclose text, select the double-quotes or single-quote character or None from the Text enclosure drop-down list.

To change the delimiter character that separates columns in the source, expand the Field delimiter drop-down list and select a character. For example, if the file has columns delimited by semicolons, change the delimiter from the default comma delimiter to a semicolon.
To convert any invalid value in a numeric source column to a null value in the target column, select the Numeric column **Convert invalid data to null** option.

**Specify Mappings**

If you select the **Create Table** or **Drop Table and Create New Table** option and you are getting the source column names from the file header, then in the **Mapping** section either accept the default values for the target columns and data types or specify different values. To specify different values, in the target column, enter a name for the column. In the Data Type column, select a data type from the drop-down list. If you are not getting the source column names from the file header, then for each source column specify a name for the target column and select a data type from the Data Type drop-down list. For the Date data type, select a date format from the Format drop-down list.

For the **Merge into Table** option, for each source column, select a target column from the drop-down list. You must specify at least one column as a key column. To specify a column as a key column, select the **Merge Key** check box for the column. Merge keys are one or more columns that uniquely identify each row in the table. Merge keys must not contain any null values. For loading tables with primary keys, this option automatically enables the selection of primary key columns as the merge keys.

For the **Insert into Table** or **Replace Data** options, for each source column, select a target column from the drop-down list of existing columns.

**Preview Source Data**

To view a selection of data in the source file, select the **File** tab. The source preview has a **Preview size** field and a portion of the data in the file. To change the number of rows displayed, you can enter a value from 1 to 100 in the field.

Any modifications you make in the source preview do not affect the loading of data from the file.

**Preview Target Data**

For all options except **Create Table**, to view the existing data in the target table, select the **Target Preview** tab. The target preview displays the data in the target table before you run the data load job.

To close the settings pane, click **Close**.

**Run the Data Load Job**

When you have added all of the sources for the job and specified the settings for each source, to run the job click the **Start** icon in the Data Load Cart menu bar. In the Run Data Load Job dialog box, click **Run**. To stop the data load job, click the **Stop** icon.

When the data load job completes, the Local Files page displays the results of the job. At the top of the page, the **Status** shows the number of items for which the load has completed over the number of items in the job, and the total time elapsed for the job.

To view information about an item in the job, click the **Settings** icon in the item. In the settings pane, the **Settings** tabs are the same as before running the job, except that the target preview now contains the data loaded by the data load job and a **Data Definition** tab. To close the pane, click **Close**.

To view a log of the load operation, click the **Logging** icon. You can save the log, clear it, or refresh it. Click **OK** to dismiss the log.
The list of tables on the Data Load / Explore page contains any new tables created. The target tables for the Insert into Table, Replace Data, Drop Table and Create New Table, and Merge into Table options contain the loaded data.

Click the Explore Catalog button on the Local files page to view the new or updated table on the Catalog page.

**Fixing a data load job**

After your data load job, you might see errors that you want to correct, or upon inspection, realize that you wanted to name a column differently. In such cases, click the Reload Cart option to reload cards from your recent cart and edit them as you did before your first attempt. Click the pencil icon to make any changes to the data load job (i.e., change a column name).

Click Done to return to the Database Actions page.

### Loading Data from Other Databases

To load data from tables in another database into your Oracle Autonomous Database, on the Data Load page, select LOAD DATA and DATABASE, then click Next. Select a database link from the drop-down list. Drag one or more tables from the list of database tables and drop them in the Data Load Cart.

Each table appears as an item in the Data Load Cart. The item shows the name of the table and the number of rows in it, and the name of the table that is the target for the data load.

To remove a table from the Data Load Cart, click the Remove (trash can) icon for the item. To remove all tables from the cart, click the Remove All (trash can) icon in the Data Load Cart menu bar.

To add a remote database to the list of database links, create a database link to the remote database. For information on creating a database link, see Database Links in Oracle® Database Database Administrator’s Guide 21c.

The databases available to you appear in the drop-down list of the database navigation pane of the Load Tables page.

You can filter the tables displayed in the navigation pane by entering a case-sensitive value in the search field at the top of the navigation tree and pressing Enter. To display all of the tables again, clear the search field and press Enter.

You can add any number of tables from the navigation pane to the Data Load Cart and load data from all of them in a single data loading job. You can set filters on the data for a table to load only the specified data.

**Specify Processing Options**

To specify settings for the data load job, preview the data in the source or the target, and see statistics about the data, click the Settings (pencil) icon for the item in the Data Load Cart.

In the settings pane, on the Settings tab, you can view the name and size of the file in the title of the Load Data dialog box.

The Table field specifies the name of the target table. The value in the field varies depending on the selection in the Options field. If the option is Create Table, then the default target value is the name of source table. To specify a different name for the
target, enter it in the Name field. For the other options, the default value is <None>. Expand the drop-down list and select a table as the target.

In the Options field for the source, select Create Table, Insert into Table, Replace Data, Drop Table and Create New Table, or Merge into Table. Point to the question mark icon to see a brief description of the selected option.

If you select Create Table, then in the Name field accept the default name, which is the name of the source table, or enter a different name.

If you select one of the other options, then expand the drop-down list of the Name field and select a table as the target.

You can set filters to load only the specified data from the source table. To set an initial filter condition on a table, do the following:

1. Click the Edit Source Filter icon.
2. In the Edit Source Filter dialog box, click Add Filter Condition.
3. In the Add Filter Condition dialog box, select a source column and an operator and specify a value.
4. Click Save.

To add another filter condition, click the Edit Source Filter icon and repeat the steps for adding a filter. The filter then has both filter conditions and the AND operator. You can change the AND to an OR by clicking the Edit Source Filter icon and manually replacing the AND with OR in the Filter field.

To remove a filter, in the Edit Source Filter dialog box, delete the value in the Filter field of the Edit Source Filter dialog box.

Specify Mappings

If you select the Create Table or the Drop Table and Create New Table option, then in the Mapping section either accept the default values for the target columns or specify different values. For the target column, enter a name for the column.

For the Insert into Table or Replace Data options, select a target column from the drop-down list of existing columns.

For the Merge into Table option, for each source column, select a target column form the drop-down list. You must specify at least one column as a key column. To specify a column as a key column, select the Merge Key check box for the column. Merge keys are one or more columns that uniquely identify each row in the table. Merge keys must not contain any null values. For loading tables with primary keys, this option automatically enables the selection of primary key columns as the merge keys.

Preview Source Data

To view the data in the source table, in the settings pane select the Source Table tab. The source preview displays the data in the table.

View Statistics

To view statistics about the source table, in the settings pane select the Source Statistics tab. It may take a moment for the statistics to appear. The statistics include the size of the table, the number of rows and columns, the column names, data types, number of distinct values, and other information. Below the details about the columns is a bar graph that displays the top unique values for the selected column.
Preview Target Data

For all options except Create Table, to view the existing data in the target table, in the settings pane select the Target Table tab. The target preview displays the data in the table before you run the data load job.

Run the Data Load Job

When you have added all of the source tables for the job and specified the settings for each table, to run the job click the Start icon in the Data Load Cart menu bar. In the Run Data Load Job dialog box, click Run. To stop the data load job, click the Stop icon.

At the top of the page, the Status shows the number of items for which the load has completed over the number of items in the job, and the total time elapsed for the job. When the data load job completes, the Load Tables page displays the results of the job.

To view information about an item in the job, click the Settings icon in the item. The settings pane has the same Settings, Source Table, Source Statistics, Target Table, SQL and Errors tabs as the settings pane before running the job, except that the target preview now contains the data loaded by the data load job. To close the settings pane, click Close.

To view a log of the load operation, click the Log icon. You can save the log, clear it, or refresh it. Click OK to dismiss the log.

The list of tables on the Data Load / Explore page contains any new tables created. The target tables for the Insert into Table, Replace Data, Drop Table and Create New Table, and Merge into Table options contain the loaded data.

Click the Explore Catalog button on the Local files page to view the new or updated table on the Catalog page.

Fixing a data load job

After your data load job, you might see errors that you want to correct, or upon inspection, realize that you wanted to name a column differently. In such cases, click the Reload Cart option to reload cards from your recent cart and edit them as you did before your first attempt. Click the pencil icon to make any changes to the data load job (i.e., change a column name).

Click Done to return to the Database Actions page.

Loading Data from Cloud Storage

You can load data from a cloud store to a table in your Autonomous Database.

You can load files in these file formats: AVRO, CSV, JSON, ORC, Delimited TXT, GZ, GNU ZIP and Tab-Separated Values. For information on supported file formats, see Format Specifications for JSON and AVRO Files.

You can set filters on the data for a table to load only the specified data. For example, to limit the files to only those that are CSV files, enter *.CSV in the file extension filter.

Configure and run a data load job from the Load Cloud Object page. To open that page:
1. Open the Database Actions Data Load page.

2. Select **LOAD DATA** and **CLOUD STORAGE**, and then click **Next**.

On the left side of the page is a **navigator pane**, where you choose a cloud store connection and the folders or files containing the data. On the right of the page is the data load **cart**, where you stage the files and folders for the data load job. You can set options for the data load job before running it. The Autonomous Database comes with predefined CPU/IO shares assigned to different consumer groups. You can set the consumer group to either low, medium or high while executing a data load job depending on your workload.

To load files from a cloud store into your database, do the following:

- Manage Cloud Storage Links for Data Load Jobs
- Prepare the Data Load Job
- Add Files or Folders for the Data Load Job
- Enter Details for the Data Load Job
- Run the Data Load Job
- View Details About the Data Load Job After It Is Run
- View the Table Resulting from the Data Load Job

### Manage Cloud Storage Links for Data Load Jobs

Before you can load data from a cloud store, you must establish a connection to the cloud store you want to use.

On the Load Cloud Object page:

1. Click **Add Cloud Storage** at the top of the navigation pane.
2. Enter your information in the **Add Cloud Storage** pane. See **Managing Cloud Storage Connections**.
3. Click **Test** to test the connection and then click **Create** to create the link and close the pane.

Alternatively,

- On the **Data Load** page, select **CLOUD LOCATIONS**, and then click **Next** to go to the Manage Cloud page.
- On the Link Cloud Object page, click the **Manage Cloud Store** button at the top of the page to go to the Manage Cloud page.

See **Managing Cloud Storage Connections**.

To return to the Load Cloud Object page, click **Data Load** in the breadcrumbs at the top of the page and then navigate back to the page.

### Prepare the Data Load Job

As you'll see below in **Enter Details for the Data Load Job**, the first decision you'll make when configuring a data load job is how to load the source data into a new or existing table in the database. The choices are:

- Create a table and insert data loaded from the source into the new table.
• Insert data loaded from the source into an existing table.
• Delete all data in an existing table and then insert new data from the source into the table.
• Drop a table, create a new table, and then insert data loaded from the source into the new table.
• Merge data from the source into a table by updating existing rows in the table and inserting new rows into the table.

You may have to adjust your source data or your target table so that the source data loads correctly into the external target table. The number, order, and data types of columns in the source must match those in the target. Consider:

• If you're creating a new table or if the columns in your source exactly match the columns in an existing target, you don't have to do any special preparation.
• If the columns in your source don't match the columns in an existing target, you must edit your source files or target table so they do match.
• If you're loading multiple files, you must make sure that:
  – All the source files are of the same type, for example, CSV, JSON, etc.
  – The number, order, and data types of the columns in all the source files match (and that they match the target, if you're loading into an existing table).

• If you want to partition by date:
  – The source file must contain data where the data type is date or timestamp.
  – You must load a folder containing two or more data sources.
  – The names of the files in the folder must indicate a date or dates, for example, MAR-1999.csv or 2017-04-21.xlsx.

Add Files or Folders for the Data Load Job

Add files from the cloud store to the data load cart, where you can edit the details of the data load job. To add the files:

1. From the list at the top of the navigator pane on the left, select the bucket with your source data.
   The list shows links that were established on the Manage Cloud Storage page. If you haven't yet registered the cloud store you want to use, click the Manage Cloud Store button at the top of the page and register a connection. See Manage Cloud Storage Links for Data Load Jobs, above.

2. Drag one or more items from the file navigator on the left and drop them into the cart on the right.
   • You can add files, folders, or both. A card is added to the cart for each file or folder you drag into it. The card lists the name of the source file or folder and a proposed name for the target table.
   • If you add a folder that contains multiple files, all the files must be of the same type, that is, CSV, TXT, etc.
     When you add the folder to the cart, a prompt is displayed that asks if you want to load all the objects from the multiple source files into a single target table. Click OK to continue or Escape to cancel.
• When you add multiple individual files or multiple folders to the cart, the data represented by each card will be loaded into a separate table, but all the items in the cart will be processed as part of the same data load job.

• You can add files or folders from a different bucket, but if you do that, you’re prompted to remove all files that are already in the cart before proceeding. To select files from a different bucket, select the bucket from the drop-down list in the navigator pane on the left and then add the file(s), as described above.

• You can drop files or folders into the data load cart and then navigate away from the Data Load Object page. When you return to the page, those items remain on the page, but you may receive a message, “Remove All Data Load Items. Changing to another Cloud storage location requires all items to be removed from the data load job. Do you wish to continue?” Click Yes to remove the items from the cart. Click No to keep the items in the cart. Then you can continue to work.

You can remove items from the cart before running the data load job:

• To remove an item from the cart, click Remove on the card for the item.

• To remove all items from the cart, click Remove All in the data link cart menu bar at the top of the pane.

Enter Details for the Data Load Job

Enter the details about the data load job in the Load Data from Cloud Storage pane.

On the card in the data link cart, click Settings to open the Load Data from Cloud Storage pane for that job. The pane contains:

• Settings Tab - Table Section
• Settings Tab - Properties Section
• Settings Tab - Mapping Section
• File Tab
• Table Tab
• Error Tab
• Close Button - Save and Close the Pane

Settings Tab - Table Section

Set details about the target table in the Table section.

• Option: Select an item from the Option list to specify how the data should be loaded into a new or existing table. The processing options are:
  – Create Table: Creates a table and inserts the data into the new table. When you select this option, the Name field on the Settings tab is filled with a default name, based on the name of the source file or folder. You can change it if you want.
  – Insert into Table: Inserts data loaded from the source into an existing table. When you select this option, the Name field on the Settings tab presents a list of the tables in the current schema. Select the table into which you want to insert the data.
Replace Data: Deletes all data in the existing table and then inserts new data from the source into the table. When you select this option, the Name field on the Settings tab presents a list of the tables in the current schema. Select the table you want to use.

Drop Table and Create New Table: Drops the table (if it already exists), creates a new table, and then inserts the new data into the table. When you select this option, the Name field on the Settings tab presents a list of the tables in the current schema. Select the table you want to use.

Merge into Table: Updates existing rows and inserts new rows in the table. When you select this option, the Name field on the Settings tab presents a list of the tables in the current schema. Select the table you want to use.

- Name: The name of the target table.
- Partition Column:

List Partitions and Date-based partitions are the different types of partitions available in data loading.

List partitioning is required when you specifically want to map rows to partitions based on discrete values.

To partition according to a specific column, click the Partition Column drop-down list and select the column you want to use for the partitioning.

You will have N files per partition value, all partitioned by the partition column you select.

**Note:**

- For linked files (from external tables) there is also a requirement that for each file, the list partitioning column can contain only a single distinct value across all of the rows.
- If a file is list partitioned, the partitioning key can only consist of a single column of the table.

Date-based partitioning is available when you load a folder containing two or more data sources that contain date or timestamp data.

To partition according to date, click the Partition Column drop-down list and select the DATE or TIMESTAMP column you want to use for the partitioning.

**Settings Tab - Properties Section**

Specify options to control how the source data is interpreted, previewed, and processed. These options vary, depending on the type of source data.

- **Encoding:** Select a character encoding type from the list. This option is available when the loaded file is in plain text format (CSV, TSV, or TXT). The default encoding type is UTF-8.

- **Text enclosure:** Select the character for enclosing text: " (double-quote character), ' (single-quote character) or None. This option is visible only when the selected file is in plain text format (CSV, TSV, or TXT).
• **Field delimiter:** Select the delimiter character used to separate columns in the source. For example, if the source file uses semicolons to delimit the columns, select **Semicolon** from this list. The default is **Comma**. This option is visible only when the selected file is in plain text format (CSV, TSV, or TXT).

• **Rows to Skip:** Specifies the number of rows to skip when loading the source data into the target:
  - If you select the **Get from file header** option under **Source column name** (see below) and if you enter a number greater than 0 in the **Rows to skip** field, then that number of rows after the first row are not loaded into the target.
  - If you deselect the **Get from file header** option under **Source column name**, and if you enter a number greater than 0 in the **Rows to skip** field, then that number of rows including the first row are not loaded into the target.

• **Source column name:** Select the **Get from file header** checkbox to use the column names form the source table in the target table.
  - If you select this option, the first row in the file is processed as column names. The rows in the **Mapping** section, below, are filled with those names (and with the existing data types, unless you change them).
  - If you deselect this option, the first row is processed as data. To specify column names manually, enter a name for each target column in the **Mapping** section. (You will also have to enter data types.)

• **Numeric column:** Select the **Convert invalid data to null** checkbox to convert an invalid numeric column value into a null value.

### Settings Tab - Mapping Section

The settings in the **Mapping** section control how data from the source files are loaded into the rows of the target database table. For each row, the data from the column listed under **Source column** will be loaded into the column listed under **Target column**.

As mentioned above, the contents of the **Mapping** table change according to what processing option you chose in the **Table** section and which properties you set in the **Properties** section.

• **Source column:** Lists the columns from the source file.
  - If the **Get from file header** option under **Properties** is selected, **Source column** shows the names of the columns in the source file. If the **Get from file header** option is not selected, generic names like **COLUMN_1, COLUMN_2**, etc., are used. This field is always read only.

• **Target column:** Lists the columns in the target table. Accept, select, or enter a column in the target table.
  - The contents of this column differ, depending on what you selected for the table processing **Option** and whether you selected for the **Get from file header** option.
    - If (1) the processing option is **Create Table** or **Drop Table and Create New Table** and (2) the **Get from file header** option is selected, then the **Target column** uses the names of the columns in the source file. You can change the name of a target column by replacing the provided name with a new one.
    - If (1) the processing option is **Create Table** or **Drop Table and Create New Table** and (2) the **Get from file header** option is not selected, then generic names like **COLUMN_1, COLUMN_2**, etc., are used. You can change the name of a target column by replacing the provided name with a new one.
– If (1) the processing option is **Insert into Table**, **Replace Data**, or **Merge Into Table** and (2) the **Get from file header** option is selected, then the **Target column** has a drop-down list of all the columns in the target table, with their data types. By default, the column with the name corresponding to the source column is selected, but you can select a different one from the list.

– If (1) the processing option is **Insert into Table**, **Replace Data**, or **Merge Into Table** and (2) the **Get from file header** option is not selected, then the **Target column** has a drop-down list of all the columns in the target table, with their data types. Select a column from the list to use as the target column.

**Note:**

If you’re loading multiple files from a folder in a single data load job, only the first file will be shown in the **Mapping** section. However, as long as the column names and data types match, the data from all source files will be loaded.

- **Data Type:** Lists the data type to use for data in that column. This column is displayed only for **Create Table** or **Drop Table and Create New Table**. The contents change depending on whether the **Get from file header** option is selected.
  - If the **Get from file header** option is selected, **Data type** shows the data types of the columns in the source file (for **Create Table**) or in the existing table (for **Drop Table and Create New Table**). If you want to change the data type for the target, click the name and select a different one from the list.
  - If the **Get from file header** option is not selected, **Data type** shows all available data types. Select the data type to use for the target column from the list.

- **Length/Precision** (Optional): For columns where the **Data Type** is **NUMBER**, enter the length/precision for the numbers in the column. Precision is the number of significant digits in a number. Precision can range from 1 to 38.

  For columns where Data Type is VARCHAR2, the **Auto** value in **Length/Precision** field enables the Auto Size feature.

  With the Auto-Size column Width feature, you can automatically size any column to fit the largest value in the column. Select **Auto** from the **Length/Precision** drop-down values or pick a value from the drop-down list.

- **Scale** (Optional): For columns where the **Data Type** is **NUMBER**, enter the scale for the numbers in the column. Scale is the number of digits to the right (positive) or left (negative) of the decimal point. Scale can range from ranges from -84 to 127.

- **Format:** If the data type in the **Data type** column is **DATE** or one of the **TIMESTAMP** types, select a format for that type from the from the **Format** drop-down list.

- **Merge Key:** This option is used only for the processing option **Merge into Table**.

  For the **Merge into Table** option, you must specify at least one column to use as a key column. Merge keys are one or more columns that uniquely identify each row in the table. To specify a key column, select the **Merge Key** checkbox for the column. Merge keys must not contain any null values. For loading tables with
primary keys, this option automatically enables the selection of primary key columns as the merge keys.

File Tab
The File tab displays the source data in tabular form. The display reflects the settings you chose in the Properties section.

If you dragged a folder containing multiple files into the data load cart and then clicked Settings for that card, the File pane includes a Preview Object (File) drop-down list at the top of the pane that lists all the files in the folder. Select the source file you want to preview from that list.

Table Tab
The Table tab displays what the target table is expected to look like after the data has been loaded. If you chose the Create Table processing option, no table is shown.

Errors Tab
The Error tab lists any errors generated when attempting to run the data load job.

SQL Tab
The SQL tab displays the SQL commands that will be run to complete this data load job.

Note:
You can see the SQL code even before the table is created.

Close Button - Save and Close the Pane
After entering all the details for the data load job, click Close at the bottom of the page. This saves the details you entered and returns you to the Load Data from Cloud Storage pane. To close the page without saving your entries, press Escape.

Run the Data Load Job
Once you've added data sources to the data load cart and entered details about the data load job, you can run the job.

To run the job:
1. If you haven’t already done so, click the Close button in the Load Data from Cloud Storage pane to save your settings and close the pane. If any of the settings are invalid, an error message reports the problem. Fix the problem and click Close.

2. Click Start in the data load cart menu bar. To stop the data load job, click Stop.

When the data load job completes, the Load Cloud Object page displays the results of the job. At the top of the page, a Status message shows the number of items for which the load has completed over the number of items in the job and the total time elapsed for the job.
View Details About the Data Load Job After It Is Run

To view details about the data load job after it is run, click **Settings** in the card for the item. The **Load Data from Cloud Storage** pane is displayed again, with the settings used for the job plus some additional details about job run.

**Settings Tab**

The **Settings** tab shows the details that were set in the **Settings** tab when preparing for job.

**File Tab**

The **File** tab shows the source file or files used for the data load job.

**Table Tab**

The **Table** tab shows the table created or modified from the data load job.

**Error Tab**

The **Error** tab shows rows rejected from the data link job, if any. You can click the icon at the top of the page to download the rejected rows as a CSV file.

**SQL Tab**

The **SQL** tab shows the SQL that was created and run to load the data. You can click the **Copy to clipboard** button at the top of the pane to copy it.

**Data Definition Tab**

The **Data Definition** tab shows the data definition of the table created from the data load job.

View the Table Resulting from the Data Load Job

To view the new tables or tables modified by the data load job, you can:

1. Fix your data load job. After your data load job, you might see errors that you want to correct, or upon inspection, realize that you wanted to name a column differently. In such cases, click the **Reload Cart** option to reload cards from your recent cart and edit them as you did before your first attempt. Click the pencil icon to make any changes to the data load job (i.e., change a column name).

2. On the Load Cloud Object page, click the **Explore Catalog** button.

3. On the Catalog page, find the new or updated table.

4. Click **View Details** on the right side of the card and review the table. See The Catalog Page.

Alternatively, you can:
1. From the Database Actions Data Load page, click **Explore**.

2. On the Explore page, click the name of the table you want to review. See **Exploring Data**.

## Linking Data

You can link to data in remote databases or in cloud storage buckets.

When you link to data in a remote database or in cloud storage, the target object produced is an external table or a view. When you select that target table or view on the Data Load - Explore page, the source preview for the object shows the current data in the source object.

Linking to columns in a remote database or to files in cloud storage can be useful when the source data is being continually updated. For example, if you have a database table with columns that are updated for each new sales transaction, and you have run a data load job that links columns of the source table to targets in your Oracle Autonomous Database, then those targets have the new sales data as it is updated in the source.

See:

- Linking to Other Databases
- Linking to Objects in Cloud Storage

## Linking to Other Databases

To link to data in tables in another database from your Oracle Autonomous Database, on the Data Load page, select **LINK DATA** and **DATABASE**, then click **Next**. Select a database from the drop-down list. Drag one or more tables from the list of database tables and drop them in the Data Load Cart.

Each table appears as an item in the Data Load Cart. The item shows the name of the table and the number of rows in it, and the name of the table that is the target for the data load.

To remove a table from the Data Load Cart, click the Remove (trash can) icon for the item. To remove all tables from the cart, click the Remove All (trash can) icon in the Data Load Cart menu bar.

To add a remote database to the list of databases, create a database link to the remote database. For information on creating a database link, see Database Links in *Oracle® Database Database Administrator’s Guide*.

The databases available to you appear in the drop-down list of the database navigation pane of the Link Tables page.

You can filter the tables displayed in the navigation pane by entering a case-sensitive value in the search field at the top of the navigation tree and pressing Enter. To display all of the tables again, clear the search field and press Enter.

You can add any number of tables from the navigation pane to the Data Load Cart and create links to each of them in a single data loading job. You can set filters on the data for a table to load only the specified data. When you run the data load job, a view to each table is created in your Oracle Autonomous Database.

### Specify the Target, Create Filters, and View Mappings

To specify settings for the data load job, preview the data in the source or the target, and see statistics about the data, click the Settings (pencil) icon for the item in the Data Load Cart.
In the settings pane, on the Settings tab, the **Source** field displays the name of the table and the number of rows in the table.

The **Target** field specifies the name of the target view. To specify a different name for the target, enter it in the **Target** field.

You can set filters to load only the specified data from the source table. To set an initial filter condition on a table, do the following:

1. Click the Edit Source Filter icon.
2. In the Edit Source Filter dialog box, click **Add Filter Condition**.
3. In the Add Filter Condition dialog box, select a source column and an operator and specify a value.
4. Click **Save**.

To add another filter condition, click the Edit Source Filter icon and repeat the steps for adding a filter. The filter then has both filter conditions and the **AND** operator. You can change the **AND** to an **OR** by clicking the Edit Source Filter icon and manually replacing the **AND** with **OR** in the filter field.

To remove a filter, in the Edit Source Filter dialog box, delete the value in the Filter field of the Edit Source Filter dialog box.

The **Mapping** section displays the columns in the source table and those in the target view.

**Preview Source Data**

To view the data in the source table, in the settings pane select the **Source Preview** tab. The source preview displays the data in the table.

**View Statistics**

To view statistics about the source table, in the settings pane select the **Statistics** tab. It may take a moment for the statistics to appear. The statistics include the size of the table, the number of rows and columns, the column names, data types, number of distinct values, and other information. Below the details about the columns is a bar graph that displays the top unique values for the selected column.

**Preview Target Data**

To view the data in the target view, in the settings pane select the **Target Preview** tab. The target preview displays the data in the target view. If the view does not yet exist, then the target data is the same as the source data, regardless of any filters set for the data load job.

**Run the Data Load Job**

When you have added all of the source tables for the job and specified the settings for each table, to run the job click the Start icon in the Data Load Cart menu bar. In the Run Data Load Job dialog box, click **Run**. To stop the data load job, click the Stop icon.

At the top of the page, the **Status** shows the number of items for which the load has completed over the number of items in the job, and the total time elapsed for the job. When the data load job completes, the Link Tables page displays the results of the job.
To view information about an item in the job, click the Settings (circled i) icon in the item. The settings pane has the same **Settings, Source Preview, Statistics,** and **Target Preview** tabs as the settings pane before running the job, except that the **Target Preview** now contains the data loaded by the data load job. To close the settings pane, click **Close.**

To view a log of the load operation, click the Logging icon. You can save the log, clear it, or refresh it. Click **OK** to dismiss the log.

The list of views on the Data Load / Explore page contains any new views created. A preexisting view that was the target for the data load job now contains the loaded data.

On the Database links page, click the **Reload Cart** button. Clicking the button enables you to add the data sources to the data load the cart again.

Click **Done** to return to the Database Actions page.

### Linking to Objects in Cloud Storage

When you create a link to files in a cloud store bucket from your Oracle Autonomous database, you create an external table that links to the files in the cloud store.

You can link to files in these file formats: AVRO, CSV, JSON, Parquet, ORC, Delimited TXT. For information on supported file formats, see Format Specifications for JSON, AVRO, and XML Files.

Configure and run a data link job from the Link Cloud Object page. To open that page:

1. Open the Database Actions **Data Load page.**
2. Select **LINK DATA** and **CLOUD STORAGE,** and then click **Next.**

On the left side of the page is a **navigator pane,** where you choose a cloud store connection and the folders or files containing the data. On the right of the page is the data load **cart,** where you stage the files and folders for the data link job. You can set options for the data link job before running it. The Autonomous Database comes with predefined CPU/IO shares assigned to different consumer groups. You can set the consumer group to either low, medium or high while executing a data load job depending on your workload.

To link to files from a cloud store, do the following:

- **Manage Cloud Storage Links for Data Link Jobs**
- **Prepare the Data Link Job**
- **Add Files or Folders for the Data Link Job**
- **Enter Details for the Data Link Job**
- **Run the Data Link Job**
- **View Details About the Data Link Job After It Is Run**
- **View the Table Resulting from the Data Link Job**

### Manage Cloud Storage Links for Data Link Jobs

Before you can link to data in a cloud store, you must establish a connection to the cloud store you want to use.

On the Link Cloud Object page:

1. Click **Add Cloud Storage** at the top of the navigation pane.
2. Enter your information in the Add Cloud Storage pane. See Managing Cloud Storage Connections.

3. Click Test to test the connection and then click Create to create the link and close the pane.

Alternatively,

- On the Data Load page, select CLOUD LOCATIONS, and then click Next to go to the Manage Cloud page.
- On the Load Cloud Object page, click the Manage Cloud Store button at the top of the page to go to the Manage Cloud page.

See Managing Cloud Storage Connections.

To return to the Link Cloud Object page, click Data Load in the breadcrumbs at the top of the page and then navigate back to the page.

Prepare the Data Link Job

You may have to adjust your source data or your target table so that the source data links correctly to the external target table. Consider:

- If you’re linking to multiple files, you must make sure that:
  - All the source files are of the same type, for example, CSV, JSON, etc.
  - The number, order, and data types of the columns in all the source files match.
- If you want to partition by date:
  - The source file must contain data where the data type is date or timestamp.
  - You must load a folder containing two or more data sources.
  - The names of the files in the folder must indicate a date or dates, for example, MAR-1999.csv or 2017-04-21.xlsx.

Add Files or Folders for the Data Link Job

Add files from the cloud store to the data link cart, where you can edit the details of the data link job. To add the files:

1. From the list at the top of the navigator pane on the left, select the bucket with your source data.

   The list shows links that were established on the Manage Cloud Storage page. If you haven't yet registered the cloud store you want to use, click the Manage Cloud Store button at the top of the page and register a connection. See Manage Cloud Storage Links for Data Link Jobs, above.

2. Drag one or more items from the file navigator on the left and drop them into the cart on the right.

   - You can add files, folders, or both. A card is added to the cart for each file or folder you drag into it. The card lists the name of the source file or folder and a proposed name for the target table.
   - If you add a folder that contains multiple files, all the files must be of the same type, that is, CSV, TXT, etc.
When you add the folder to the cart, a prompt is displayed that asks if you want to load all the objects from the multiple source files into a single target table. Click **OK** to continue or **Escape** to cancel.

- When you add multiple individual files or multiple folders to the cart, the data represented by each card will be loaded into a separate table, but all the items in the cart will be processed as part of the same data load job.
- You can add files or folders from a different bucket, but if you do that, you’re prompted to remove all files that are already in the cart before proceeding. To select files from a different bucket, select the bucket from the drop-down list in the navigator pane on the left and then add the file(s), as described above.
- You can drop files or folders into the data load cart and then navigate away from the Data Link Object page. When you return to the page, those items remain on the page, but you may receive a message, “Remove All Data Link Items. Changing to another Cloud storage location requires all items to be removed from the data load job. Do you wish to continue?” Click **Yes** to remove the items from the cart. Click **No** to keep the items in the cart. Then you can continue to work.

You can remove items from the cart before running the data link job:

- To remove an item from the cart, click **Remove** on the card for the item.
- To remove all items from the cart, click **Remove All** in the data link cart menu bar at the top of the pane.

**Enter Details for the Data Link Job**

Enter the details about the data link job in the Link Data from Cloud Storage pane.

On the card in the data link cart, click **Settings** to open the Link Data from Cloud Storage pane for that job. The pane contains:

- **Settings Tab - Table Section**
- **Settings Tab - Properties Section**
- **Settings Tab - Mapping Section**
- **File Tab**
- **Table Tab**
- **Error Tab**
- **SQL Tab**
- **Close Button - Save and Close the Pane**

**Settings Tab - Table Section**

Set details about the target table in the **Table** section.

- **Name:** The name of the target table.
- **Partition Column:**
  
  List Partitions and Date-based partitions are the different types of partitions available in data linking.
List partitioning is required when you specifically want to map rows to partitions based on discrete values.

To partition according to a specific column, click the **Partition Column** drop-down list and select the column you want to use for the partitioning.

You will have N files per partition value, all partitioned by the partition column you select.

### Note:
- For linked files (from external tables) there is also a requirement that for each file, the list partitioning column can contain only a single distinct value across all of the rows.
- If a file is list partitioned, the partitioning key can only consist of a single column of the table.

Date-based partitioning is available when you link a folder containing two or more data sources that have columns that contain date or timestamp data.

To partition according to date, click the **Partition Column** drop-down list and select the **DATE** or **TIMESTAMP** column you want to use for the partitioning.

- **Validation Type:** Validation examines the source files, optional partitioning information, and report rows that do not match the format options specified. Select **None** for no validation; select **Sample** to perform validation based on a sample of the data; or select **Full** to perform validation based on all the data.
- **Use Wildcard:** This check box enables use of wildcard characters in search condition to retrieve specific group of files that matches the filter criteria. You can use a wildcard character, such as an asterisk (*) that searches, filters, and specifies groups of files that detect and add new files to the external table.

For example, if you enter file*, then file01, file02, file03, and so on are considered to match the keyword. The asterisk (*) matches zero or more characters of the possibilities, to the keyword.

### Note:
The wildcard support is incompatible with partitioning. The validation of source file fails if you use wildcards with partitioned data.

### Settings Tab - Properties Section
Specify options to control how the source data is interpreted, previewed, and processed. These options vary, depending on the type of source data.

- **Encoding:** Select a character encoding type from the list. This option is available when the linked file is in plain text format (CSV, TSV, or TXT). The default encoding type is UTF-8.
• **Text enclosure:** Select the character for enclosing text: " (double-quote character), ' (single-quote character) or None. This option is visible only when the selected file is in plain text format (CSV, TSV, or TXT).

• **Field delimiter:** Select the delimiter character used to separate columns in the source. For example, if the source file uses semicolons to delimit the columns, select **Semicolon** from this list. The default is **Comma**. This option is visible only when the selected file is in plain text format (CSV, TSV, or TXT).

• **Rows to Skip:** Specifies the number of rows to skip when linking the source data to the target external table:
  - If you select the Get from file header option under **Source column name** (see below) and if you enter a number greater than 0 in the Rows to skip field, then that number of rows after the first row are not linked to the target.
  - If you deselect the Get from file header option under **Source column name**, and if you enter a number greater than 0 in the Rows to skip field, then that number of rows including the first row are not linked to the target.

• **Source column name:** Select the Get from file header checkbox to use the column names form the source table in the target table.
  - If you select this option, the first row in the file is processed as column names. The rows in the Mapping section, below, are filled with those names (and with the existing data types, unless you change them).
  - If you deselect this option, the first row is processed as data. To specify column names manually, enter a name for each target column in the Mapping section. (You will also have to enter data types.)

• **Numeric column:** Select the Convert invalid data to null checkbox to convert an invalid numeric column value into a null value.

### Settings Tab - Mapping Section

The settings in the **Mapping** section control how data from the source files are linked to the rows of the target external table. For each row, the data from the column listed under **Source column** will be linked to the column listed under **Target column**.

• **Source column:** Lists the columns from the source file.

  If the Get from file header option under **Properties** is selected, **Source column** shows the names of the columns in the source file. If the Get from file header option is not selected, generic names like COLUMN_1, COLUMN_2, etc., are used. This field is always read only.

• **Target column:** Lists the columns in the target table.
  - If the Get from file header option is selected, then the **Target column** uses the names of the columns in the source file. You can change the name of a target column by replacing the provided name with a new one.
  - If the Get from file header option is not selected, then generic names like COLUMN_1, COLUMN_2, etc., are used. You can change the name of a target column by replacing the provided name with a new one.
**Note:**

If you’re linking multiple files from a folder in a single data link job, only the first file will be shown in the **Mapping** section. However, as long as the column names and data types match, the data from all source files will be linked.

- **Data Type:** Lists the data type to use for data in that column. The contents change depending on whether the **Get from file header** option is selected.
  - If the **Get from file header** option is selected, **Data type** shows the data types of the columns in the source file. If you want to change the data type for the target, click the name and select a different one from the list.
  - If the **Get from file header** option is not selected, **Data type** shows all available data types. Select the data type to use for the target column from the list.

- **Length/Precision** (Optional): For columns where the **Data Type** is **NUMBER**, enter the length/precision for the numbers in the column. Precision is the number of significant digits in a number. Precision can range from 1 to 38.
  
  For columns where Data Type is VARCHAR2, the **Auto** value in **Length/Precision** field enables the **Auto Size** feature.

  With the Auto-Size column Width feature, you can automatically size any column to fit the largest value in the column. Select **Auto** from the **Length/Precision** drop-down values or pick a value from the drop-down list.

- **Scale** (Optional): For columns where the **Data Type** is **NUMBER**, enter the scale for the numbers in the column. Scale is the number of digits to the right (positive) or left (negative) of the decimal point. Scale can range from ranges from -84 to 127.

- **Format:** If the data type in the **Data type** column is **DATE** or one of the **TIMESTAMP** types, select a format for that type from the from the **Format** drop-down list.

**File Tab**

The **File** tab displays the source data in tabular form. The display reflects the settings you chose in the **Properties** section.

If you dragged a folder containing multiple files into the data link cart and then clicked **Settings** for that card, the **File** pane includes a **Preview Object (File)** drop-down list at the top of the pane that lists all the files in the folder. Select the source file you want to preview from that list.

**Table Tab**

The **Table** tab displays what the target table is expected to look like after the data has been linked.

**Error Tab**

The **Error** tab lists any errors generated when attempting to run the data link job.
SQL Tab

The SQL tab displays the SQL commands that will be run to complete this data link job.

Note:

You can see the SQL code even before the table is created.

Close Button - Save and Close the Pane

After entering all the details for the data link job, click Close at the bottom of the page. This saves the details you entered and returns you to the Link Data from Cloud Storage pane. To close the page without saving your entries, press Escape.

Run the Data Link Job

Once you've added data sources to the data link cart and entered details about the data link job, you can run the job.

To run the job:

1. If you haven't already done so, click the Close button in the Link Data from Cloud Storage pane to save your settings and close the pane. If any of the settings are invalid, an error message reports the problem. Fix the problem and click Close.

2. Click Start in the data link cart menu bar. To stop the data link job, click Stop.

When the data link job completes, the Link Cloud Object page displays the results of the job. At the top of the page, a Status message shows the number of items for which the link has completed over the number of items in the job and the total time elapsed for the job.

View Details About the Data Link Job After It Is Run

To view details about the data link job after it is run, click Settings in the card for the item. The Link Data from Cloud Storage pane is displayed again, with the settings used for the job plus some additional details about job run.

Settings Tab

The Settings tab shows the details that were set in the Settings tab when preparing for job.

File Tab

The File tab shows the source file or files used for the data link job.

Table Tab

The Table tab shows the external table created for the data link.
Error Tab
The Error tab shows rows rejected from the data link job, if any. You can click the icon at the top of the page to download the rejected rows as a CSV file.

SQL Tab
The SQL tab shows the SQL that was created and run to link the data. You can click the Copy to clipboard button at the top of the pane to copy it.

View the Table Resulting from the Data Link Job
After running a data link job, the Link Cloud Object page is shown, with an Explore Catalog button on the bottom of the page. To view the table created by the data link job:

1. On the Database links page, click the Reload Cart button. Clicking the button enables you to add the data sources to the data load the cart again.
2. Click the Explore Catalog button.
3. On the Catalog page, find the new or updated table.
4. Click on the right side of the card and review the table. See The Catalog Page.

Alternatively, you can:

1. From the Database Actions Data Link page, click Explore.
2. On the Explore page, click the name of the table you want to review. See Exploring Data.

Feeding Data
You can run a live table feed on demand, on a schedule, or as the result of a notification.

The bucket can contain files in these formats: AVRO, CSV, JSON, Parquet, ORC, Delimited TXT. All of the files must have the same column signature.

About the Live Feed Page
On the Database Actions - Data Load page select FEED DATA and CLOUD STORAGE, then click Next to display the Live Feed page. On this page, you can:

- Manage Cloud Storage Connections for Live Table Feeds
- Create a Live Table Feed Object
- List, Filter, and Sort Live Table Feed Objects
- Find and View Live Table Feed Objects
- Run a Live Table Feed
- Suspend or Resume a Live Table Feed Job
• **Delete a Live Table Feed**

**Manage Cloud Storage Connections for Live Table Feeds**

Before you create a live table feed, you must establish a connection to the cloud store you want to use:

1. Click the **Manage Cloud Store** button at the top of the page to go to the Manage Cloud page. For instructions, see Managing Cloud Storage Connections.

2. To return to the Live Feed page, click **Data Load** in the breadcrumbs at the top of the page and then navigate to the Live Feed page, as described in About the Live Feed Page, above.

**Create a Live Table Feed Object**

To create a live table feed object,

1. On the Live Feed page, click the **Create Live Table Feed** button to display the Create Live Feed pane. Enter information as follows:
   - **Live Table Feed Name**: Accept the default name or enter a different name to identify this live table feed.
   - **Target Table Name**: Accept the default name or enter a different name. This is the name of the target table that will be created in your Autonomous Database instance to store the data from the live feed.
   - **Object Filter (Regular Expression)**: Enter a regular expression to limit the live table feed to only those files in the bucket that match the expression. For example, to limit the files to only those that are CSV files with names that start with SALES, enter `SALES.*.CSV`
   - **Schema**: Select the schema you want to use from the drop-down list.
   - **Cloud Storage**: Select the cloud connection for the bucket containing the file you want to use for feeding data.

   Connections that were established on the Manage Cloud page are listed here. If you haven't registered the cloud store you want to use yet, click the **Manage Cloud Store** button at the top of the page and register a connection to a bucket in a cloud store. See Managing Cloud Storage Connections.

   - **Enable for Notification**: Select this option so that new or changed data in the data source will be loaded based on an Oracle Cloud Infrastructure notification. When you select this option, you can avoid delays that might occur when polling is initiated on a schedule (that is, if you selected the live table feed **Scheduled** option).

   When you select the **Enable for Notification** option, you must also:
   - Configure your object store bucket to emit notifications
   - Create a Notifications service subscription topic
   - Create an Events service rule
   - Copy the notification URL
   - Create a Notifications service subscription
   - Confirm that notifications are allowed

For complete instructions, see Creating a Notification-Based Live Table Feed.
Scheduled: Select this option to set up a schedule for running the live table feed object; that is, to poll the data source on a regular basis:

- In the time interval fields, enter a number, and select a time type and the days on which to poll the bucket for new or changed files. For example, to poll every two hours on Monday, Wednesday, and Friday, enter 2, select Hours, and then select Monday, Wednesday, Friday in the appropriate fields.
- Select a start and end date. If you don't select a start date, the current time and date are used as the start date. The end date is optional. However, without an end date, the live feed will continue to poll.

2. Click Create.

List, Filter, and Sort Live Table Feed Objects

When you open the Live Feed page, existing live table feed objects are displayed as cards on the page. They are identified as LIVE_TABLE_FEED entities.

To filter live table feed objects:

1. Click the Display or hide filter panel (funnel) icon at the top left of the page to display filter options. By default, the live table feed objects from the current user's schema are shown.

2. To include objects from other schemas, select the check boxes next to the names of the schemas you want, under Schemas. To remove a schema from the filter list, deselect the box next to its name.

3. To show objects from all available schemas, select all the schemas in the list or deselect all the schemas in the list.

To sort live table feed objects

1. Click the Sort by button at the top right of the page.

2. Select a sorting option. To sort ascending, click the icon with the up arrow. To sort descending, click the icon with the down arrow.

Find and View Live Table Feed Objects

To search for available live table feed entities in the selected schemas, enter a value in the search field at the top of the page and press Enter. The display then includes only the entities whose names contain the characters in the search field. To clear the search field, click the Clear search results (X) icon in the search field.

To remove a schema or sorting value from the selected filters, deselect the schema or sorting value in the filter panel, or click the Remove filter (X) icon for the schema or sorting value above the display of live table feed objects. To close the filter panel, click the Hide filter panel (X) icon in the panel.

To refresh the display of live table feeds, click the Refresh icon at the top of the page.

Edit a Live Table Feed Object

To edit details of a live table feed object,

1. On the Live Feed page, find the card for the live table feed whose details you want to edit.
2. Click the Actions icon (three dots) on the card and select **Edit Live Table Feed**. You can edit the following options:

- **Object Filter (Regular Expression):** Enter a regular expression to limit the live table feed to only those files in the bucket that match the expression. For example, to limit the files to only those that are CSV files with names that start with SALES, enter `SALES.*.CSV`.

- **Enable for Notification:** Select this option so that new or changed data in the data source will be loaded based on an Oracle Cloud Infrastructure notification. When you select this option, you can prevent any delays that might occur when polling is initiated on a schedule (that is, the live table feed **Scheduled** option).

  When you select the **Enable for Notification** option, you must also:
  - Copy the live table feeds notification URL
  - Configure your cloud store to emit notifications
  - Configure Oracle Cloud Infrastructure to route events to the endpoint used for the live table feed.
  - Create a rule.
  - Create a subscription.
  - Confirm that notifications are allowed at the live feed service.

  For complete instructions, see [Creating a Notification-Based Live Table Feed](#).

- **Scheduled:** Select this option to set up a schedule for running the live table feed object; that is, to poll the data source on a regular bases:

  - In the time interval fields, enter a number, and select a time type and the days on which to poll the bucket for new or changed files. For example, to poll every two hours on Monday, Wednesday, and Friday, enter `2` hours, select **Hours**, and then select **Monday, Wednesday, Friday** in the appropriate fields.

  - Select a start and end date.

3. Click **Save**.

### Run a Live Table Feed

You can run a live table feed on demand, on a schedule, or as the result of a notification.

**To run a live table feed on demand:**

1. On the Live Feed page, find the card for the live table feed you want to run.

2. Click the Actions icon (three dots) on the card and select **Run Live Table Feed Immediately (Once)**.

**To run a live table feed on a schedule:**

You can set a schedule for running live table feeds on the **Create Live Table Feed** pane (when creating a new table feed) or the **Edit Live Table Feed** pane (when editing an existing table feed). See [Create a Live Table Feed Object](#) or [Edit a Live Table Feed Object](#).

**To run a live table feed as the result of a notification:**

See [Creating a Notification-Based Live Table Feed](#).

Select the **Scheduled** check box to display the schedule options and then set the schedule by selecting the options you want.
To view live table feed run details:

1. On the Live Feed page, find the card for the live table feed whose run details you want to see.
2. Click the Actions icon (three dots) on the card and select **Live Table Feed Run Details**.
   The **Objects** tab on the Live Table Feed Run Details pane displays information about the jobs, such as when the run occurred, the objects involved in the run, the rows loaded and the rows rejected, and other details. Click the **All** tab to view more details, such as the event type.

Suspend or Resume a Live Table Feed Job

1. On the Live Feed page, find the card for the live table feed job you want to suspend or resume.
2. Click the Actions icon (three dots) on the card and select **Suspend Live Table Feed Job** to suspend the job or **Resume Live Table Feed Job** to resume it.

Delete a Live Table Feed

1. On the Live Feed page, find the card for the live table feed job you want to delete.
2. Click the Actions icon (three dots) on the card and select **Delete Live Table Feed**.

Creating a Notification-Based Live Table Feed

You can load data through a live table feed based on an Oracle Cloud Infrastructure notification.

In addition to being able to run a live table feed on demand or on a schedule, as described in Feeding Data, you can also run a feed as the result of a notification. When data in the source bucket is changed, a notification is sent which triggers a run of the table feed. With a notification-based live table feed, you can avoid any delay that might come from running on-demand or scheduled live table feed jobs.

**Note:**

Notification-based live table feeds aren’t available on the Oracle Cloud Infrastructure free tier. You must be on a paid tenancy with appropriate permissions on your account to use this feature.

To create a notification-based live table feed:

• Step 1: Configure your object store bucket to emit notifications
• Step 2: Create a Notifications service subscription topic
• Step 3: Create an Events service rule
• Step 4: Create and configure a live table feed to use notifications, and copy the notification URL
• Step 5: Create a Notifications service subscription
• Step 6: Confirm that the endpoint can receive notifications
Tip:

To complete those steps, you will alternate between Oracle Cloud Infrastructure Console pages and Oracle Database Actions pages. You may find it convenient to open the Cloud Console in one browser page or tab and Database Actions in another, so it's easy to move back and forth.

Step 1: Configure your object store bucket to emit notifications

Where: Oracle Cloud Infrastructure Console: Object Storage & Archive Storage - Buckets page

Configure the bucket containing your source data so that it will emit notifications when the data changes. You can set this option when you create a bucket or you can set it in an existing bucket.

1. Open the Cloud Console navigation menu and click Storage. Under Object Storage and Archive Storage, click Buckets.

2. If you're creating a new bucket:
   a. On the Buckets page, click the Create Bucket button to create a new bucket, as described in Managing Buckets. In the Create Bucket wizard, select the Emit Object Events option, along with the other options for your new bucket.
   b. Click Create.

   If you're using an existing bucket:
   a. On the Buckets page, click the name of the bucket you want to use, as described in Managing Buckets.
   b. On the Bucket Details page, click the Edit link next to Emit Object Events.
   c. Select the Emit Objects Events check box, and then click Save Changes.

Step 2: Create a Notifications service subscription topic

Where: Oracle Cloud Infrastructure Console: Notifications - Topics page

1. Open the Cloud Console navigation menu and click Developer Services. Under Application Integration, click Notifications.

2. Click Create Topic, enter a name and optional description, and then click Create. See also Managing Topics and Subscriptions.

Step 3: Create an Events service rule

Where: Oracle Cloud Infrastructure Console: Events - Rules page


2. Click Create Rule, and fill out the Create Rule page as described in Managing Rules for Events.
   - Under Rule Conditions, select:
     - Condition: Event Type
     - Service Name: Object Storage
Event Type: Object - Create

Under Actions, select:

- Action Type: Notifications
- Notifications Compartment: Select the compartment to use for the notifications.
- Topic: Select the name of the topic you created above, in Step 2: Create a Notifications service subscription topic.

3. Click Create Rule.

Step 4: Create and configure a live table feed to use notifications, and copy the notification URL

Where: Database Actions: Live Feeds page

You can configure a new or an existing live table feed to use notifications:

1. Go to the Database Actions Live Feeds page, as described in Feeding Data.
2. Create or edit a live table feed object, as described in Create a Live Table Feed Object or Edit a Live Table Feed Object. Select the Enable for Notification option.
3. Click Create or Save.
4. Click the Actions (three vertical dots) icon on the card for your live feed, and select Show Confirmation URL.
5. In the Notification URL dialog box, click the Copy icon to copy the URL to the clipboard. You may want to copy it to a temporary file, so you can retrieve it later. You’ll use this URL in the next step, Step 5: Create a Notifications service subscription.

Step 5: Create a Notifications service subscription

Where: Oracle Cloud Infrastructure Console: Notifications - Subscriptions page

2. On the Notifications page, click the Subscriptions tab (on the left side of the page), the status will be Active.
3. Click Create Subscription and fill in the Create Subscription page:
   - Subscription topic: Select the subscription topic you created in Step 2: Create a Notifications service subscription topic.
   - Protocol: HTTPS (Custom URL)
   - URL: Paste in the URL you copied in Step 4: Create and configure a live table feed to use notifications, and copy the notification URL.
   - Click Create. The subscription will be listed in the Subscriptions table in a state of “Pending.”

Step 6: Confirm that the endpoint can receive notifications

Where: Database Actions: Live Feeds page

1. Return to the Database Actions Live Feeds page and find the card for the live table feed you are configuring for a notification-based feed.
2. Click the Actions (three vertical dots) icon on the card, and select Show Confirmation URL.

3. In the Confirmation URL dialog box, click the link to confirm the URL. This does not close this dialog box. If the link is successful, a message is displayed that confirms the subscription is active.

4. Return to the Confirmation URL dialog box and select the Check only when the cloud store confirmation process is complete check box, and click OK.

Once you finish the above steps, any new files uploaded to the bucket will automatically be loaded into the live table feed table.

Creating a Notification-Based Live Table Feed using Amazon Simple Storage Service (S3)

You can integrate Amazon Simple Storage Service (S3) and Oracle Cloud Infrastructure (OCI) to automate the process of live feed notifications when storage objects it is observing have updates. The following section provides instructions for creating event notifications in your Amazon S3 bucket where your data files are stored.

Tip:

To complete these steps, you will need to alternate between Amazon Web Services (AWS) Management console and Oracle Database Actions pages. You may find it convenient to open the Amazon Web Services in one browser page or tab and Database Actions in another, so it is easy to move back and forth.

To create a notification- based live feed with Amazon S3 as cloud storage you must:

- Step 1: Create your object store bucket in Amazon S3
- Step 2: Create Access Keys
- Step 3: Add an OCI Cloud Storage using Amazon S3
- Step 4: Create and configure a live table feed to use notifications and copy the notification URL
- Step 5: Create a notifications service subscription topic
- Step 6: Enable and configure event notifications using the Amazon S3 console
- Step 7: Create a notifications service subscription
- Step 8: Confirm that the endpoint can receive notifications

Step 1: Create your object store bucket in Amazon S3

Where: Amazon Web Services (AWS) Management console

Configure and create your bucket containing source data so that it emits notifications when the data changes.

1. Log in to AWS Management console and open the Amazon S3 console.
2. On the home page click the Create Bucket icon.
3. In **Bucket name**, enter a valid name for your bucket. For example: testbucket. After you create the bucket, you cannot change its name.

4. In **Region**, select the Amazon Web Services (AWS) Region from the dropdown. For example: us-west-2

5. In Bucket settings for Block Public Access, select the **Block Public Access** settings that you want to apply to the bucket. It is recommended to keep all settings enabled unless you know that you need to turn any of them off.

6. Select **Advanced settings**, and accept all the default options if you want to enable S3 Object Lock. This step is optional.

7. Select **Create bucket**.

**Step 2: Create Access Keys**

*Where: AWS Management console*

To access Amazon Simple Notification Service (SNS), you must have credentials that Amazon Web Services (AWS) can use to validate your requests. These credentials must have permissions to access Amazon SNS topics. The following steps provide you details on steps to create access keys using AWS Identity and Access Management (IAM) for security purposes.

1. Log in to AWS Management console and open Amazon Identity and Access Management (IAM) console.
2. On the navigation menu, select **Users**.
3. Select your **user name**.
4. In the Security Credentials tab, select **Create access key**.
5. Copy the **Access key ID** and **Secret access key** in the display. Paste them in a clipboard.
6. To download the keys, select **Download.csv file** icon. This way you can store the file in a secure location.

**Step 3: Add an Amazon S3 Cloud Storage Link**

*Where: Database Actions: Manage Cloud page*

Before you create a live table feed, you must establish a connection to the cloud store you want to use.

1. Click the **Manage Cloud Store** button at the top of the page to go to the Manage Cloud page. For further instructions on adding source files residing in cloud storage provided by Amazon S3, refer to **Create an Amazon S3 Cloud Storage Link** topic in Managing Cloud Storage Connections.

**Note:**

Paste the Access key ID and Secret access key generated in the previous step (Step 2: Create Access Keys) to their respective text fields in the Add Cloud Storage page.
Step 4: Create and configure a live table feed to use notifications, and copy the notification URL

Where: Database Actions: Live Feeds page

Creating a live table feed enables you to load data in real time from external storage sources to your table in ADB. External storage you use include as Oracle Object Store, AWS S3 or Microsoft Azure containers.

You can configure a new or an existing live table feed to use notifications:

1. Go to the Database Actions Live Feeds page, as described in Feeding Data.
2. Create or edit a live table feed object, as described in Create a Live Table Feed Object or Edit a Live Table Feed Object. Select the Enable for Notification option
3. Click Create or Save.
4. Click the Actions (three vertical dots) icon on the card for your live feed, and select Show Notification URL.
5. In the Notification URL dialog box, click the Copy icon to copy the URL to the clipboard. You may want to copy it to a temporary file, so you can retrieve it later. You will use this URL in the subsequent step (Step 7: Create a notifications service subscription).

Step 5: Create a notifications service subscription topic

Where: Amazon Simple Notification Service (SNS) console

You receive Amazon S3 notifications using Amazon Simple Notification Service (Amazon SNS) topic. You need to add a notification configuration to your bucket using an Amazon SNS topic. SNS topics are shared locations which are used to send notifications of various events that happen in AWS buckets.

During creation, you select a topic name and topic type. After creating a topic, you cannot change the topic type or name. All other configuration choices are optional during topic creation, which you can edit later.

To access any AWS service, you must first create an AWS account.

Navigate to the AWS Management console, and then select Create an AWS Account.

Follow the instructions as provided in the Amazon SNS link to create your first IAM administrator user and group. Now you can log in to any of the AWS services as an IAM user.

1. Log in to Amazon SNS console as an IAM user.
2. On the Topics page, select Create topic.
3. Specify the following fields on the Create topic page, in the Details section.
   - Type: Standard (Standard or FIFO)
   - Name: notify-topic. For a FIFO topic, add fifo to the end of the name.
   - Display Name: This field is optional.
4. Expand the Encryption section and select Disable encryption.
5. Expand the Access policy section and configure additional access permissions, if required. By default, only the topic owner can publish or subscribe to the topic. This step
is optional. Edit the JSON format of the policy based on the topic details you enter.

Here is a sample of Access policy in JSON format.

```json
{
    "Version": "2008-10-17",
    "Id": "__default_policy_ID",
    "Statement": [{
        "Sid": "__default_statement_ID",
        "Effect": "Allow",
        "Principal": {"AWS": "*"},
        "Action": [
            "SNS:Publish",
            "SNS:RemovePermission",
            "SNS:SetTopicAttributes",
            "SNS:DeleteTopic",
            "SNS:ListSubscriptionsByTopic",
            "SNS:GetTopicAttributes",
            "SNS:AddPermission",
            "SNS:Subscribe"
        ],
        "Condition": {
            "StringEquals": {
                "AWS:SourceOwner": "555555555555"
            }
        }
    },
    { "Sid": "s3_policy", //This field accepts string values
        "Effect": "Allow",
        "Principal": {
            "Service": "s3.amazonaws.com"
        },
        "Action": [
            "SNS:Publish"
        ],
        "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "555555555555" //This is the Account ID
            }
        },
        "ArnLike": {
            "aws:SourceArn": "arn:aws:s3:*:*:testbucket /*testbucket is the s3 bucket name. You will get notifications only when file is uploaded to this bucket. */
        }
    }
}
```
6. Expand the Delivery retry policy (HTTP/S) section to configure how Amazon SNS retries failed message delivery attempts. This step is optional.

7. Expand the Delivery status logging section to configure how Amazon SNS logs the delivery of messages to CloudWatch. This step is optional.

8. Expand Tags section to add metadata tags to the topic. This step is optional.

9. Select Create topic.

10. The topic’s Name, ARN (Amazon Resource Name), and Topic owner's AWS account ID are displayed in the Details section.

11. Copy the topic ARN to the clipboard.

Step 6: Enable and configure event notifications using the Amazon S3 console

Where: Amazon S3 Management console

You can enable Amazon S3 bucket events to send a notification message to a destination whenever those events occur. You configure event notifications for your S3 bucket to notify OCI when there is an update or new data available to load. The following steps explain the procedure to be followed in Amazon S3 console to enable event notifications.

1. Log in to Amazon S3 Management console and sign in as an IAM (Amazon Identity and Access Management) user.

2. In the Buckets list, select the name of the bucket i.e. testbucket. This is the bucket that you had created in Step 1: Create your object store bucket in Amazon S3.

3. Select Properties icon.

4. Navigate to the Event Notifications section and select Create event notification icon.

5. In the General configuration section, specify the following values for event notification.
   - **Event name**: bucket-notification
   - **Prefix**: This value is to filter event notifications by prefix. It is an optional value. This is added to filter event activity.
   - **Suffix**: This value is to filter event notifications by suffix. It is an optional value. This is added to filter event activity.

6. In the Event types section, select one or more event types that you want to receive notifications for. If you are unsure of what event types to pick, then select the All object create events option.

7. In the Destination section, select SNS Topic as the event notification destination.

Note:

Before you can publish event notifications, you must grant the Amazon S3 the necessary permissions to call the relevant API. This is so that it can publish notifications to a Lambda function or an SNS topic.

8. After you select SNS topic as the event notification destination, select the SNS topic i.e. notify-topic from dropdown. This is the topic you created in Step 5: Create a notifications service subscription topic.

9. Select Save changes.
Step 7: Create a notifications service subscription

Where: Amazon SNS console

Every Amazon SNS topic has a set of subscriptions. Once a message is published to a topic, SNS handles distributing the message to all its subscribers. The subscribers can be AWS Lambda functions, HTTP(S) endpoints, email addresses and mobile phone numbers capable of receiving SMS messages.

Amazon SNS matches the topic to a list of subscribers who have subscribed to that topic and delivers the message to each of those subscribers.

1. Log in to Amazon SNS console.
2. In the left navigation pane, select Subscriptions.
3. Select Create subscription on the subscriptions page.
4. In the Details section of the Create subscription page, specify the following values.
   • Topic ARN: Paste the ARN value copied from previous step (Step 5: Create a notifications service subscription topic).
   • Protocol: HTTPS
   • Endpoint: Paste the endpoint value you copied while creating the live table feed in previous step (Step 4: Create and configure a live table feed to use notifications and copy the notification URL).
5. Expand the Subscription filter policy section to configure a filter policy. This step is optional.
6. Expand the Redrive policy (dead-letter queue) section to configure a dead-letter queue for the subscription. This step is optional.
7. Select Create subscription.

Note:

HTTP(S) endpoints, email addresses, and AWS resources in other AWS accounts require confirmation of the subscription before they can receive messages.

Step 8: Confirm that the endpoint can receive notifications

Where: Database Actions: Live Feeds page

1. Return to the Database Actions Live Feeds page and find the card for the live table feed you are configuring for a notification-based feed.
2. Click the Actions (three vertical dots) icon on the card, and select Show Confirmation URL.
3. In the Confirmation URL dialog box, click the link to confirm the URL. This does not close this dialog box. If the link is successful, a message is displayed that confirms the subscription is active.
4. Return to the **Confirmation URL** dialog box and select the Check only when the cloud store confirmation process is complete check box, and click **OK**.

Once you finish the above steps, any new files uploaded to the bucket will automatically be loaded into the live table feed table.

For more information on how to enable and configure event notifications using the Amazon S3 console, refer [Enabling and configuring event notifications using the Amazon S3 console](#).

**Creating a Notification-Based Live Table Feed using Microsoft Azure**

A notification-based Live Table Feed is an interface between Oracle Cloud Infrastructure and a third-party cloud message queuing service such as Azure Event Grid.

The following section explains the procedure to generate automatic live feed messages using Microsoft (MS) Azure as the cloud storage. When there is an update in the container and the notification conditions are met, a log message is generated and displayed in the live feed in Oracle Cloud Infrastructure.

To create a notification-based live feed with Microsoft Azure as cloud storage you must:

- **Step 1**: Create a resource group in Microsoft Azure
- **Step 2**: Create a storage account in Microsoft Azure
- **Step 3**: Create Access Keys
- **Step 4**: Create a container
- **Step 5**: Add cloud storage using Microsoft Azure cloud store
- **Step 6**: Create and configure a live table feed to use notifications and copy the notification URL
- **Step 7**: Enable Event Resource Provider
- **Step 8**: Create Event subscription
- **Step 9**: Confirm that the endpoint can receive notifications

**Tip:**

To complete the steps above, you will need to alternate between Microsoft Azure portal and Oracle Database Actions pages. You may find it convenient to open the Microsoft Azure portal in one browser page or tab and Database Actions in another, so it is easy to move back and forth.

**Step 1: Create a resource group in Microsoft Azure**

*Where*: Microsoft Azure Portal

Resource groups are logical containers where you can manage Azure resources like storage accounts. Resource groups are created so you can deploy, update and delete them as a group. You can create a resource group by following these steps:

1. On the Azure portal, click the **Resource groups** button.
2. Select **Add**.
3. Enter the following values:
• **Subscription**: Select your Azure subscription, such as Microsoft Azure Enterprise.

• **Resource group**: Enter a new resource group name, such as resource-group.

• **Region**: Select your location, such as US west.

4. Click **Review+create**.

5. Click **Create**. It takes a few seconds to create a resource group.

**Step 2: Create a storage account in Microsoft Azure**

*Where*: Microsoft Azure Portal

An Azure storage account contains all your storage data objects like blobs, tables, disks etc. You can create a storage account inside the resource group. It provides a unique namespace for your data. To create a storage account, do the following:

1. From the left portal menu, select **Storage accounts** to display a list of your storage accounts.

2. On the Storage accounts page, click the **Create** icon.

3. On the Basic tab, provide the following information for your storage account.
   • **Subscription**: Microsoft Azure Enterprise
   • **Resource group**: resource-group
   • **Storage account name**: teststorage
   • **Region**: Select your location, such as US west.
   • **Redundancy**: Locally-redundant storage (LRS)

4. You can select **Review+create** to accept the default options and proceed to validate the account.

5. After the validation passes, you can proceed to click on **Create storage account**. In case the validation fails, the portal indicates which settings must be modified.

**Step 3: Create access Keys**

*Where*: Microsoft Azure Portal

You must grant Microsoft Azure the permissions necessary to obtain access keys on your storage locations. The access keys specific to the storage account are generated automatically after the storage account is created in the previous step. The following steps describes the procedure to create access keys.

1. In **Security+Networking**, select **Access keys**. Your account access keys appear with the complete connection string for each key.

2. Select **Show keys** to show your access keys and connection string for each key and to copy values.

3. Copy the connection string value under key1. This value will be pasted in **Azure storage account access key** text field of next step ([Step 5: Add cloud storage using Microsoft Azure cloud store](#)).

4. Copy the storage account name i.e. teststorage and paste it in **Azure storage account name** text field of next step ([Step 5: Add cloud storage using Microsoft Azure cloud store](#)).

5. Test the credentials to see if it works or not.
Step 4: Create a container

Where: Microsoft Azure Portal

A container is a location (also known as buckets in Amazon S3 and OCI) which holds Azure Blob (Binary large object) storage. Follow these steps to create a container.

1. Navigate to your new storage account in the Azure portal.
2. In the left menu for the storage account, scroll to the Data storage section, then select Containers.
3. Click the +Container icon.
4. Enter the name for your new container. The container name must be lowercase, must start with a letter or number, and can include only letters, numbers, and the dash character.
5. Set the level of Public Access Level to Private. The default level is Private.
6. Select Create to create the container.

Step 5: Add cloud storage using Microsoft Azure cloud store

Where: Database Actions: Manage Cloud page

1. Click the Manage Cloud Store button at the top of the page to go to the Manage Cloud page. For further instructions on adding source files residing in cloud storage provided by Microsoft Azure cloud storage, refer to Create an Microsoft Azure Cloud Storage Link topic in Managing Cloud Storage Connections section.

Note:

Paste the connection string value under key 1 of previous step Step 3: Create Access Keys in Azure storage account access key text field of Add Cloud Storage page. Also paste the storage account name generated in the previous step Step 3: Create Access Keys in Azure storage account name text field of Add Cloud Storage page.

Step 6: Create and configure a live table feed to use notifications and copy the notification URL

Where: Database Actions: Live Feeds page

The Live table feed object enables data to be loaded from Microsoft Azure cloud storage with no polling delay. This object creates an integration between Oracle Cloud Interface and Microsoft Azure.

You can configure a new or an existing live table feed to use notifications:

1. Go to the Database Actions Live Feeds page, as described in Feeding Data.
2. Create or edit a live table feed object, as described in Create a Live Table Feed Object or Edit a Live Table Feed Object. Select the Enable for Notification option.
3. Click Create or Save.
4. Click the **Actions** (three vertical dots) icon on the card for your live feed, and select **Show Confirmation URL**.

5. In the **Notification URL** dialog box, click the **Copy** icon to copy the URL to the clipboard. You may want to copy it to a temporary file, so you can retrieve it later. You will use this URL in the subsequent step, *(Step 8: Create Event subscription)*.

**Step 7: Enable Event Resource Provider**

*Where*: Microsoft Azure Portal

If this is the first time you are using Event Grid, you must enable Event Grid resource provider.

1. Select **Subscriptions** on the left menu.
2. Select the subscription you are using for Event Grid i.e. Microsoft Azure Enterprise.
3. On the left menu, under Settings, select **Resource Providers**.
4. Search **Microsoft.EventGrid**.
5. Select **Register**.

It takes a minute for the registration to finish.

**Step 8: Create Event subscription**

*Where*: Microsoft Azure Portal

You create an Event Subscription by configuring the subscription and specifying the endpoint that will receive the notifications.

1. Select the storage account you created in Step 2: Create a storage account in Microsoft Azure.
2. Select the **Events** icon in the left navigation pane.
3. Click on **+Event Subscription**.

The **Create Event Subscription** window appears.

1. Specify the following fields in **Event Subscription** details section:
   - **Name**: Eventssub. This is the name of the Event Subscription we create.
   - **Event Schema**: Event Grid Schema
2. Specify the following fields in **Topic Details** section:
   - **Topic Type**: Storage account
   - **System Topic Name**: eventtopic.
3. Specify the following fields in **Event Types** section:
   - **Event Type**: MicrosoftStorage.BlobCreated
4. Specify the following fields in **Endpoint Details** section:
   - **Endpoint Type**: Web Hook
   - **Endpoint**: Paste the Notification URL copied in Step 6: Create and configure a live table feed to use notifications and copy the notification URL.
5. Select **Create**.
This way Microsoft Azure creates a system topic first and then the Event Subscription for the topic.

**Step 9: Confirm that the endpoint can receive notifications**

*Where:* Database Actions: Live Feeds page

1. Return to the Database Actions Live Feeds page and find the card for the live table feed you are configuring for a notification-based feed as created in Step 6: Create and configure a live table feed to use notifications and copy the notification URL.

2. Click the **Actions** (three vertical dots) icon on the card, and select **Show Confirmation URL**.

3. In the **Confirmation URL** dialog box, click the link to confirm the URL. This does not close this dialog box. If the link is successful, a message is displayed that confirms the subscription is active.

![Note:]

The **Confirmation URL** link expires after a few minutes. You must make sure that you click the link before it expires.

4. Return to the **Confirmation URL** dialog box and select the **Check only when the cloud store confirmation process is complete** check box, and click **OK**.

Once you finish the above steps, upload a new file to the Microsoft Azure container you created in Step 4: Create a container.

1. Navigate to the container you created.

2. Select the **Container** to show a list of blobs it contains.

3. Select **Upload** button to open your local repository and browse the file you need to upload as a block blob.

4. Select the **Upload** button to upload the blob.

5. You can now view the new blob listed within the container.

6. Return to the Database Actions Live Feeds page and find the card for the live table feed you are configuring for a notification-based feed.

7. Click the **Actions** (three vertical dots) icon on the card, and select **Live table feed Run Details**.

You should be able to view logs for the blob uploaded to the Live Feed table from the Microsoft Azure storage in **Live table feed Run Details** window.

For more details on how to create a topic and subscription in Azure portal, refer to **Azure Event Grid Notifications**.
The Catalog Page

The Catalog page displays information about entities in the Oracle Autonomous Database.

To reach the Catalog page, click Catalog in the Database Actions page, or click the Selector icon and select Catalog from the Data Tools menu in the navigation pane.

The following topics describe the Catalog page and how to use it.

• About the Catalog Page
• Sorting and Filtering Entities
• Searching for Entities
• Viewing Entity Details

About the Catalog Page

Use the Catalog page to get information about the entities in and available to your Oracle Autonomous Database. You can see the data in an entity, the sources of that data, the objects that are derived from the entity, and the impact on derived objects from changes in the sources.

The Catalog page lists and displays details about:

• Database objects such as business models, cloud storage links, and tables that are created by Database Actions applications such as Data Analysis and Data Load.
• Database data dictionary objects such as tables, columns, database links, analytic views, packages, and procedures that have been created by the database tools or a database application such as SQL Developer.

When you first open the Browse catalog page, it contains the Search catalog field, the Catalog User Preferences button, a list of recently viewed objects, and several suggested searches. You can enter a search string, click on one of the recent objects to see its details, or click on Show search suggestions icon to view suggestions on the right side of the page.
As soon as you perform a search or select an item, the page is refreshed with additional items, as described below.

The Catalog page contains:

1. **Search catalog field**
   You can:
   - Click the field and type or paste a new search string into it.
   - Click the field and edit a search string already in the field.
   - Click the field and build a query by selecting items from the drop-down list.
   You can modify an existing query or build a new one. The **query scope** icon takes the current search query and turn it to saved query scope.

   The saved query drop-down besides this field lets you search for the entities from the selected query scopes that you had previously saved using the **query scope** icon.

   Click X to clear your searches in the Catalog field.

   For details about how to construct a search string, see Searching for Entities.

2. **Toolbar**
   The toolbar appears after you run an initial search. It contains these buttons:
• **Sort By**

To select sorting values, click the **Sort By** button to open the list of options. Then click the **Ascending** or **Descending** icon next to one or more of the sorting values. For example, if you select the **Ascending** icon next to **Entity name** and the **Descending** icon next to **Entity type**, the entities will be sorted in alphabetical order by entity name and then in reverse alphabetical order by entity type.

Click **Reset** in the list to clear the choices in the list.

The sorting values you choose are listed next to the **Sort by** label beneath the toolbar. Click the X icon on a sorting value to remove it.

• **Page size**

By default, up to 25 entities are displayed on the page. If you want more entities on a page, select a number from this list.

• **Previous** and **Next**

If the search results are displayed on multiple pages, click these buttons to navigate through the pages.

• **Refresh**

Click to refresh the entities shown on the page, based on the current search string.

• **Entity view options**

Choose one of these three options to set how entities are displayed on the page.

Click **Open Card view** to display entities as card arranged into one or two columns; click **Open Grid View** to display entities as rows in a table; or click **Open List View** to display entities in a single column of panels.

The above views display information about the entity, including name, type, owner, application, and other details, depending on the entity view option and on the entity type.

Click **Action** and select **View Details** to view the details of an entity. In the Card view and the List view, you can also click the name of the entity to show its details.

• **Suggestions**

Click to open or close the **Suggestions** panel.

3. **Catalog User Preferences**

When you first open the Catalog, this button appears to the top of the **Search catalog** field. After performing a search, it appears in the toolbar.

Click **Catalog User Preferences** to set the behavior of the Catalog. When you set these options, they take place immediately and are also saved as the default behavior for the page. Clicking Catalog User Preferences opens the General tab where you can view the following options.
• **Show system tables**
  Select this option to include system tables in the search results.

• **Show private tables**
  Select this option to include private tables in the search results.

• **Page size**
  Select the number of entities to display on the page.

• **Entities view**
  Select how entities are shown on the page. There are three options: Card view (default) displays entities as cards arranged into one or two columns; Grid view displays entities as rows in a table; and List view displays entities in a single column of panels.

• **Search criteria**
  Enter a search string that will be used when you click User configured search in the Suggestions panel on the right side of the Catalog page. For example:

  owner: your_schema AND (type: TABLE OR type: ANALYTIC_VIEW OR type: VIEW)

• **Save last 5 search queries**
  Select this option to save the five previous search queries that you entered into the Search catalog field. When the Search catalog field is empty and you click in the field, the last five search queries are listed in the drop-down list. (Any predefined searches selected from the Suggestions panel won't appear in this list, unless you edited them to make a new search.)

Gradually progress to use the Query Scopes tab to view, create, and delete query scopes. You can search for catalogs and save this search using query scopes. See Searching for Entities for exploring this tab.

4. **Filters panel**
   Select one or more filter values to limit the entities shown on the page. Only those entities that match the filter values are shown. That is, the items returned by a search are filtered by these filter settings. Selecting all or none of the options shows all entities. See Sorting and Filtering Entities.

5. **Sort by settings**
   When you set sorting values by using the Sort By control in the toolbar (see above), the settings are displayed in small boxes beneath the toolbar. You can delete a setting by clicking the X icon in the box. Or you can change the settings by returning to the Sort By control in the toolbar.

6. **Display area**
   The area beneath the Search catalog field displays the entities returned by a search and that match the filter criteria set in the Filters panel. You can sort the entities by clicking the Sort By button and then setting sort values. See Sorting and Filtering Entities.
Click the name of the entity or click Action to view details about the entity. See Viewing Entity Details.

7. **Suggestions** panel

The items listed under **Suggestions** represent predefined queries (search strings) that search for the items described in the link. For example, **All tables** is associated with the search string `type:TABLE`. When you select one of the suggested search strings, the string is inserted into the **Search catalog** field, the query is run, and the results are displayed on the page. You can modify the search string in the **Search catalog** field to modify the search.

### Sorting and Filtering Entities

On the Catalog page, you can sort and filter the displayed entities.

#### Sort Entities

To set a sort order for the entities on the page,

1. Click **Sort by** on the toolbar.
2. Find the value you want to use to sort, for example **Entity Type**.
3. Click the **Ascending** icon or the **Descending** icon next to that value.

After you've selected a value, it is displayed in a box next to the **Sort by** label beneath the toolbar.

4. You can set one or more parameters. If you want to add a parameter to your search, repeat the steps above. For example, you can set **Entity name (ASC)** and then set **Created on (DESC)**. Both appear as separate boxes next to the **Sort by** label beneath the toolbar.

To remove a sort value, you can do either of the following:

- Click the **X** icon in the box that shows the sort value (next to the Sort by label beneath the toolbar).
- Click **Sort by** on the toolbar and click the **Ascending** or **Descending** icon next to the value you want to remove.

To return to the default (**Updated on DESC**), click **Sort by** and select **Reset**.

#### Filter Entities

Restrict which entities that are returned by a search are displayed on the page by setting filters in the **Filters** panel on the left side of the Catalog page. Select one or more filter values. Only those entities that are returned by a search and that match the filter values are shown. Selecting all or none of the options shows all entities returned by the search.
By default, system tables and private tables are not displayed. To display them, click Show User Preferences in the toolbar and then click the Show system tables slider or the Show private tables slider, or both.

Searching for Entities

Search for entities in the Catalog by entering a search string (query) into the Search catalog field at the top of the Catalog page. You can type or paste in a new search string, edit an existing one, or construct a search string by clicking in the field and selecting items from the drop-down list.

The search operates on the entities that meet the filter criteria you have set in Filters panel. To clear the Search catalog field, click the X icon.

Entering Search Strings

Syntax and instructions for creating search strings are presented below.

You can enter a search string into the Search catalog field in several ways:

- Click an item under Suggestions on the right side of the page. When you click one of these items, a search string is entered into the Search catalog field and the search is run.
- Type or paste a string directly. When you start typing, suggestions are shown in the drop-down list below the Search catalog field. You can keep typing or you can select a suggestion from the list (see next bullet).
- Select or construct a search string by clicking the Search catalog field and selecting items.

If the Search catalog field is empty, the drop-down list contains basic search parameters you can select to start building the query. When you select one, or when you click one of the links under Suggestions, the string is inserted in the Search catalog field, and the drop-down list is updated to present possible additional parameters. You can choose the additional parameters, or edit the string directly, or both.

If the Search catalog field is empty and if you set Save last 5 search queries in your preferences, the last five search strings that you entered into the Search catalog field will also appear in the drop-down. (Searches initiated by clicking one of the items under Suggestions won't appear in this list.)

When you finish constructing the string, press Enter or click the arrow button on the right side of the field.

Click the → icon at the end of the Search field to accept the default query and view all the current owner's tables, views, and analytic views.

Enter a string in the Search Catalog field to find the entities (schema, table, or view) whose label includes the specific string you enter. Click on the Query Scope icon to save your search. This allows you to store frequently used searches and access them instantly without any inconvenience.

Selecting Query Scope icon prompts a Catalog User Preferences wizard which enables you to save your catalogs based on your preferences.
You can view, create, and edit the previously saved query scopes in this wizard. The query scopes are categorized based on by whom its created.

- **Select Custom** to view, create, edit, or delete the query scopes created by you.
- **Select Predefined** to view the query scopes which are already defined by the Database Actions. This option does not allow you to create, edit, or delete the query scopes.
- **Select All** to view all the query scopes. This includes Custom and Predefined query scopes.

Specify the following fields in the Query Scope tab.

- **Name**: Enter the name of the Query Scope. This is a mandatory field.
- **Label**: This is a mandatory field. Enter a descriptive name here. You will use this field to refer to a query scope.
- **Definition**: Enter the Oracle Autonomous Database Data Definition Language (DDL) that creates the search entity. This is the same search criteria you enter in the Search Catalog field.

Click **Create** to create the Query Scope. Click **Cancel** to cancel its creation.

Once you have created the new query scope, it is visible in the list of query scopes in the Catalog User Preferences wizard.

This wizard also appears on selecting **Catalog User Preferences**. See **Catalog User Preferences** for details.

### Create a saved search

You can save your time from redefining the same search again in the future. You can diagnose problems faster since you are just few clicks away from accessing a saved search. Here is how you can create a saved search.

1. From the browse objects drop-down list, select the objects from where you want to search the entities (schema, table, or view).
2. Specify the search criteria in the Search Catalog field. For example, you want to search for a specific Entity type and a specific owner.
3. Click **Add** to Saved Searches icon.
4. Clicking the icon prompts the Catalog User Preferences wizard.

5. Enter the name of the saved search in the Title field. The wizard automatically generates the Scope of the search. The definition of the search is also automatically created by concatenating fields used in the Search Catalog field, for example, type: TABLE units AND owner: ADPTEST.

6. Enter description of the search in Description field. This is not a mandatory step.

7. Select Create to create the saved search. Click Cancel to cancel its creation.

After the creation of the saved search, it appears on the list of Saved Searches.

You can change the columns displayed in the search results by clicking the pencil icon in the Actions column. Click the delete icon in the Actions column to delete the search you save. The saved searches you create are available for selection in the Saved Search panel in the right of the Catalog page.

Basic Structure

The query string consists of a set of search terms. Here are some examples.

- sales
- type:TABLE
- owner!=SH
- #deployment
- type:TABLE,VIEW

Combine search terms by using the two Boolean operators AND and OR:

- sales and type:TABLE
- sales or type:TABLE

If you don't specify an explicit operator, then AND is assumed. All of the following are equivalent:

- sales type:TABLE owner:sh
- sales AND type:TABLE owner:sh
- sales type:TABLE AND owner:sh
- sales AND type:TABLE AND owner:sh

Negate a single search term by prefacing it by either NOT or by a - (hyphen). All of the following exclude tables from the search:

- sales and NOT type:TABLE
- sales -type:TABLE

Enclose search terms in parentheses to control order:

- sales and (type:TABLE or type:VIEW)
- sales and NOT (type:TABLE or type:VIEW)

If you don't add parentheses, the operators are read from left to right. The following are equivalent:

- sales or type:TABLE and owner:SH
Search Terms

Search terms come in three forms:

• Simple Search Terms
• Property Search Terms
• Classification Search Terms

Simple Search Terms

A simple search term is a simple string, with or without quotes.

• Unquoted string: sales
• Single-quoted string: 'sales'
• Double-quoted string: “sales”

All three types compare the given value, sales in the examples above, to the ENTITY_NAME, but they differ in how the comparison works, as shown in the table below:

<table>
<thead>
<tr>
<th>Search String</th>
<th>Equivalent SQL WHERE Clause</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>sales</td>
<td>WHERE REGEXP_LIKE(entity_name, 'SALES')</td>
<td>A plain regular expression.</td>
</tr>
<tr>
<td>'sales'</td>
<td>WHERE REGEXP_LIKE(entity_name, 'sales')</td>
<td>Similar to the unquoted version, but the search term is not automatically converted to uppercase.</td>
</tr>
<tr>
<td>sales costs</td>
<td>WHERE REGEXP_LIKE(entity_name, 'SALES') AND REGEXP_LIKE(entity_name, 'COSTS')</td>
<td>Treated as two independent search terms combined with the AND operator.</td>
</tr>
<tr>
<td>'sales costs'</td>
<td>WHERE REGEXP_LIKE(entity_name, 'sales costs')</td>
<td>Treated as a single search term. You use single quotes to handle spaces and other special characters, such as colons.</td>
</tr>
<tr>
<td>'don''t'</td>
<td>WHERE REGEXP_LIKE(entity_name, 'don''t')</td>
<td>Escaped single quotes within a search string.</td>
</tr>
<tr>
<td>&quot;sales&quot;</td>
<td>WHERE entity_name = 'sales'</td>
<td>The search string is used as an exact match.</td>
</tr>
</tbody>
</table>

Property Search Terms

A property search term is a combination of three items:

1. The name of a search property (for example, name, type, or owner)
2. A search operator (for example, :, >, or !=)
3. A search value (for example, sales, 'sales', or "sales")

For example:

• name: sales
• owner=SH
• daysSinceCreated>20

You can also specify a comma-delimited list of search values when using the operators :, =, or ~=

• If the operator is : or =, then the condition works like an IN LIST. For example, name:X,Y is equivalent to entity_name IN ('X', 'Y') and to (entity_name = 'X' OR entity_name = 'Y').
• If the operator is !=, then the condition works like a NOT IN LIST. For example, type!=TABLE,VIEW is equivalent to entity_type NOT IN ('TABLE', 'VIEW') and to (entity_type != 'TABLE' AND entity_type != 'VIEW').

The property name must be one of the following predefined strings. Property names are case sensitive, so TYPE, for example, is not supported.

Some properties apply to all entity types, and some properties apply only to a specific entity type, as shown in the two tables below.

The following table shows those properties that apply to all entity types.

<table>
<thead>
<tr>
<th>Property</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>application</td>
<td>The name of the entity application as defined by the ALL_LINEAGE_APPLICATIONS view. Examples include DATABASE and INSIGHTS.</td>
</tr>
<tr>
<td>created</td>
<td>The timestamp when the entity was created.</td>
</tr>
<tr>
<td>dateCreated</td>
<td>The date when the entity was created. This is the same as created, but truncated to the nearest day.</td>
</tr>
<tr>
<td>dateUpdated</td>
<td>The date when the entity was last updated. This is the same as created, but truncated to the nearest day.</td>
</tr>
<tr>
<td>daysSinceCreated</td>
<td>The number of days since the entity was created. The value is zero if the entity was created today.</td>
</tr>
<tr>
<td>daysSinceUpdated</td>
<td>The number of days since the entity was last updated. The value is zero if the entity was updated today.</td>
</tr>
<tr>
<td>link</td>
<td>The name of the database link where the entity is defined. This can be used to search entities in other, linked, databases.</td>
</tr>
<tr>
<td>local</td>
<td>The value YES if the entity is defined within the database itself; otherwise, the value NO. Tables or Insights defined in a schema are examples of local entities. Objects in cloud storage, such as CSV or Parquet files, are examples of entities that are not local.</td>
</tr>
<tr>
<td>name</td>
<td>The name of the entity.</td>
</tr>
<tr>
<td>namespace</td>
<td>The namespace of the entity as defined by ALL_LINEAGE_NAMESPACES.</td>
</tr>
<tr>
<td>oracleMaintained</td>
<td>The value YES, if the entity is created and maintained by Oracle; otherwise, the value NO. The ALL_TABLES view is an example of an Oracle maintained entity.</td>
</tr>
<tr>
<td>owner</td>
<td>The owner of the entity.</td>
</tr>
<tr>
<td>parent</td>
<td>The full entity path of the parent entity, if it exists.</td>
</tr>
<tr>
<td>namesSpace</td>
<td>The name of the parent entity, if it exists.</td>
</tr>
<tr>
<td>parentPath</td>
<td>The full entity path of the parent entity, if it exists.</td>
</tr>
<tr>
<td>parentType</td>
<td>The type of the parent, if it exists.</td>
</tr>
<tr>
<td>Property</td>
<td>Meaning</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>path</td>
<td>The full path of the entity.</td>
</tr>
<tr>
<td>rootName</td>
<td>The name of the outermost containing entity. If the entity has no parent, then the rootName is equal to the entity name. If the entity does have a parent, then the rootName is defined, recursively, as the rootName of the parent entity.</td>
</tr>
<tr>
<td>rootNamespace</td>
<td>The namespace of the outermost containing entity. If the entity has no parent, then the rootNamespace is equal to the entity namespace. If the entity does have a parent, then the rootNamespace is defined, recursively, as the rootNamespace of the parent entity.</td>
</tr>
<tr>
<td>type</td>
<td>The entity type, as defined by ALL_LINEAGE_ENTITY_TYPES.</td>
</tr>
<tr>
<td>updated</td>
<td>The timestamp when the entity last updated.</td>
</tr>
</tbody>
</table>

Some searchable properties are specific to certain entity types. The following table shows those entity-type-specific properties. When searching for entities with these properties, it will speed up your search to specify the entity type. For example, instead of just searching on numRows, specify the entity type that has the numRows property:

<table>
<thead>
<tr>
<th>Entity Type</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>TABLE</td>
<td>numRows - The number of rows in the table.</td>
</tr>
<tr>
<td></td>
<td>status - The status of the object: VALID or INVALID.</td>
</tr>
<tr>
<td></td>
<td>partitioned - Indicates whether the table is partitioned (YES) or not (NO).</td>
</tr>
<tr>
<td></td>
<td>external - Indicates whether the table is an external table (YES) or not (NO).</td>
</tr>
<tr>
<td></td>
<td>sharded - Indicates whether the object is sharded (Y) or not (N).</td>
</tr>
<tr>
<td>COLUMN</td>
<td>dataType - The data type of the column.</td>
</tr>
<tr>
<td></td>
<td>nullable - Indicates whether a column allows NULLs. The value is N if there is a NOT NULL constraint on the column or if the column is part of a PRIMARY KEY. Otherwise, the value is Y.</td>
</tr>
<tr>
<td>PACKAGE</td>
<td>status - The status of the object: VALID or INVALID.</td>
</tr>
<tr>
<td>PROCEDURE</td>
<td>status - The status of the object: VALID or INVALID.</td>
</tr>
<tr>
<td>ATTRIBUTE_DIMENSION</td>
<td>status - The status of the object: VALID or INVALID.</td>
</tr>
<tr>
<td>HIERARCHY</td>
<td>status - The status of the object: VALID or INVALID.</td>
</tr>
<tr>
<td>ANALYTIC_VIEW</td>
<td>status - The status of the object: VALID or INVALID.</td>
</tr>
<tr>
<td>MEASURE</td>
<td>dataType - Data type of the measure, such as NUMBER.</td>
</tr>
<tr>
<td></td>
<td>nullable - Indicates whether a column allows NULLs. The value is N if there is a NOT NULL constraint on the column or if the column is part of a PRIMARY KEY. Otherwise, the value is Y.</td>
</tr>
<tr>
<td></td>
<td>measureType - Type of the OLAP measure:</td>
</tr>
<tr>
<td></td>
<td>• BASE - Base measures store the data</td>
</tr>
<tr>
<td></td>
<td>• DERIVED - Derived measures calculate the data from base measures; also called calculated measures.</td>
</tr>
<tr>
<td>MINING_MODEL</td>
<td>status - The status of the object: VALID, INVALID, or N/A.</td>
</tr>
<tr>
<td>Entity Type</td>
<td>Properties</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>FUNCTION</td>
<td>status - The status of the object: VALID, INVALID, or N/A.</td>
</tr>
<tr>
<td>DIRECTORY</td>
<td>status - The status of the object: VALID, INVALID, or N/A.</td>
</tr>
<tr>
<td>EXTERNAL_LOCATION</td>
<td>fileName - The name of a file in Oracle Directories.</td>
</tr>
<tr>
<td></td>
<td>url - The URI of file a file in object storage.</td>
</tr>
<tr>
<td>LIVE_TABLE_FEED</td>
<td>enableNotifications - Indicates whether a live table feed is enabled for notifications (TRUE) or not (FALSE).</td>
</tr>
</tbody>
</table>

The operator must be one of the following.

<table>
<thead>
<tr>
<th>Property</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>=</td>
<td>Equal to</td>
</tr>
<tr>
<td>!=</td>
<td>Not equal to</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater than</td>
</tr>
<tr>
<td>&gt;=</td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td>&lt;</td>
<td>Less than</td>
</tr>
<tr>
<td>&lt;=</td>
<td>Less than or equal to</td>
</tr>
<tr>
<td>:</td>
<td>Equal to For example, type:TABLE is the same as type=TABLE.</td>
</tr>
<tr>
<td>~=</td>
<td>Represents REGEXP_LIKE For example, the following two search items are equivalent:</td>
</tr>
<tr>
<td></td>
<td>• sales</td>
</tr>
<tr>
<td></td>
<td>• name~=sales</td>
</tr>
</tbody>
</table>

The value of the property search term is a string. As with the simple search term, this string can be unquoted, single-quoted, or double-quoted. Unquoted strings are converted to uppercase and quoted strings are left as they are.
### Search String

<table>
<thead>
<tr>
<th>Search String</th>
<th>Equivalent SQL WHERE Clause</th>
</tr>
</thead>
<tbody>
<tr>
<td>name=sales</td>
<td>WHERE entity_name = 'SALES'</td>
</tr>
<tr>
<td>name='sales'</td>
<td>WHERE entity_name = 'sales'</td>
</tr>
<tr>
<td>type:table</td>
<td>WHERE entity_type = 'TABLE'</td>
</tr>
<tr>
<td>parent~='COSTS'</td>
<td>WHERE REGEXP_LIKE(entity_parent_path, 'COSTS')</td>
</tr>
<tr>
<td>daysSinceCreated&gt;10</td>
<td>WHERE TRUNC(CURRENT_TIMESTAMP AT TIME ZONE 'UTC') - TRUNC(&quot;CREATED&quot;) &gt; 10</td>
</tr>
<tr>
<td>owner!='DemoUser'</td>
<td>WHERE owner != 'DemoUser'</td>
</tr>
<tr>
<td>parentName=null</td>
<td>WHERE entity_parent_name IS NULL</td>
</tr>
<tr>
<td>parentName!=null</td>
<td>WHERE entity_parent_name IS NOT NULL</td>
</tr>
</tbody>
</table>

### Classification Search Terms

Classifications are metadata about an entity, such as a caption or a description. Classification search terms are similar to property search terms, but they work with entity classifications such as captions or descriptions instead of standard entity properties. The name of the classification must be prefixed by a hash tag (#) and may be unquoted or enclosed in double quotation marks ("). A query converts unquoted classifications to uppercase, which means that the following forms of the CAPTION classification are equivalent.

- #caption
- #Caption
- #CAPTION
- #"CAPTION"

Classifications are, by their nature, multi-lingual in the sense that the value can vary by National Language Support (NLS) language. By default, the query syntax uses the value of the classification in the current NLS language. You can specify a specific NLS language by adding the name of the language after a forward slash. The language can be quoted, but the query converts it to uppercase in all forms.

- #caption/French
- #caption/"ITALIAN"

If the name of the specified language contains a space (for example, "CANADIAN FRENCH"), you must either enclose the name in quotes or replace the space with an underscore character (or both). The following are equivalent:

- #caption/Canadian_French
- #caption/"CANADIAN FRENCH"
- #caption/"Canadian_French"

If you use a classification name on its own as a search term, then the search returns entities that have any non-null value for that classification. If you've defined a classification called DEPLOYED, for example, then you can see all entities with the classification by using a simple term:

- #deployed

As with other search terms, you can negate it using NOT or - (hyphen). To exclude all entities with the DEPLOYED classification, you can use either of the following searches:
• NOT #DEPLOYED
• -#deployed

You can also use classifications with operator/value pairs, with the same semantics used by property search items; for example:
• #caption~=sales
• #description/Spanish=Ventas

Searching on Multiple Properties

The default for this is ENTITY_NAME, so that if you enter a query string like "sales" it will search for entities with "sales" in the name.

You can also list multiple properties and classifications in this argument; for example, entity_name #caption #description, which will cause it to search for entities with sales in their name or in any CAPTION or DESCRIPTION classifications they may have.

Viewing Entity Details

To view details about an entity, click the Actions icon at the right of the entity entry, then click View Details.

For all entities, the details include Lineage and Impact sections. The inclusion of other details, such as Preview and Statistics, varies by entity type.

Preview

Preview displays the data of the entity. For a table, the Preview displays the columns of the table and the data in those columns. You can sort the data in the column into ascending or descending order by clicking the up or down arrow to the right of the column name.

Describe

For an analytic view, the Describe tab has information about the entity, and displays the hierarchies, levels, level depth, dimension tables, level columns, and number of distinct values for a level.

Lineage

Lineage displays all known information about the upstream dependencies of the entity, and therefore how the entity was created and how it is linked to other entities.

For example, for a table that you created in your database, the lineage is just the table. For a table that you created by loading a CSV file from cloud storage, the lineage includes the ingest directive for the data load and the CSV file that is the source of the data.

Pointing to the name of an item in the lineage displays the table name, the application that created it, the type of entity, the path to it, and the schema it is in.

Arrows point from an entity to the entity that it derives from. For example, for a table created in a data load job, an arrow points from the table to the ingest job and an arrow points from the ingest job to the CSV file. If you point to an arrow, then a Links...
Information box appears that shows information about the relationship between the two entities.

To view more details about an item, click the Actions icon for the item, then click **Expand**. For a table, the columns of the table are displayed. Pointing to the name of the table or of a column displays the name, application, type, path, and schema of the table or column. To collapse the display, click the Actions icon, then click **Collapse**.

You can increase or decrease the size of the displayed objects by using the + (plus) and - (minus) keys. You can reposition the objects by grabbing a blank spot in the display and dragging vertically or horizontally.

The lineage for some entities, such as analytic views, is more complex. An analytic view entity deployed for a business model has links to columns in a fact table and to hierarchies. The fact table has links to attribute dimensions and, for a data load job, to an ingest directive for the job. The ingest directive has a link to the source file. The attribute dimensions have links to tables for the dimensions. Those tables have links to ingest directives that have link to source files.

**Impact**

Impact shows all known information about the downstream use of an entity, and therefore how a change in the definition of an entity may affect other entities that depend on it. For example, if a table is used in an analytic view, a change to one of the column definitions in the table may affect the mapping from that column to the analytic view.

**Classifications**

For an analytic view and its attribute dimensions, hierarchies, and measures, the Classifications tab displays classifications and their values. Classifications are metadata that applications can use to present information about analytic views. When creating a business model, you may specify the values for the Caption and Description classifications.

**Optimize**

For an analytic view, the Optimize tab has information about caches created for the analytic view. A cache may exist if the advanced option Enable Autonomous Aggregate Cache was selected for the business model for which the analytic view is deployed.

**Statistics**

Statistics display information about the entity. For example, the statistics for a table include the size of the table and the numbers of rows and columns. They also include the names of the columns, their data types, the number of distinct values and null values, the maximum and minimum values, and other information.

The data is represented in the form of histogram which is column statistic which provides more detailed information about data distribution in a table's columns.

The histograms in the statistics pane can be representative of the following types:

- **Frequency**: In a frequency histogram, each distinct column value corresponds to a single bucket of the histogram. Since each value has its own dedicated bucket, some buckets may have many values, whereas others have few.
- **Top-frequency**: A top frequency histogram is a variation on a frequency histogram that ignores non-popular values that are statistically insignificant.
- **Height-Balanced**: In this histogram, column values are divided into buckets so that each bucket contains approximately the same number of rows.
• Hybrid: A hybrid histogram combines characteristics of both height-based histograms and frequency histograms. This approach enables the optimizer to obtain better selectivity estimates in some situations.

Data Definition

Data Definition displays the Oracle Autonomous Database DDL that created the entity.
The Data Insights Page

The Data Insights page displays information about patterns and anomalies in the data of entities in your Oracle Autonomous Database.

To reach the Data Insights page, click **Data Insights** in the Database Actions page, or click the Selector icon and select **Data Insights** from the Data Tools menu in the navigation pane.

The following topics describe insights and how to generate and use them.

About Insights

You can generate insights for a table or for the analytic view deployed for a business model.

The insights that Data Insights generates for the analytic view of a business model can be more useful than those for a table because of the additional metadata that an analytic view provides.

Insights highlight data points as potentially anomalous if the actual value for a measure when filtering on pairs of analytic view hierarchy values or table column values is considerably higher or lower than the expected value, calculated across all hierarchy or column values. Insights highlight unexpected patterns, which you may want to investigate.

Insights are automatically generated by various analytic functions built into the database. The results of the insight analysis appear as a series of bar charts in the Data Insights dashboard.

Data Insights uses the following steps to generate insights:

1. Finds the values of a measure, for example Sales, across all of the distinct pairs of the hierarchy or column values for the measure. If Sales has the hierarchies or columns Marital Status, Age Band, Income Level, and Gender, then the pairs would be the values of each distinct value of each hierarchy or column paired to each distinct value of each of the other hierarchies or columns. For example, if the values of Marital Status are Married and Single, and the values of Age Band are A, B, and C, then the pairs would be Married and A, Married and B, Married and C, Single and A, Single and B, and Single and C. Each distinct value of Marital Status would also be paired with each distinct value of Income Level and Gender, and so on.

2. Estimates an expected value for the measure for each hierarchy or column pair.

3. Calculates the actual value for the measure for each hierarchy or column pair, for example Marital Status = S, Age Band = C, and then the difference between the actual value and the expected value.

4. Scores all of the differences and selects the largest variations between the actual and expected values to highlight as potential insights.

The resulting insights highlight cases where the measure value is significantly larger or smaller for a given hierarchy or column value pair than expected, for example much higher Sales where Marital Status = S and Age Band = C.
Insights for business models tend to use the higher levels of a hierarchy because the differences between the estimated and actual values are generally larger than they are for lower level attributes. For example, the difference in dollars between the estimated and actual sales for the entire USA are generally larger than the difference between the estimated and actual sales for a town with a population under 1000. The difference is calculated in absolute values, not percentages.

Insights for tables categorize columns as dimension columns or measure columns based on their data types and cardinality. A VARCHAR2 column is always categorized as a dimension, but a NUMBER column may be either a dimension or a measure. For example, a NUMBER column for YEAR values that has only 10 distinct values in a table with 1 million rows is assumed to be a dimension.

Generate Insights and View Reports

Use these procedures to generate Insights and view reports about them.

Generate Insights

To generate insights for a table or business model, do the following:

1. In the Schema field, select a schema.
2. In the Analytic View/Table field, select an analytic view or a table.
3. In the Column field, select a column that contains data about which you want gain insights.
4. Click Search.

A confirmation notice announces that the request for insights has been successfully submitted. Dismiss the notice by clicking the Close (X) icon in the notice.

A progress bar indicates that the search is in progress and when it has completed. The insights appear in the Data Insights dashboard as a series of bar charts.

To refresh the display of the insights, click Refresh. To have refreshes occur automatically, click Enable Auto Refresh.

To see a log of any errors that may have occurred, click View Errors for the desired insights search. The results appear in a new browser tab.

View the Report

The charts in the Data Insights dashboard show the data that contain anomalous results. The bars in a chart show the actual values. The expected values are indicated by green horizontal lines. The bars that are outlined in black contain the most significant differences between the expected and the actual values.

For example, if the fact table for the insights records values about an insurance program, and the measures of the fact table are AGE_CODE, GENDER_CODE, INCOME_CODE, NUM_INSURED, NUM_UNINSURED, and YEAR, then insights might be generated for the NUM_INSURED measure. In that case, the dashboard would have a series of charts labeled YEAR and INCOME_CODE. Each chart would have a value of the related dimension in the upper left corner. For example, an INCOME_CODE chart that has a related AGE_CODE might have the AGE_CODE value 2 in the upper left corner.
Clicking a chart displays more details about it. At the top of the expanded view of the chart is the dimension name and value and a short textual analysis of notable insights. Below the analysis is the chart showing the values and insights about them.

For example, a chart for INCOME_CODE might have at AGE_CODE = 2 at the top, plus the textual analysis. In the chart, the INCOME_CODE values would be on the x-axis and the NUM_INSURED values would be on the y-axis. Pointing to a bar on the expanded chart displays the actual and the expected NUM_INSURED value for that INCOME_CODE and AGE_CODE.

Click the **Back** button to return to the Data Insights dashboard.

**View Previous Reports**

To see the results of a previous search, click the Recent Searches icon at the upper right. In the Recent Searches panel, click anywhere in the box for the insights search that you want to see.

To filter the previous searches, enter a value in the search field at the top of the Recent Searches panel.

To close the Recent Searches panel without selecting a search, click the X at the upper right of the panel.
The Data Analysis Tool

The Data Analysis tool enables you to create Analytic Views with multidimensional metadata. You create Analytic Views on top of a fact table with several dimensions and hierarchies. Analytic views refer to tables in the database and allow users to create hierarchies for dimensions.

Select the Data Analysis card in the Database Actions home page to access this tool. You can also access it by clicking the Selector icon and selecting Data Analysis from the Data Tools menu in the navigation pane.

The Data Analysis page is used to obtain information about the Analytic Views available on your Oracle Autonomous Database.

You can select both hierarchies and measures from Analytic Views. Hierarchies are DB objects that allow users to define relationships between various levels or generations of dimension members. As the name implies, hierarchies organize data using hierarchical relationships. With this tool you can analyze and visualize data in different Points of View (POV). You can export the metadata and visualize it with sophisticated tools like Oracle Analytics Cloud (OAC) and Tableau.

Advantages of Data Analysis tool

With Data Analysis tool you can:

• Visualize, analyze and inspect your data clearly and efficiently with pivot tables
• Calculate total number of errors present in the Analytic View you create and provide solutions to minimize the errors
• Automatically display meaningful insights to help you make better decisions
• Analyze your data across dimensions with support for hierarchical aggregation and drill-down
• Share your Analytic Views with the tool of your choice over various options of raw data consumption to draw meaningful insights and make them accessible to any user.

By identifying relationships among tables and columns, Analytic Views enable your system to optimize queries. They also open new avenues for analyzing data. These avenues include data insights, improved hierarchy navigation, and the addition of hierarchy-aware calculations.

This tool runs complex and hierarchical SQL queries along with SQL extensions in the background, which simplifies real-time calculations. It makes complex data more accessible and easier to understand.

The Data Analysis Page

The following section describes searching and obtaining information about Analytic Views, creating Analytic Views, inspecting your data, discovering insights and visualizing data using tools like Oracle Analytics Cloud (OAC), Tableau, and Microsoft Power BI.

Note:

• OAC has in-built tools to search and utilize Analytic Views.
• We have no direct support for Microsoft Power BI, yet its users can map their tool to the AV transparency views to avail some of the benefits of Analytic Views.

Read these topics for detailed descriptions of the Data Analysis tool features:
• Searching and obtaining information about Analytic Views
• Creating Analytic Views
• Analyzing data

Searching and obtaining information about Analytic Views

When you first open the Data Analysis page, it displays the list of schemas and Analytic Views. With Select Schema, you can select a preferred Schema from a list of schemas available in the drop-down.

The Select Analytic Views drop-down enables you to select an available Analytic View associated with the schema. When you create an Analytic View, it appears in the drop-down option with your schema. The Refresh AV icon refreshes the contents of the selected Analytic View.

The Action icon next to the Refresh AV button enables you to manage Analytic Views. You can Create Analytic View, Edit Analytic View, Compile Analytic View, Show the Data Definition Language (DDL) that generates the Analytic View or Delete Analytic View from the menu.
Obtain information about Analytic Views

By default, Analytic Views are filtered by the current user's schema, as indicated by the schema list below the menu-bar. You can remove the selected schema filter by selecting another user's schema. To search for Analytic Views in other schemas, select one of the schemas from the drop-down.

If there is no Analytic View associated with the schema selected, the tool prompts you to create an Analytic View.

Creating Analytic Views

You can create Analytic Views and view information about them. You can also edit and perform other actions on them.

When you create an Analytic View, you identify a fact table that contains the data to inspect. The Generate Hierarchies and Measures button looks at the contents of that table,
identifies any hierarchies in the fact table, and searches for other tables that may contain related hierarchies.

While creating an Analytic View, you can enable or disable the following advanced options:

- Autonomous Aggregate Cache, which uses the dimensional metadata of the Analytic View to manage a cache and that improves query response time.
- Analytic View Transparency Views, which presents Analytic Views as regular database views and enables you to use your analytic tools of choice while gaining the benefits of Analytic Views.
- Analytic View Base Table Query Transformation, which enables you to use your existing reports and tools without requiring changes to them.

Create Analytic View

To create Analytic View, click Create Analytic View to begin the process.

Click Cancel to cancel the creation of the Analytic View at any time.

Specify Attributes of the Analytic View

On the General tab of the Create Analytic View pane, specify the following:

- The name for the Analytic View
- The fact table for the view
- Advanced options

You can also preview the data of the fact table and see statistics about that data.

In the Name field, specify a name of your choice.

The Schema field has the current user's schema. You can only create an Analytic View in that schema.

In the Fact Table field, expand the drop-down list and click More Sources. The Select Sources dialog box has a list of the available tables and views. Select a table or view from the list.

To filter the list, begin typing characters in the Filter field. As you type, the list changes to show the tables or views that contain the characters. Clear the field to show the complete list again. After you select a table or view, click OK.

To enable or disable the advanced options, on the Create Analytic View pane, click the Show Advanced Options icon at the bottom left. Select or deselect options as desired.

To view the data in the fact table and statistics about the data, click the Preview Data button. In the Preview and Statistics pane, the Preview tab shows the columns of the table and the data in the columns.

The Statistics tab shows the size of the table and the number of rows and columns. The statistics may take a few moments to appear, during which time the message, "No statistics available..." may appear. The statistics include the names of the columns, their data types, the number of distinct values and null values, the maximum and minimum values, and other information. The bar graph displays the top unique column values and the number of their occurrences for the selected column. Point to a bar in the graph to see the number of occurrences of the unique value.
Click **Close** to close the Preview and Statistics pane and return to the Create Analytic View pane.

Click on **Generate Hierarchies and Measures** icon.

The Generating Hierarchies and Measures dialog box displays the progress of searching for dimension tables, analyzing the dimension tables and identifying and creating the data sources, joins, hierarchies, and measures to use. When the process completes, click **Close**.

### Generating Hierarchies and Measures

<table>
<thead>
<tr>
<th>Steps</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Searching for dimension tables</td>
<td>6 Potential dimension tables selected</td>
</tr>
<tr>
<td>Analyzing dimension tables</td>
<td>Complete</td>
</tr>
<tr>
<td>Creating Hierarchies</td>
<td>4 Hierarchies created</td>
</tr>
<tr>
<td>Creating Measures</td>
<td>2 Measures created</td>
</tr>
</tbody>
</table>

The **Search for Dimension Tables** check box when selected, enables you to search for dimension tables while generating hierarchies and measures.

After the hierarchies and measures are generated, they are displayed under their respective tabs. Review the hierarchies and measures created for you.

Specify the **Name, Fact Table** and select **Advanced Options** in the **General** tab of Create Analytic View pane. Click **Create** to generate an Analytic View.

### View Data Sources

The Data Sources tab displays the sources of the data and the relationships among them. It has a graphical display of the fact table and the related dimension tables. For example, a fact table of health insurance data might have columns for geography identifiers, income codes, and gender codes. The Data Sources tab would display items for the fact table and for the geography, income, and gender dimension tables.

You can add hierarchies from data sources even after generating hierarchies from the existing fact table. You can add one or more hierarchies to your new or existing analytic view. Multiple hierarchies can be defined and used in an analytical view, however only one will be used by default.

Right-click the Data Sources tab and select **Add Hierarchy Sources** or select **Add Hierarchy Sources**.

Selecting **Add Hierarchy Sources** launches an **Add Hierarchy Source** dialog box.
You can view all the fact tables and views associated with the analytic view.

In the filter field, you can either manually look for the source or start typing to search for the fact table or views from the list of available fact tables and views. After typing the full name of the source, the tool automatically matches the fact table or view.

Select **Generate and Add hierarchy from Source** to generate analysis and hierarchies associated with the source data you select.

Select **Find and Add Joins** to link all the data sources with the fact table.

Click **OK** to select the source.

The Generating Hierarchies and Measures dialog box displays the progress of analyzing the dimension tables and creating the hierarchies. When the process completes, click **Close**.

**Note:**

When you add a hierarchy from the data source, you see the new hierarchy in the list of hierarchies in the Hierarchies tab. You can navigate between the Data Sources tab, the Hierarchies tab, the Measures tab, the Calculations tab. You can add a hierarchy from a source that is not connected by navigating back to the Data Sources tab.
Select **Remove Hierarchy Source** to remove the hierarchies you create from the data sources. You cannot remove hierarchies generated from the fact table you select from this option.

Expand **Joins** to view the **Hierarchy Source**, **Hierarchy Column** and the **Fact column** mapped with the Analytic View. The **Joins** is visible only when the hierarchy table differs from the fact table.

Expand **Sources** to view the fact table associated with the Analytic View. The data model expands to include the data from the source that you added.

Pointing to an item displays the name, application, type, path and the schema of the table. Click the **Actions** (three vertical dots) icon at the right of the item to display a menu to expand or collapse the view of the table.

An expanded item displays the columns of the table. Pointing to a column displays the name, application, type, path, and schema of the column.
The lines that connect the dimension tables to the fact table indicate the join paths between them. Pointing to a line displays information about the join paths of the links between the tables. If the line connects a table that is collapsed, then the line is dotted. If the line connects two expanded tables, then the line is solid and connects the column in the dimension table to the column in the fact table.

View and Manage Hierarchies

The Hierarchies tab displays the hierarchies generated by the Analytic View creation tool. The display includes the name of the hierarchy and the source table.

An analytic view must include at least one hierarchy.

To add a Hierarchy, click Add Hierarchy. This results in a display as a list of column in that table. Select a column that operates as the detailed level of the hierarchy and be the join-key to the fact table.
To remove the hierarchy, select the hierarchy you want to remove from the list and click **Remove Hierarchy**.

Select **Move Up** or **Move Down** to position the order of the Hierarchy in the resulting view.

Click **Switch Hierarchy to Measure** to change the hierarchy you select to a measure in the Measures list.

You can also **Add Hierarchy** and **Add Hierarchy From Table** by right-clicking the Hierarchy tab.

If you click on a hierarchy name, a dialog box displays the Hierarchy Name and Source.

To change the source, select a different source from the drop-down list.

Select **Add Level** to add a level to the hierarchy. Click **Remove Level** to remove the selected level from the hierarchy.

To change the order of the hierarchy, select the hierarchy you want to move and click on **Move Up** or **Move Down**.
To view the data in the fact table and statistics about the data, click the **Preview Data** button. In the Preview and Statistics pane, the Preview tab shows the columns of the table and the data in the columns. The Statistics tab shows the size of the table and the number of rows and columns.

If you click on a particular level in the Hierarchy tab, a dialog box displays its respective Level Name, Level Key, Alternate Level Key, Member Name, Member Caption, Member Description, source, and Sort By drop-down. To change any of the field values, enter the value in the appropriate field.

Member Captions and Member Descriptions generally represent detailed labels for objects. These are typically end-user-friendly names. For example, you can caption a hierarchy representing geography areas named GEOGRAPHY_HIERARCHY as "Geography" and specify its description as "Geographic areas such as cities, states, and countries."

To see the measures for the Analytic View, click **Measures** tab. To immediately create the Analytic View, click **Create**. To cancel the creation, click **Cancel**.

**View and Manage Measures**

The Measures tab displays the measures suggested for the Analytic View. It displays the Measure Name, Column, and operator Expression for each measure.

The measures specify fact data and the calculations or other operations to perform on the data.

To add measures, click **Add Measure**. You can view a new measure at the bottom of the measures list. To remove the measure, select the measure you want to remove from the list and click **Remove Measure**.
To alternatively add a measure from the data source, right-click the Measures tab. This pops up a list of columns that can be used as measures. Select one measure from the list.

You can exclude a column from the measures on right-clicking the Measures tab and selecting Remove Measure.

Click **Switch Measure to Hierarchy** to change the measure you select to hierarchy in the Hierarchies list.
You must specify a measure as the default measure for the analytic view; otherwise, the first measure in the definition is the default. Select Default Measure from the drop-down.

To add a measure, right-click the Measures tab and select Add Measure. To remove a measure, select the particular measure you want to remove, right-click on it and select Remove Measure.

You can select a different column for a measure from the Column drop-down list. You can select a different operator from the Expression drop-down list.

In creating an analytic view, you must specify one or more hierarchies and a fact table that has at least one measure column and a column to join to each of the dimension tables outside of the fact table.

Create new calculated measures

You can add measure calculations to a query of an analytic view.

The measures and hierarchies associated with the analytic views enable us to create new calculated measures.

Calculated measures return values from data stored in one or more measures. You compute these measures at run time.

**Note:**

You can create the measures without increasing the size of the database since the calculated measures do not store the data. However, they may slow performance. You need to decide which measures to calculate on demand.

The Analytic Views provides easy-to-use templates for creating calculated measures.

Once you create a calculated measure, it appears in the list of measures of the Analytic View. You can create a calculated measure at any time which is available for querying in SQL.

The Data Analysis tool provides easy-to-use templates for creating calculated measures.
Click **Add Calculated Measure** to add calculations to the measures. You can view the new calculation with system generated name in the **Calculations** tab.

Click the newly created calculated measure.

In the **Measure Name** field, enter the name of the calculated measure.

You can select preferred category of calculation from a list of options such as Prior and Future Period, Cumulative Aggregates, Period To Date, Parallel Period, Moving Aggregates, Share, Qualified Data Reference, and Ranking using the **Calculation Category** drop-down.

Your choice of category of calculation dynamically changes the **Calculation Template**.

For more details on how to use Calculation templates, see Using Calculation Templates.

Select the **Measure** and **Hierarchy** on which you want to base the calculated measures.

Select **Offset** value by clicking the up or the down arrow. The number specifies the number of members to move either forward or backward from the current member. The ordering of members within a level is dependent on the definition of the attribute dimension used by the hierarchy. The default value is 0 which represents POSITION FROM BEGINNING.

The Expression field lists the expressions which the calculated measure uses.

On the creation of the Analytic view, the calculated measure appears in the navigation tree in the Calculated Measures folder.

Click **Create**. A confirmation dialog box appears that asks for your confirmation. Select **Yes** to proceed with the creation of Analytic View.

After creating the Analytic View, you will view a success message informing you of its creation.

On editing the Analytic View you create, you can view the calculated measure in the navigation tree in the Calculations folder.

Click the **Tour** icon for a guided tour of the worksheet highlighting salient features and providing information if you are new to the interface.

Click the **help** icon to open the contextual or online help for the page you are viewing.

Click **Show DDL** to generate Data Definition Language statements for the analytic view.
Edit Analytic View

You might want to edit an Analytic View to make changes to the data sources, the hierarchies, or the measures.

To edit an Analytic View, click the Action icon on the Analytic View item, then click Edit Analytic View. On the Edit Analytic View screen, select a tab and make changes as desired.

When you have completed the changes, click Update.

Using Calculation Templates

The Data Analysis tool provides templates for all of the calculations typically in demand for business intelligence applications.

The following topics describe the types of calculations available as calculation templates in the tool.

- Cumulative Aggregates
- Prior and Future Period
- Period to Date
- Parallel Period
- Moving Aggregates
- Share
- Rank

Cumulative Aggregates

Cumulative calculations start with the first time period and calculate up to the current member, or start with the last time period and calculate back to the current member.

The tool provides several aggregation methods for cumulative calculations:

- **Cumulative Average**: Calculates a running average across time periods.
- **Cumulative Maximum**: Calculates the maximum value across time periods.
- **Cumulative Minimum**: Calculates the minimum value across time periods.
- **Cumulative Total**: Calculates a running total across time periods.

You can choose the measure, the time dimension, and the hierarchy. For selecting the time range see "Choosing a Range of Time Periods" in Oracle OLAP User’s Guide.

Cumulative Calculation Example

This template defines a calculated measure using Cumulative Minimum.

Cumulative minimum of SALES in the TIME dimension and TIME.CALENDAR hierarchy within ancestor at level TIME.CALENDAR_YEAR. Total from beginning to current member.

These are the results of a query against the calculated measure, which displays values for the descendants of calendar year 2021. The minimum value for quarters
begins with Q1-21 and ends with Q4-21, and for months begins with Jan-21 and ends with Dec-21.

<table>
<thead>
<tr>
<th>TIME</th>
<th>TIME_LEVEL</th>
<th>SALES</th>
<th>MIN_SALES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1.21</td>
<td>CALENDAR_QUARTER</td>
<td>32977874</td>
<td>32977874</td>
</tr>
<tr>
<td>Q2.21</td>
<td>CALENDAR_QUARTER</td>
<td>35797921</td>
<td>32977874</td>
</tr>
<tr>
<td>Q3.21</td>
<td>CALENDAR_QUARTER</td>
<td>33526203</td>
<td>32977874</td>
</tr>
<tr>
<td>Q4.21</td>
<td>CALENDAR_QUARTER</td>
<td>41988687</td>
<td>32977874</td>
</tr>
<tr>
<td>JAN-21</td>
<td>MONTH</td>
<td>11477898</td>
<td>11477898</td>
</tr>
<tr>
<td>FEB-21</td>
<td>MONTH</td>
<td>10982016</td>
<td>10982016</td>
</tr>
<tr>
<td>MAR-21</td>
<td>MONTH</td>
<td>10517960</td>
<td>10517960</td>
</tr>
<tr>
<td>APR-21</td>
<td>MONTH</td>
<td>11032057</td>
<td>10517960</td>
</tr>
<tr>
<td>MAY-21</td>
<td>MONTH</td>
<td>11432616</td>
<td>10517960</td>
</tr>
<tr>
<td>JUN-21</td>
<td>MONTH</td>
<td>13333248</td>
<td>10517960</td>
</tr>
<tr>
<td>JUL-21</td>
<td>MONTH</td>
<td>12070352</td>
<td>10517960</td>
</tr>
<tr>
<td>AUG-21</td>
<td>MONTH</td>
<td>11108893</td>
<td>10517960</td>
</tr>
<tr>
<td>SEP-21</td>
<td>MONTH</td>
<td>10346958</td>
<td>10346958</td>
</tr>
<tr>
<td>OCT-21</td>
<td>MONTH</td>
<td>14358605</td>
<td>10346958</td>
</tr>
<tr>
<td>NOV-21</td>
<td>MONTH</td>
<td>12757560</td>
<td>10346958</td>
</tr>
<tr>
<td>DEC-21</td>
<td>MONTH</td>
<td>14872522</td>
<td>10346958</td>
</tr>
</tbody>
</table>

Prior and Future Period

The Data Analysis tool provides several calculations for prior or future time periods. Here are the calculations used for for prior or future time periods:

- **Prior Period**: Returns the value of a measure at an earlier time period.
- **Difference From Prior Period**: Calculates the difference between values for the current time period and an earlier period.
- **Percent Difference From Prior Period**: Calculates the percent difference between the values for the current time period and an earlier period.
- **Future Period**: Returns the value of a measure at a later time period.
- **Difference From Future Period**: Calculates the difference between the values for the current time period and a later period.
- **Percent Difference From Future Period**: Calculates the percent difference between the values for the current time period and a later period.

When creating a calculation for prior or future time periods, you choose the measure, the time dimension, the hierarchy, and the number of periods from the current period.

**Prior Period Example**

This template defines a calculated measure using Prior Period:

Prior period for measure SALES in TIME dimension and TIME.CALENDAR hierarchy 1 period ago.

These are the results of a query against the calculated measure. The PRIOR_PERIOD column shows the value of Sales for the preceding period at the same level in the Calendar hierarchy.

<table>
<thead>
<tr>
<th>TIME</th>
<th>TIME_LEVEL</th>
<th>SALES</th>
<th>PRIOR_PERIOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>2020</td>
<td>CALENDAR_YEAR</td>
<td>136986572</td>
<td>144290686</td>
</tr>
<tr>
<td>2021</td>
<td>CALENDAR_YEAR</td>
<td>140138317</td>
<td>136986572</td>
</tr>
<tr>
<td>Q1.20</td>
<td>CALENDAR_QUARTER</td>
<td>31381338</td>
<td>41988687</td>
</tr>
</tbody>
</table>
Period to Date

Period-to-date functions perform a calculation over time periods with the same parent up to the current period.

These functions calculate period-to-date:

- **Period to Date**: Calculates the values up to the current time period.
- **Period to Date Period Ago**: Calculates the data values up to a prior time period.
- **Difference From Period to Date Period Ago**: Calculates the difference in data values up to the current time period compared to the same calculation up to a prior period.
- **Percent Difference From Period To Date Period Ago**: Calculates the percent difference in data values up to the current time period compared to the same calculation up to a prior period.

When creating a period-to-date calculation, you can choose from these aggregation methods:

- Sum
- Average
- Maximum
- Minimum

You also choose the measure, the time dimension, and the hierarchy.

**Period to Date Example**

This template defines a calculated measure using Period to Date.

Gregorian Year to date for SALES in the TIME dimension and TIME.CALENDAR hierarchy. Aggregate using MINIMUM from the beginning of the period.

These are the results of a query against the calculated measure. The MIN_TO_DATE column displays the current minimum SALES value within the current level and year.

<table>
<thead>
<tr>
<th>TIME</th>
<th>TIME_LEVEL</th>
<th>SALES</th>
<th>MIN_TO_DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1.21</td>
<td>CALENDAR_QUARTER</td>
<td>36154815</td>
<td>36154815</td>
</tr>
<tr>
<td>Q2.21</td>
<td>CALENDAR_QUARTER</td>
<td>36815657</td>
<td>36154815</td>
</tr>
<tr>
<td>Q3.21</td>
<td>CALENDAR_QUARTER</td>
<td>32318935</td>
<td>32318935</td>
</tr>
<tr>
<td>Q4.21</td>
<td>CALENDAR_QUARTER</td>
<td>34848911</td>
<td>32318935</td>
</tr>
<tr>
<td>JAN-21</td>
<td>MONTH</td>
<td>13119235</td>
<td>13119235</td>
</tr>
<tr>
<td>FEB-21</td>
<td>MONTH</td>
<td>11441738</td>
<td>11441738</td>
</tr>
<tr>
<td>MAR-21</td>
<td>MONTH</td>
<td>11593842</td>
<td>11441738</td>
</tr>
<tr>
<td>APR-21</td>
<td>MONTH</td>
<td>11356940</td>
<td>11356940</td>
</tr>
<tr>
<td>MAY-21</td>
<td>MONTH</td>
<td>13820218</td>
<td>11356940</td>
</tr>
<tr>
<td>JUN-21</td>
<td>MONTH</td>
<td>11638499</td>
<td>11356940</td>
</tr>
<tr>
<td>JUL-21</td>
<td>MONTH</td>
<td>9417316</td>
<td>9417316</td>
</tr>
</tbody>
</table>
Parallel Period

Parallel periods are at the same level as the current time period, but have different parents in an earlier period. For example, you may want to compare current sales with sales for the prior year at the quarter and month levels.

The Data Analysis tool provides several functions for parallel periods:

- **Parallel Period**: Calculates the value of the parallel period.
- **Difference From Parallel Period**: Calculates the difference in values between the current period and the parallel period.
- **Percent Difference From Parallel Period**: Calculates the percent difference in values between the current period and the parallel period.

To identify the parallel period, you specify a level and the number of periods before the current period. You can also decide what happens when two periods do not exactly match, such as comparing daily sales for February (28 days) with January (31 days).

You also choose the measure, the time dimension, and the hierarchy.

Parallel Period Example

This template defines a calculated measure using Parallel Period.

Parallel period for **SALES** in the **TIME** dimension and **TIME.CALENDAR** hierarchy 1 **TIME.CALENDAR.QUARTER** ago based on position from **beginning to ending** of period.

These are the results of a query against the calculated measure, which lists the months for two calendar quarters. The parallel month has the same position within the previous quarter. The prior period for **JUL-21** is **APR-21**, for **AUG-21** is **MAY-21**, and for **SEP-21** is **JUN-21**.

<table>
<thead>
<tr>
<th>TIME</th>
<th>PARENT</th>
<th>SALES</th>
<th>LAST_QTR</th>
</tr>
</thead>
<tbody>
<tr>
<td>APR-21</td>
<td>CY2006.Q2</td>
<td>11356940</td>
<td>13119235</td>
</tr>
<tr>
<td>MAY-21</td>
<td>CY2006.Q2</td>
<td>13820218</td>
<td>11441738</td>
</tr>
<tr>
<td>JUN-21</td>
<td>CY2006.Q2</td>
<td>11638499</td>
<td>11593842</td>
</tr>
<tr>
<td>JUL-21</td>
<td>CY2006.Q3</td>
<td>9417316</td>
<td>11356940</td>
</tr>
<tr>
<td>AUG-21</td>
<td>CY2006.Q3</td>
<td>11596052</td>
<td>13820218</td>
</tr>
<tr>
<td>SEP-21</td>
<td>CY2006.Q3</td>
<td>11305567</td>
<td>11638499</td>
</tr>
</tbody>
</table>

Moving Aggregates

Moving aggregates are performed over the time periods surrounding the current period.

The Data Analysis tool provides several aggregation methods for moving calculations:

- **Moving Average**: Calculates the average value for a measure over a fixed number of time periods.
- **Moving Maximum**: Calculates the maximum value for a measure over a fixed number of time periods.
• **Moving Minimum**: Calculates the minimum value for a measure over a fixed number of time periods.

• **Moving Total**: Returns the total value for a measure over a fixed number of time periods.

You can choose the measure, the time dimension, and the hierarchy. You can also select the range, as described in "Choosing a range of time periods" in *Oracle OLAP User's Guide*, and the number of time periods before and after the current period to include in the calculation.

**Moving Aggregates Example**

This template defines a calculated measure using Moving Minimum.

Moving minimum of **SALES** in the **TIME** dimension and **TIME.CALENDAR** hierarchy. Include 1 preceding and 1 following members within **level**.

These are the results of a query against the calculated measure, which displays values for the descendants of calendar year 2021. Each value of Minimum Sales is the smallest among the current value and the values immediately before and after it. The calculation is performed over all members of a level in the cube.

<table>
<thead>
<tr>
<th>TIME</th>
<th>TIME_LEVEL</th>
<th>SALES</th>
<th>MIN_SALES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1.21</td>
<td>CALENDAR_QUARTER</td>
<td>32977874</td>
<td>32977874</td>
</tr>
<tr>
<td>Q2.21</td>
<td>CALENDAR_QUARTER</td>
<td>35797921</td>
<td>32977874</td>
</tr>
<tr>
<td>Q3.21</td>
<td>CALENDAR_QUARTER</td>
<td>33526203</td>
<td>33526203</td>
</tr>
<tr>
<td>Q4.21</td>
<td>CALENDAR_QUARTER</td>
<td>41988687</td>
<td>31381338</td>
</tr>
<tr>
<td>JAN-21</td>
<td>MONTH</td>
<td>11477898</td>
<td>10982016</td>
</tr>
<tr>
<td>FEB-21</td>
<td>MONTH</td>
<td>10982016</td>
<td>10517960</td>
</tr>
<tr>
<td>MAR-21</td>
<td>MONTH</td>
<td>10517960</td>
<td>10517960</td>
</tr>
<tr>
<td>APR-21</td>
<td>MONTH</td>
<td>11032057</td>
<td>10517960</td>
</tr>
<tr>
<td>MAY-21</td>
<td>MONTH</td>
<td>11432616</td>
<td>11032057</td>
</tr>
<tr>
<td>JUN-21</td>
<td>MONTH</td>
<td>13333248</td>
<td>11432616</td>
</tr>
<tr>
<td>JUL-21</td>
<td>MONTH</td>
<td>12070352</td>
<td>11108893</td>
</tr>
<tr>
<td>AUG-21</td>
<td>MONTH</td>
<td>11108893</td>
<td>10346958</td>
</tr>
<tr>
<td>SEP-21</td>
<td>MONTH</td>
<td>10346958</td>
<td>10346958</td>
</tr>
<tr>
<td>OCT-21</td>
<td>MONTH</td>
<td>14358605</td>
<td>10346958</td>
</tr>
<tr>
<td>NOV-21</td>
<td>MONTH</td>
<td>12757560</td>
<td>12757560</td>
</tr>
<tr>
<td>DEC-21</td>
<td>MONTH</td>
<td>14872522</td>
<td>12093518</td>
</tr>
</tbody>
</table>

**Share**

Share calculates the ratio of a measure's value for the current dimension member to the value for a related member of the same dimension.

You can choose whether the related member is:

• **Top of hierarchy**: Calculates the ratio of each member to the total.

• **Member’s parent**: Calculates the ratio of each member to its parent.

• **Member’s ancestor at level**: Calculates the ratio of each member to its ancestor, that is, a member at a specified level higher in the hierarchy.

When creating a share calculation, you can choose the measure, dimension, and hierarchy. You also have the option of multiplying the results by 100 to get percentages instead of fractions.
Share Example

This template defines a calculated measure using \textit{SHARE}:

Share of measure \textit{SALES} in \textit{PRODUCT.PRIMARY} hierarchy of the \textit{PRODUCT} dimension as a ratio of \textit{top of hierarchy}.

These are the results of a query against the calculated measure. The \textit{TOTAL\_SHARE} column displays the percent share of the total for the selected products.

<table>
<thead>
<tr>
<th>PRODUCT</th>
<th>PROD_LEVEL</th>
<th>SALES</th>
<th>TOTAL_SHARE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Product</td>
<td>TOTAL</td>
<td>144290686</td>
<td>100</td>
</tr>
<tr>
<td>Hardware</td>
<td>CLASS</td>
<td>130145388</td>
<td>90</td>
</tr>
<tr>
<td>Desktop PCs</td>
<td>FAMILY</td>
<td>78770152</td>
<td>55</td>
</tr>
<tr>
<td>Portable PCs</td>
<td>FAMILY</td>
<td>19066575</td>
<td>13</td>
</tr>
<tr>
<td>CD/DVD</td>
<td>FAMILY</td>
<td>16559860</td>
<td>11</td>
</tr>
<tr>
<td>Software/Other</td>
<td>CLASS</td>
<td>14145298</td>
<td>10</td>
</tr>
<tr>
<td>Accessories</td>
<td>FAMILY</td>
<td>6475353</td>
<td>4</td>
</tr>
<tr>
<td>Operating Systems</td>
<td>FAMILY</td>
<td>5738775</td>
<td>4</td>
</tr>
<tr>
<td>Memory</td>
<td>FAMILY</td>
<td>5430466</td>
<td>4</td>
</tr>
<tr>
<td>Modems/Fax</td>
<td>FAMILY</td>
<td>5844185</td>
<td>4</td>
</tr>
<tr>
<td>Monitors</td>
<td>FAMILY</td>
<td>4474150</td>
<td>3</td>
</tr>
<tr>
<td>Documentation</td>
<td>FAMILY</td>
<td>1931170</td>
<td>1</td>
</tr>
</tbody>
</table>

Rank

Rank orders the values of a dimension based on the values of the selected measure. When defining a rank calculation, you choose the dimension, a hierarchy, and the measure.

You can choose a method for handling identical values:

- **Rank**: Assigns the same rank to identical values, so there may be fewer ranks than there are members. For example, it may return 1, 2, 3, 3, 4 for a series of five dimension members.
- **Dense Rank**: Assigns the same minimum rank to identical values. For example, it may return 1, 2, 3, 3, 5 for a series of five dimension members.
- **Average Rank**: Assigns the same average rank to identical values. For example, it may return 1, 2, 3.5, 3.5, 5 for a series of five dimension members.

You can also choose the group in which the dimension members are ranked:

- **Member's level**: Ranks members at the same level.
- **Member's parent**: Ranks members with the same parent.
- **Member's ancestor at level**: Ranks members with the same ancestor at a specified level higher in the hierarchy.

Rank Example

This template defines a calculated measure using Rank:

Rank members of the \textit{PRODUCT} dimension and \textit{PRODUCT.PRIMARY} hierarchy based on measure \textit{SALES}. Calculate rank using \textit{RANK} method with \textit{member's parent} in order \textit{lowest to highest}. Rank NA (null) values \textit{nulls last}.

These are the results of a query against the calculated measure in which the products are ordered by \textit{RANK}:
## Analyzing data

The output display pane for an Analytic View enables you to Analyze the data from Analytic View.

There are three tabs displayed in the output display pane for an Analytic View. They are the following:

- **Analyze** tab- In this tab, you select **Hierarchy Levels** and **Measures** from the Analytic View to analyze the data.
- **Data Quality** tab- This tab displays structural errors in the table data and dimension sources for the Analytic View.
- **Export** tab- This tab exports the Analytic View you create to different tools available for better visualization.

### Analyze tab

The Analytic View Browser displays the Hierarchies, Levels and calculated measures generated with the selected Analytic View. You can drag and drop Levels and Measures from the Analytic View browser to the drop area in Columns, Rows or Values and Filters in the output pane of Analyze tab. The Rows and Columns are interchangeable. The values should be Analytic View measures.

Drag and drop measures to the filter area in the **Analyze** tab. After you drop the selected measure to the drop area, a faceted search window appears.
You can analyze filtered and refined searches in the output by dragging and dropping multiple measures in the filter area. It helps you drill down to the values you are looking for.

Choose one of the two options in the top-right of the dialog box to set the mode of display on the window.

Click **List View** to display the filter condition in two columns. The left column is the search column in which you can drill down to the specific value in the measures. You can select multiple values from the complete list of values displayed in the left column. The right column shows the values you have selected from the list.

Click **Multi-Select** view to display the search condition as a single column. Click **Add Filter** to view the complete list of values. You can select multiple values from this list.

Click the **display or hide funnel** icon to open faceted search list.

The Faceted search list consists of measures and the count next to each value displays the total number of items that fit the category. You can click on attributes of the measures that apply as a filter to your search that further narrows the search down.

Choose one of the three options in the top-right of the dialog box to view the result of the query in the appropriate mode. The modes of display are Table view, Pivot view and Chart view.

Click the **SQL** icon on the display area to view the SQL query of the filter condition. The worksheet editor page appears.
Click the Run Query icon to execute the statements in the worksheet editor. The area beneath the SQL worksheet editor displays the results of the search condition that match the filter criteria.

After creating an Analytic View, you can use it to analyze its data in different formats. You can choose from different visualization formats such as pivot, table or chart. These are generated based on the levels and measures you select. You can analyze the data in pivot, table and chart forms by selecting their respective icons displayed in the right corner. These are generated based on the levels and measures you select.

You can add totals to the pivot table data. This Total value is created as an additional row and an additional column. The tool automatically calculates the aggregate value based on the values, rows, and columns you drop from the Analytic View browser to the drop area.

The grand total values are displayed at the end of the total row and total column. Based on the grouping of data, the grand total values are further broken down into sub-total values.

This helps in comparing values and summarizing data which further helps to gain insights from the different sets of data you select.

The Related Insights panel displays the generated graphically based on measures selected to visualize the data. It is designed in such a way that it could trigger immediate insights.
Data Quality tab

This tab helps you measure the total number of structural errors in the Analytic View you create. For example, null values in a join column is an error. In addition to being a join column, this error will be signaled if there is a null value in the detail level of a hierarchy. This way you can resolve the errors and enhance the quality of the data. This tab displays the fact table and dimensions along with the associated number of errors. Click the triangular icon on the box with errors to expand the list to display first five rows with that specific error as a sample. The errors are categorized based on dimensions and fact tables.

Click the Download icon to view all errors as a CSV file.

Select the cards to view which category the errors belong to. Select Analyze icon to switch the view back to Analyze tab.

Export tab

This tab exports the data to tools like Tableau.

---

**Note:**

- OAC has in-built tools to search and utilize Analytic Views.
- We have no direct support for Microsoft Power BI, yet its users can map their tool to the AV transparency views to avail some of the benefits of Analytic Views.

This allows anyone to connect to data, visualize and create interactive dashboards in a few clicks.
Oracle Autonomous Database Add-in for Excel

The Oracle Autonomous Database Add-in for Excel integrates Microsoft Excel spreadsheets with the Autonomous Database to retrieve and analyze data from Analytic Views in the database. You can run custom SQL queries and view their results in the worksheet.

Install the Oracle Autonomous Database add-in on Mac

The Oracle Autonomous Database add-in for Excel is supported on Mac OS running Microsoft Office 365.

Before you can install the Autonomous Database Add-in for Excel, download the oracleplugin.zip file and extract it to get the install.cmd script file from your Database Actions instance.

- Open the Database Actions Launchpad.
- On the right of the launchpad page, in the Getting Started panel, click the Download link present below the Add-in for Excel Spreadsheet link.
You can now view a zipped folder in the Downloads folder of your system.

Create a new folder named **Add-in** in your Downloads folder.

Extract the zipped folder in the **Add-in** folder.

Follow these steps to install the add-in.

1. Quit Excel before you run the installer.
2. Click the Launchpad icon in the Dock of your system, type Terminal in the search field, then click **Terminal**.
3. In the Terminal application on your Mac, use the `cd` command to move into the directory that contains the file you want to make executable.

   ```
cd ~/Downloads/Manifest/
   
4. Enter the `chmod u+x` command to add execution privilege to `install.sh` file.

   ```
   chmod u+x ./install.sh
   
5. After making the script file executable, you can run it by entering its pathname.

   ```
   cd ~/Downloads/Manifest/
   
   ./install.sh
   
6. Start Excel and open a new or existing workbook.

7. From the **Insert** menu in the Excel ribbon, select drop-down menu of **My Add-ins**.
A new Oracle Autonomous Database entry appears under Developer Add-Ins dialog box.

8. Select Oracle Autonomous Database.

A new Autonomous Database ribbon tab appears in MS Excel.

Install the Oracle Autonomous Database add-in on Windows

The Oracle Autonomous Database Add-in for Excel is supported on Windows 10 operating systems running Microsoft Excel 365.

Before you can install the Autonomous Database Add-in for Excel, download the oracleplugin.zip file and extract it to get the install.cmd script file from your Database Actions instance.

- Open the Database Actions Launchpad.
- On the right of the launchpad page, in the Getting Started panel, click the Download link present below the Add-in for Excel Spreadsheet link.
You can now view a zipped folder in the Downloads folder of your system.

Create a new folder named **Add-in** in your Downloads folder.

Extract the zipped folder in the **Add-in** folder.

Follow these steps to install the add-in.

1. Quit Excel before you run the installer.
2. Double-click the **install.cmd** file that you downloaded previously in the **Add-in** folder. The installer automatically creates a manifest folder at %LOCALAPPDATA%\Oracle\Autonomous Database\manifest\.

**Note:**

After running the installer on Windows, the add-in automatically creates a network share and adds the shared location as a trusted catalog location for Office add-ins. A catalog is used to store the manifest for the Excel Add-in. It enables publishing and management of the Excel add-in as well as other add-ins that are available in the Office Store and licensed for corporate use. You can acquire the Excel add-in by specifying the shared manifest folder as a trusted catalog.

3. Start Excel and open a new or existing workbook.
4. From the **Insert** menu in the Excel ribbon, select **My Add-ins**.
   A new **Oracle Autonomous Database** folder appears under **Shared folder** tab of Office add-ins dialog box.
5. Select the folder and click **Add**.
   After you install the add-in, a new **Autonomous Database** ribbon tab appears in MS Excel.

**Note:**

You can change the functionality of the installer after initial installation. Re-run the installer and choose the option of your preference. You can either choose to repair your existing installation by deleting it and selecting the installed trusted catalog or adding another manifest to the working installation.
Uninstall the Oracle Autonomous Database add-in

The following section describes the steps to uninstall the Oracle Autonomous Database add-in.

To uninstall the Oracle Autonomous Database Add-in for Excel for Windows:

- Delete the `manifest.xml` file from the Add-in folder in your Downloads located on your PC.
- Click Refresh in the Office Add-ins window to remove the Autonomous Database tab from MS Excel.

No add-ins will now be available from the Shared folder of the Office Add-ins window.

To uninstall the Oracle Autonomous Database Add-in for Excel for Mac:

- Enter the following command in the terminal to remove the `manifest.xml` file.

  ```bash
  rm ~/Library/Containers/com.microsoft.Excel/Data/Documents/wef/manifest.xml
  ```

The Oracle Autonomous Database Add-in is uninstalled from Excel for Mac.

**Note:**

After uninstalling the Add-in, if you re-install it from different Autonomous Database (ADB) then the add-in attempts to load the old ADB. You need to then check if the location (share path) of the shared manifest folder is pointing to the right location. Refer to Configuring the Excel Trusted Add-in Catalog in FAQs for Troubleshooting errors with Excel Add-in chapter for more details.

Using Oracle Autonomous Database Add-in for Excel

After you install the add-in, a new ribbon tab, Autonomous Database appears in MS Excel.

You can sign in to the Autonomous Database, work with Analytic Views and view the data in the worksheet.
This ribbon provides buttons that lets you connect to the Autonomous Database.

The Sign In icon controls access to the database. Click Sign In to integrate the Autonomous Database with the Excel worksheet. After you sign in, the About icon, the Native SQL icon, and Query Wizard icons are enabled.

Click About to view the Add-in and the supported excel versions. The About window also displays whether the spreadsheet is connected to the database or not.

Click Native SQL to write and run custom SQL queries.

Click Query Wizard to select the Analytic View you want to query. You can review and edit the query, add or edit filters and calculations to it and choose the output format from tabular and pivot formats.

Selecting Native SQL icon or Query Wizard icon from the ribbon launches the Oracle Autonomous Database wizard in the Excel task pane.
Click **Move** in the drop-down of the wizard pane to move the wizard to your preferred location.

The **Resize** option in the drop-down resizes the query window. As you select this option, you can resize the wizard window by moving the double-headed arrow sideways. The wizard expands when you move the arrow to the left and it contracts when you drag it to the right.

Click **Close** to close the wizard.

Run native SQL queries in an Excel worksheet

The Oracle Autonomous Database add-in for Excel lets you run native SQL queries to work with your data in an Excel worksheet.

With the add-in, you can create a table and insert, update and delete rows from the existing table. You can view the results in the existing worksheet or in a different worksheet.

The following image shows your data retrieved from the Autonomous Database and displayed in the worksheet. The Query Info section comprising the Timestamp, User name and SQL Query are shown in Excel. You can edit custom queries, run them, query Analytic Views, and add or edit filters to the Analytic View. The worksheet displays the results of queries in tabular format, with an option to add a worksheet with an excel pivot table from the retrieved data.

The add-in maintains a live connection with the database. However, the data retrieved is local to Excel. In case of inactivity, the connection times out, and you will need to log in again. The image shows the results from a single query, but you can insert many queries in a single worksheet.

To run a query using the add-in, run Excel and create a blank workbook using the standard Excel workbook file format.

1. In the Excel ribbon, select the **Autonomous Database**.
2. Click the **Native SQL** icon from the ribbon. This opens an Oracle Autonomous Database dialog box in the Excel Task Pane.
3. Write a query in the SQL query editor.
4. Click **Execute** to run the query results in the worksheet. You can click + sign besides the **Select worksheet** drop-down to display the results in a new worksheet.

5. The worksheet also displays information such as the timestamp and the user who creates and runs the query. To run another query follow these steps:

1. Clear the previous query from the SQL editor and write the new query.

2. In the **Select worksheet** drop-down, select a new sheet, *Sheet 2* in this case. The Add-in adds a sheet for the user. If you choose to work on the same sheet, the Add-in refreshes the data in the existing worksheet.

3. Click **Execute** to display the query results.

The worksheet displays the result of the query at a go. While this behavior works for most scenarios, sometimes, for large data sets, the query result might exceed 10K rows. Although you can view the 10K rows, a confirmation window appears which asks if you would want to view the rest of the result.

Select **Yes** to view the entire result set. Loading the entire data may take a while. You must fetch the entire query result before working with Pivot tables, or else it will lead to incorrect results from aggregation in Pivot tables.

Close the Query Wizard panel to cancel the operation of fetching the result.

---

**Note:**

Close the Query Wizard panel to cancel the operation of fetching the result.

---

**Query an Analytic View in an Excel worksheet**

The Query Wizard menu enables you to make a copy of the data you select from the Autonomous Database. Once the wizard copies and retrieves the data, it becomes local to Excel. You can further edit the data in Excel but cannot write back to the Autonomous Database. The add-in lets you modify an existing Analytic View by using the Query Wizard icon.

You can retrieve the Analytic View and manipulate the query according to your requirements to visualize the result data in the worksheet. You can search for the Analytic View, select measures, hierarchies, and levels from the query. You also have options to add filters and calculated measures to the query and view the query result in the spreadsheet.

By default, the data is retrieved in tabular format. You also have the option to create an Excel pivot from this data.

The Query Wizard has three panels:

1. **Analytic View panel:** The Analytic View panel contains a list of analytic views from which you build queries. You edit the query by selecting measures, hierarchies and levels and progress to the next panel.
2. **Filter panel:** The Filter panel displays to the right of the Analytic View panel. You can create filter conditions and add calculations to the query in this panel.
Add/Edit filter conditions

- Column Name:
  - TIME_HIER: YEAR

- Attribute Name:
  - YEAR_NAME

- Operator:
  - >

- Values (separated):
  - 30000

Add filter

Add/Edit calculations

Previous  Next
3. **Query Result panel:** The Query Result panel displays to the right of the Filter panel. Once you select the filter criteria and determine what calculated measures to add to your query, you run the query. You can view and revise the SQL query. After the SQL query runs, you view the query results in the worksheet. You can select the output format of the result here. You can view the results in tabular format or the form of a Pivot table.
To query an analytic view and explore the Query Wizard menu in the MS Excel ribbon:

1. On the ribbon, select the Query Wizard icon.
2. Selecting the Query Wizard opens an Oracle Autonomous Database dialog box in the Excel Task Pane.
3. Select an existing Analytic View from the drop-down in the Analytic View pane. As you select the Analytic View, it appears on the Analytic View field.
4. Select your choice of measures, hierarchies, and levels from list of available measures, hierarchies and levels associated with the Analytic View. Click Next.
5. The wizard window progresses to the **Filter** pane where you can add or edit filters to query.

6. **Under Add or Edit filter conditions, do the following.**

   - Select the column name and the attribute name from the drop-down. The values of the attribute change dynamically with the change in column names.
   - Select an operator in the Operator field to apply to the values that you specify in the Value field.
   - Specify a value or values from the list that contains members of the column that you select. You need to enter the value into the Values field manually. For example, you can select > in the Operator field to use only values greater than the value that you select in the Value list. If you select 100,000 from the Value list, then the filter uses values from the column that are greater than 100,000. You can use this information in an analysis to focus on products that are performing well.
     - Click **Add Filter** to add another filter condition.

7. **Under Add or Edit Calculations, do the following.**

   - Specify the column whose values you want to include in the group or calculated item.
   - On the Calc expression field, enter a custom calculated expression you want to perform on the column value. You can add functions or conditional expressions.

8. Click **Next** to progress to the **Query Result**.

9. You can view, edit, and review the query you have generated from the Query Review editor.

10. Select **Remove empty columns** to remove empty columns from the result.

11. Select **Column per level** to retrieve all levels of a hierarchy in a single column.

12. Select the worksheet from the drop-down where you want to view the result.

13. Click **Execute** to run the query.

14. You can view the result of the query in the worksheet you select.

15. You can always modify the query in the Oracle Autonomous Database dialog box editor even after results are generated.

16. Select **Table** in the Query Result pane to view the results in the worksheet in a tabular format.

17. Select **Pivot** in the Query Result pane to view the results in the worksheet in Pivot format.

**View the results in Pivot tables**

Pivot table view is an interactive view that allows you to transpose rows and columns. A pivot table can summarize, sort, reorganize, count total and perform average of the result data. They are navigable and drillable.

Apart from tabular mode, to view the query results in pivot table mode, select the **Pivot Table** option in the Autonomous Database wizard. Click **Execute** to view the query results in Pivot table.
Clicking **Execute** opens the query results in a new sheet with a PivotTable Fields wizard. Click anywhere outside the table in the spreadsheet to switch the Pivot Table wizard to Native SQL query wizard. Select any cell in the table to continue editing the Pivot Table fields.

You can view the **Grand Total** of the entire pivot table in the last row of the table.

![Pivot Table Example](image)

**Work with the PivotTable Fields wizard**

This topic describes what you need to know to work with the PivotTable Fields wizard.

You can edit the pivot tables in different ways to display data the way you want. You use drop targets to modify the way data is arranged in a data view by dragging and dropping columns to different targets within the view.

For example, you can edit a pivot table in an analysis by dragging the month field to the rows area, the sales field to the Values area and the year name to the Columns area to display its corresponding sales for each month and year.

You can add filters in the Filters area to limit the data in your worksheet.

If you want to remove a field from the pivot table analysis, click on the field you want to remove and select **Remove Field**.

To select the type of calculation you want to summarize the data, select the appropriate option from the **Values** field. The different types of calculations you can perform on the Values field are sum, count, average, maximum, and minimum.

You can customize the appearance of Fields section and Areas section in the Pivot Table Fields wizard. To change the way you want to view the Fields section and the Areas section, select the **Tools** icon in the Pivot Table Fields wizard.
Edit Query Results in the worksheet

Oracle Autonomous Database add-in automatically manages the appearance of an integrated Excel workbook through built-in styles and data format types.

Once the plug in runs the query and retrieves the data into the worksheet, you can view the details of the plug-in's automatically generated query results by hovering over each cell of the workbook.

Sorting the data

When you are working with a lot of data in the worksheet, you need to sort it to optimize your view.

Right click the cell you want to sort, go to Sort and select a sorting option (for example, Sort Oldest to Newest and Sort Newest to Oldest).

Drilling the data

Many of the results in the worksheet represent hierarchical data structures. The metadata specifies these hierarchies, and this enables you to access the different levels of detail within them. Drilling is a way to navigate through data in views quickly and easily.

- Drill down to display data in more detail, which displays more members.
- Drill up to display less data.
When you drill down in a table, the detail level data is added to the current data.

For example, when you drill from a month, the table displays data for the month and for the days in that month.

1. Hover over the cell in the spreadsheet which contains + sign.
2. Click the + sign besides the member you want to drill.

The details are added to the pivot table.

You can now create, manage, and run queries directly with the analytic views in the autonomous database and create powerful data driven reports.

FAQs for Troubleshooting errors with Excel Add-in

If you experience issues with Oracle Autonomous Database Add-in for Excel refer frequently asked questions mentioned in this section to identify and resolve issues.

Why is the My Add-ins icon from the Insert ribbon in the MS Excel workbook greyed out?

Even before installing the Excel add-in, sometimes the My Add-ins icon from the Insert ribbon in the MS Excel workbook appears to be greyed out.

1. From the File menu in Excel ribbon, go to Account and select Manage Settings from the Account page.
2. Ensure that you select the Turn on optional connected experiences.
3. From the File menu in Excel ribbon, go to Options and select the Trust Center option from Excel Options.
4. Click Trust Center Settings and ensure that you deselect Disable all Application Add-ins (if selected) from the Add-ins tab in the Trust Center dialog box.
5. Select the Trusted Add-in Catalogs menu from the Trust Center dialog box and ensure that you deselect the Don't Allow any web add-ins to start checkbox.

Why doesn't the sign-in page of the Excel Add-in load or appear?

At times you might encounter issues with the Excel Add-in even after it is loaded correctly. For example, an add-in fails to load or is inaccessible. Check the compatibility version of the Excel and the operating system you use.

If the compatibility is correct and the sign-in page to the Excel Add-in still does not show up, or it does not load properly, we recommend applying all pending Windows, Office, and browser updates.

1. From the Windows Start menu, select Settings, Update & Security, and then Windows Update.
2. If updates are available on the Windows Update page, review the updates and click Install Now.
Why doesn’t the add-in work correctly after re-installing?

Configure the Excel trusted Add-in catalog to set the add-in correctly after re-installation.

To configure the Excel add-in, check or remove the add-in if it is pointing at a wrong location in the Trusted catalog address. This address should be the same as the location (share path) of the shared manifest folder.

Click Excel’s **File > More > Options > Trust Center > Trust Center Settings > Trusted Add-in Catalogs**

Checking is only required the first time you use the installer, or if the shared manifest folder is changed. The change occurs during uninstalling and re-pointing to a new ADB.

To remove the catalog from the trusted table and add a new catalog pointing to a different address:

- Select the Catalog you want to remove from the trusted catalog table and click **Remove**.
• Enter the correct share path of the shared manifest folder in the **Catalog url** field and click **Add catalog** to add the shared folder to trusted catalog.

Restart Excel to make the new shared folder active to access the add-in.

**Why doesn’t the add-in work correctly even after configuring the Excel trusted Add-in catalog?**

Let’s say you configure the Excel trusted add-in catalog after re-installing the add-in but even then, it does not load correctly. Sometimes the database server changes are not reflected in Excel even after you set the share path of the shared manifest folder as a trusted add-in catalog. Clear the Office cache to resolve this issue.

Refer to this page [https://docs.microsoft.com/en-us/office/dev/add-ins/testing/clear-cache#clear-the-office-cache-on-windows](https://docs.microsoft.com/en-us/office/dev/add-ins/testing/clear-cache#clear-the-office-cache-on-windows) to clear the Office cache on Windows and Mac.

Clearing the Office cache unloads the Excel add-in. Install the add-in and check the configuration of the Excel trusted add-in catalog. This should solve the issue of incorrect loading of the Excel add-in.
The Business Models Page

This page enables you to create, edit, find and view information about business model.

Note:

If you are an Oracle Autonomous Database on dedicated Exadata infrastructure user then the Data Tools will offer a card for either Create Business Models or Data Analysis application. The Data Analysis application is built on the existing Create Business Model functionality and is an extension of this functionality.

To reach the Business Models page, click the Business Models card in the Database Actions page or click the Selector icon and select Business Models from the Data Tools menu in the navigation pane.

These topics describe business models, how to find them, and how to create, edit, or view information about one.

- About Business Models
- Finding and Getting Information about Business Models
- Creating and Editing Business Models

About Business Models

Business models describe the business entities that are derived from data in an Oracle Autonomous Database schema or other sources linked to the database.

Business entities include tables, such as dimension tables for Time and Product values and fact tables for Sales data. Analytic views are also business entities that are derived from database tables and views. Business models contain additional metadata about those entities.

A business model is deployed to an analytic view. A business model doesn't store data but it does explain the relationships between the entities.

Advantages of Business Models

Business models are curated way of modeling a set of data for analytic purposes that can then be used by multiple users.

With business models, you can do the following:

- Analyze your data and answer business queries
- Have a multidimensional view of your data
- Use analytics such as time series calculations, forecasting, allocation operators, and advanced aggregation with additive and non-additive operators
• Analyze your data in terms of hierarchies and use calculations across dimensions and hierarchies, including drilling up and down through hierarchies

• Make your business models available to other users

By identifying relationships among tables and columns, business models enable your system to better optimize queries. They also open new avenues for analyzing data. These avenues include Data Insights, improved hierarchy navigation, and the addition of hierarchy-aware calculations. They can also improve federated access to data not stored in your Oracle Autonomous Database, as they can identify hot data to cache and can optimize how queries are processed in a distributed environment.

Business models work with queries against tables and views. A query against a table that is the base of a business model can be transformed into a query of the analytic view of the business model. The additional metadata in a business model provides more context for the database query optimizer, which enables it to provide a better execution plan. This can transparently improve query performance, such as by reducing joins, performing intelligent caching, generating optimal SQL, and other optimizations.

Finding and Getting Information about Business Models

The Business Models page displays the business models in your Oracle Autonomous Database. When you create a business model, an item for the business model appears on the page.

The item displays the name and a description of the model, the fact table the model is based on, and the state of the model.

Each item has a menu from which you can edit the business model, analyze the model, delete it, or show the DDL that generates the model. From the menu you can also navigate to the Data Insights page to generate Insights about the model, or to the Catalog page.

Business Model States

The color of the icon at the left of the business model item indicates the state of the model. The colors and the indicated states are the following:

• Green: the business model is deployed to an analytic view and the analytic view state is valid.

• Red: the business model is deployed to an analytic view but the analytic view state is invalid because the underlying table has changed.

To resolve this situation and enable the business model to be used again, edit the business model by clicking the Actions (three vertical dots) icon on the item, clicking Edit Business Model, and then clicking Update. If the changes to the underlying table do not affect the business model, the icon turns green. If the changes do affect the model, then the icon turns yellow and an error message appears. Edit the model again, correct the error, and update the model. For example, if in the underlying table, the column for a hierarchy level has been dropped, then you would need to remove that hierarchy level from the model.

• Yellow: the business model is saved but it is not deployed to an analytic view, possibly because the model is incorrect and cannot be deployed to an analytic view.
This can occur if you've made a mistake in editing a model. An error message indicates the cause of the error. Edit the business model by clicking the Actions icon on the item and then clicking **Edit Business Model**. Correct the error and then click **Update**.

**Search for Business Models**

To search for available business models, enter a value in the **Search for Business Models** field and press Enter. The display then include only items whose names contain the characters in the search field and that meet the filter requirements. To clear the search field, click the Clear search results (X) icon in the search field.

**Filter the Display**

By default, business models are filtered by the current user's schema, as indicated by the **Schemas** list below the search field. You can remove the selected schema filter by clicking the Remove filter (X) icon next to it. To search for business models in other schemas, click the Filter (funnel) icon and select one or more schemas. To display the available business models in all schemas, deselect all schema filters.

**Sort the Display**

You can sort the business models by entity type, name, or creation date. By default, the models are sorted by entity name in ascending order, as indicated by the **Sort by** list. To change the order, click the Ascending or Descending (up or down arrow) icon next to the sort value. To remove the sort value, click the Remove sort (X) icon next to the value. To change the sort value, click the Filter (funnel) icon and select one or both of the **Entity Name** or **Created On** options.

**Analyze the Business Model**

When a business model has been deployed to an analytic view, you can analyze the model to view its data in various formats. To analyze the business model, click the Actions menu and then click **Analyze**.

On the Hierarchies tab of the Analyze Business Model pane, you can specify the layout for the data. For each hierarchy, you can choose to have the data display in a row, a column, or a page.

On the Measures tab, you can select the measures to display.

The Data tab displays the data of the selected measures in the chosen format. You can resize a row by grabbing the bottom edge of the row and dragging it to the desired height. You can also right-click the row, point to **Resize**, then **Resize Height**, and enter a value between 20 and 1,000. You can resize a column by grabbing the right edge of the column and dragging it to the desired width. You can also right-click the column, point to **Resize**, then **Resize Width**, and enter a value between 20 and 1,000.

To exit the Analyze Business Model pane, click **OK**.

**Creating and Editing Business Models**

From the Business Models page, you can create a business model and view information about it. You can also edit a business model and perform other actions on it.

When you create a business model, you identify a fact table that contains the data to inspect. The Business Model creation tool looks at the contents of that table, identifies any hierarchies
in the fact table, and searches for other tables that may contain related hierarchies. It then suggests a business model to which you can make any desired adjustments.

While creating a business model, you can enable or disable the following advanced options:

- Autonomous Aggregate Cache, which uses the dimensional metadata of the business model to manage a cache and can improve query response times.
- Analytic View Transparency Views, which presents analytic views as regular database views and enables you to use your analytic tools of choice while gaining the benefits of analytic views.
- Analytic View Base Table Query Transformation, which enables you to use your existing reports and applications without requiring changes to them.

Create a Business Model

To create a business model, click the **Create Model** button.

To cancel the creation of the business model at any time, click **Cancel** on the Create Business Model pane.

Specify Attributes of the Business Model

On the General tab of the Create Business Model pane, specify the following:

- The name for the business model
- The fact table for the model
- Advanced options, including a caption and a description for the model

You can also preview the data of the fact table and see statistics about the data.

In the **Name** field, leave the default model name or replace it with a name of your choice. If you do not replace the default name, then when you select a fact table, the name of the fact table is added to the name. You can change or replace that name.

The **Schema** field has the current user's schema. At present, you can only create a business model in that schema.

In the **Fact Table** field, expand the drop-down list and click **More Sources**. The Select Sources dialog box has a list of the available tables and views. Select a table or view from the list.

You can filter the list by the name of the source or by the owner. To filter the list, begin typing characters in the **Filter** field. As you type, the list changes to show the tables or views that contain the characters. Clear the field to show the complete list again. After you select a table or view, click **OK**.

To enable or disable the advanced options, on the Create Business Model pane, click the Advanced Options icon at the upper right. Select or deselect options as desired.

You can also change the caption and description of the business model by entering values in the **Caption** or **Description** fields. Click **OK** to return to the Create Business Model pane.

To view the data in the fact table and statistics about the data, click the **Preview Data** button. In the Preview and Statistics pane, the Preview tab shows the columns of the table and the data in the columns.
The Statistics tab shows the size of the table and the number of rows and columns. The statistics may take a few moments to appear, during which time the message, "No statistics available..." may appear. The statistics include the names of the columns, their data types, the number of distinct values and null values, the maximum and minimum values, and other information. The bar graph displays the top unique column values and the number of their occurrences for the selected column. Point to a bar in the graph to see the number of occurrences of the unique value.

Click **Close** to close the Preview and Statistics pane and return to the Create Business Model pane.

After specifying the name, the fact table, and any advanced options, click **Next**.

The Generating Business Model dialog box displays the progress of analyzing the fact table and identifying the data sources, joins, hierarchies, and measures to use. When the process completes, click **Close**.

The Data Sources tab appears and has a graphical display of the fact table and the related dimension tables.

**View Data Sources**

The Data Sources tab displays the sources of the data and the relationships between them. For example, a fact table of health insurance data might have columns for geography identifiers, income codes, and gender codes. The Data Sources tab would display items for the fact table and for the geography, income, and gender dimension tables. You can increase the size of the display of the items by pressing the + key or - key.

Pointing to an item displays the name of the table, the application it is associated with, its type, the path to it, and the schema it is in. Click the Actions (three vertical dots) icon at the right of the item to display a menu to expand or collapse the view of the table. For a dimension table, the Actions menu includes a Delete option. To remove the dimension table from the creation of the business model, click **Delete**.

An expanded item displays the columns of the table. Pointing to a column displays the name, application, type, path, and schema of the column.

The lines that connect the dimension tables to the fact table indicate the join paths between them. Pointing to a line displays information about the join paths of the links between the tables.

If the line connects a table that is collapsed, then the line is dotted. If line connects two expanded tables, then the line is solid and connects the column in the dimension table to the column in the fact table.

To return to the General tab, click **Previous**. To see the hierarchies for the business model, click **Next**. To immediately create the business model, click **Create**. To cancel the creation of the business model, click **Cancel**.

**View and Manage Hierarchies**

The Hierarchies tab displays the hierarchies suggested by the Business Model creation tool. The display includes the name of the hierarchy, a caption, a description, the source table, and the levels for the hierarchy.

You can add a hierarchy by clicking the + icon. You can edit or remove a hierarchy by clicking the Actions icon for the hierarchy. For some hierarchies, you can also change the hierarchy into a measure.
If you click **Edit**, then an Edit Hierarchy dialog box displays the suggested name, caption, description, source, and levels for the hierarchy. To change the name, caption, or description of the hierarchy, enter a value in the appropriate field.

- Names determine database object names (in this case, the name of the hierarchy) of analytic view objects as they are used in the deployment of the business model. Certain elements of an analytic view (for example, calculated measures), queries, and other database object reference analytic view object names.

- Captions and Descriptions are generally used as descriptive labels for objects. These are typically 'end user friendly' names.

For example, a hierarchy representing geography areas might be named **GEOGRAPHY_HIERARCHY**, have a caption such as "Geography" and a description of "Geographic areas such as Cities, States, and Countries."

To change the source, select a different source from the drop-down list.

To add a level, click the + button. To remove a level, select it and click the - button. To move a level up or down in a hierarchy, select it and click the up or down arrow button.

To edit a level, click the Actions icon for the level, then click **Edit**. In the Edit Level dialog box, you can enter a different description for the level. You can also select a different column for the member name and the member description. Click **Save** to apply the changes or **Cancel** to discard any changes and return to the Edit Hierarchy dialog box.

When you have finished editing the hierarchy, click **Save** to apply the changes or **Cancel** to discard any changes and return to the Create Business Model pane.

To return to the Data Sources tab, click **Previous**. To see the measures for the business model, click **Next**. To immediately create the business model, click **Create**. To cancel the creation of the business model, click **Cancel**.

**View and Manage Measures**

The Measures tab displays the measures suggested for the business model. It displays the name, caption, description, column, and operator expression for each measure.

To add a measure, click the + icon. To remove a measure, select the measure and click the - icon.

You can select a different column for a measure from the Column drop-down list. You can select a different operator from the Expression drop-down list.

You can return to the Hierarchies tab by clicking **Previous**.

To cancel the creation of the business model, click **Cancel**. To generate the business model, click **Create**, then in the Create Business Model dialog box, click **Yes**.

**Edit a Business Model**

You might want to edit a business model to make changes to the data sources, the hierarchies, or the measures. If when you create a business model, the model is generated but not deployed to an analytic view, then you can edit the model to correct the problem.
To edit a business model, click the Actions (three vertical dots) icon on the business model item, then click **Edit Business Model**. On the Edit Business Model pane, select a tab and make changes as desired.

When you have completed the changes, click **Update**.
The APEX Workspaces Page

You can create APEX workspaces using the APEX Workspaces page. For more information about APEX workspaces, see Workspace and Application Administration in the Oracle Application Express Administration Guide.

To navigate to this page, do either of the following:

- In the Launchpad page, click APEX Workspaces.
- Click Selector to display the navigation menu. Under Administration, select APEX Workspaces.

For a specific workspace, you can perform the following actions using the Actions context menu:

- **Create a Workspace**: See Create a Workspace
- **Delete a Workspace**: Removes the workspace and all associated applications from the database.
- **View Schema Details**: Navigates to the Database Users page to view the related schema information.

Create a Workspace

To create an APEX workspace:

1. In the APEX Workspaces page, at the top right, click Create Workspace.
2. Enter the following fields:
   - **Workspace name**: Enter a name for the workspace.
   - **Database user**: Select a schema from the drop-down list, or type to enter a new schema.
   - **Workspace password**: Enter a password for the workspace.
• **APEX Administrator**: Enter the administrator user details.

**Show code**: Select this option to view the PL/SQL code equivalent of the Create Workspace slider. You can copy and execute this PL/SQL code in the worksheet to perform the same action that occurs when you click **Create** in the Create Workspace slider.

3. Click **Create**.

The new workspace is displayed in the APEX workspaces page. The workspace card includes details such as the number of applications in the workspace, the number of developers using the workspace, and the number of administrators responsible for the workspace. Depending on your role, the **Actions** context menu provides options such as creating and deleting a workspace, and viewing schema details.
The Database Users Page

The Database Users page enables you to perform user management tasks such as create, edit, enable REST, and delete. The actions available are based on the user privileges (CREATE, ALTER, DROP) granted to you. Users with no assigned privileges can still access the Database Users page to browse all users. However, the only action available to them is changing their password.

To navigate to the Database Users page, do either of the following:

- In the Launchpad page, click **Database Users**.
- Click **Selector** to display the navigation menu. Under Administration, select **Database users**.

The Database Users page consists of two parts: Current User and All Users.

**Current User**

Displays information about the current user such as user name, whether REST Enabled or not, REST Alias, account expiration (in days), and the last login date and time. The icon on the left displays the user status with one of the following colours: green (Open), blue (Locked), and red (Expired).

The URL at the bottom is displayed only if the user is REST Enabled. It provides the URL to the Database Actions user login page. Use **Copy to Clipboard** to copy the URL to the user’s clipboard.

Click **Actions** to open the context menu. The actions available are:

- **Edit**: Opens the Edit User Dialog, where you can edit current user information. See About Create/Edit User.
- **Enable REST**: Enables REST for a user where disabled. When this option is selected the first time, it opens the Enable REST dialog. See About REST Enable User.
- **Disable REST**: Disables REST where enabled for a user.
- **Drop REST Services**: Removes the REST data for a user, such as REST Alias, Base Path and so on, which is stored in Oracle REST Data Services (ORDS).
• **Delete**: Opens the Delete User dialog, where you can delete the user. See About Delete User.

**All Users**

Displays information about all other users that have been created in the database. You can use the Search field, which is case insensitive, to search for users, or sort the users in ascending or descending order using the sort icons, or filter by user status or REST status.

To create a user, click **Create User** to open the Create User dialog. For more information, see About Create/Edit User.

There are two views available:

- **(Card View)**: Displays the user information in a card view. This is the default display view. Each user card provides details such as user status, password expiry, user name, and the context menu.

- **(Grid View)**: Displays the user information in a tabular format. The last column in each row contains the context menu icon.

**About Create/Edit User**

You can create a new database user or edit an existing database user.

- To create a new database user, click **Create User** in the Database Users page.
- To edit an existing database user, select **Edit** from the context menu for the associated user.

The user properties are grouped under two tabs: User and Granted Roles.

**User Tab**

Specifies general properties for the database user.

- **User Name**: The user name string. For an existing user, this field is read-only. To change the name, you must drop the user and create a new user with the desired name.

- **Password**: Password string for the new user, or new password for an existing user. You must also type the same password string for **Confirm Password**.

- **Password Expired**: If this option is selected, the password is marked as expired, and the user must change the password before being permitted to connect to the database.
• **Account is Locked**: If this option is selected, the user will not be permitted to connect to the database until a DBA user unlocks the account associated with this user.

• **OML**: If this option is selected, Oracle Maching Learning is enabled for the user. This user is granted the following three roles: OML DEVELOPER, DWROLE, PYQADMIN. The user card displays “OML Enabled”.

**Note:**

For the OML option to be available in the Add Create User slider, create an OML user named “OML$PROXY” and grant the following roles in the SQL worksheet: OML DEVELOPER, DWROLE, PYQADMIN.

**Note:**

To remove the OML roles granted, disable Enable OML. You cannot remove a single role alone, all three roles need to be removed to disable the OML option.

• **Quota on tablespace USERS**: Enter or select to assign quota on the user’s default tablespace. By default, the user has no quota on the default tablespace.

• **Web Access**: If this option is selected, the user is enabled for REST access. Expand **Web access advanced features** to specify the related fields: Authorization required, REST Alias, and URL Mapping Type.

**Granted Roles Tab**

Specifies roles to be granted to the user.

Use **Filter by role** to quickly locate the required roles.

For each role, you can select **Granted** to grant the role, **Admin** to permit the user to grant the role to other users, and **Default** to use the default settings for Granted and Admin. A new user is granted CONNECT and RESOURCE roles when Web Access is selected.

**Show code**: Select this option to view the PL/SQL code equivalent of the Create/Edit User slider. You can copy and execute this PL/SQL code in the worksheet to perform the same action that occurs when you click **Create/Edit** in the Create/Edit User slider.

### About REST Enable User

You can enable REST for a user that has not been REST enabled.

In the Database Users page, select **Enable REST** from the user’s context menu.

When you select Enable REST for a user for the first time, REST Enable User dialog is displayed. Subsequently, if you disable REST and then select Enable REST again, you receive a message stating that REST is enabled. In this case, the REST data previously provided is used for enabling REST. To enter new REST data, select **Drop REST Services** and then select **Enable REST** again.

**Schema Alias**: Enter the alias for the schema name to use in the URL.

**URL Mapping Type**: Select **BASE_PATH** or **BASE_URL**.
**Authorization Required:** For a schema, controls whether Oracle REST Data Services should require user authorization before allowing access to the Oracle REST Data Services metadata catalog of this schema.

### About Delete User

Use this option to delete users.

In the Database Users page, select **Delete** from the user's context menu to delete a user.

**WARNING:**

The number of user's active sessions is displayed in the Delete User dialog window.

- **Cascade:** If this option is selected, all dependent objects are also deleted.
- **Drop REST Services:** If this option is selected, all user REST data is removed from ORDS.

**Note:**

If you do not select this option when deleting a user, the next time you create a user with the same user name, it will still retain the REST-enabled property.

Click **Delete User** and a confirmation notification is displayed.

**Note:**

An error notification is displayed, if the user has active sessions. In such cases, you must close the active sessions before you can delete the user.
The Monitoring Pages

The Monitoring menu provides access to several pages to view the performance and other characteristics of your database.

- The Monitoring Overview Page
- The Performance Hub Page
- The Alerts Page
- The Sessions Page
- The Storage Page
- The Parameters Page
- The Database Monitor Page

The Monitoring Overview Page

The Monitoring Overview page displays general information about the database.

To navigate to the Overview page, click **Selector** to see the navigation menu and then select **Monitoring**.

- **Used Online Database Storage**: Displays how much storage is being used by the database. You can click the title to open the Storage page.
- **Users**: Displays how many user accounts are in the open, locked and expiring statuses. You can hover over one of the statuses to see a list of the user accounts with that status.
- **Sessions**: Displays the status of open sessions in the database. You can click the title to open the Sessions page.
- **Waits**: Displays how many wait events are occurring in the database for various reasons. You can click the title to open the Performance Hub page.
The Performance Hub Page

The Performance Hub page shows performance data for a time period you specify.

To navigate to the Performance Hub page, do either of the following:

• In the Launchpad page, click **Performance Hub**.

• Click **Selector** to display the navigation menu. Under Monitoring, select **Performance Hub**.

**Note:**

The Performance Hub page is available in the following user interface languages: French, Japanese, Korean, Traditional Chinese, and Simplified Chinese. If you change the language to German, Spanish, Italian, or Portuguese in Preferences, the Performance Hub page reverts to English.

The Performance Hub page consists of these parts:

• **Time Range Area**: Use the controls in time range area at the top of the page to specify the time period for which you want to view performance data.

• **ASH Analytics Tab**: Use this tab to explore ASH (Active Session History) information across a variety of different dimensions for the specified time period.

• **SQL Monitoring Tab**: Use this tab to view the top 100 SQL statement executions by different dimensions for the specified time period, and to view details of SQL statement executions you select.

**Time Range Area**

Use these controls in the time range area to specify the time period for which you want to view performance data:

• **Select Duration**: Use this drop-down list, located in the top right of the time range area, to set the timeframe displayed in the timeline. You can choose **Last hour**, **Last 8 hours**, **Last 24 hours**, **Last Week**, or you can choose **Custom** and define your own timeframe.

• **Timeline**: The timeline displays a graph spanning the timeframe selected in the timeframe dropdown, showing Waits, User I/O and CPU usage during the period. At its end is the time slider.

• **Time Slider**: The time slider is a box you can drag back and forth along the current timeline. Use it to pick the specific time period within the timeframe for which you want to view performance data. You can also drag the side handles on...
the time slider to make it wider or narrower to encompass a longer or shorter time period.

**ASH Analytics Tab**

The ASH Analytics tab consists of the Average Active Sessions chart and two secondary tables below it.

- **Average Active Sessions Chart**: This chart shows performance information for the time period defined by the time slider. You can choose to chart different dimensions of information, such as *Wait Class*, *Wait Event*, or *Service*, by selecting the dimension from the drop-down list to the right of the chart title.

  You can download an AWR (Automatic Workload Repository) report for the current time period by right-clicking in the Average Active Sessions chart area and choosing **Generate AWR Report**.

  For more information about ASH and AWR, see Active Session History (ASH) and Automatic Workload Repository (AWR) in *Oracle Database Concepts*.

- **Secondary Tables**: The two tables below the Average Active Sessions chart show the information dimension chosen in the chart filtered by another dimension you choose. For example, if the Average Active Sessions chart is showing Wait Class, you could show SQL ID and User Session dimension information by Wait Class, one in each of the two secondary tables.

  In the secondary tables, the SQL ID and User Session dimensions provide links to SQL Details and Session Details pages, respectively, for the dimension data listed in the table.

**SQL Monitoring Tab**

The SQL Monitoring tab shows a table of the top 100 monitored SQL statements that were executing or that completed during the selected time period.

The table displays information about monitored SQL statement executions. If there is a green spinning icon in the Status column, then the monitored statement did not complete during the selected time period. A red cross indicates that the SQL did not complete either due to an error or due to the session getting terminated. If there is a check mark in the Status column, then the statement completed its execution during the selected time period.

SQL statements are monitored only if they have consumed at least 5 seconds of CPU or I/O time.

You can view information such as the status of a statement, its duration, its type (SQL, PL/SQL, or DBOP), its SQL ID, its SQL plan hash, the user who issued it, whether it executed as a serial or parallel statement, the time the database spent performing CPU activity, I/O, or other activity for the statement, the read and write requests and bytes associated with the statement, and the start and end time for the statement.

Click a SQL ID to display the SQL Details page with more information about that SQL statement.

**The Alerts Page**

*Note:*

Available only if you signed in to an Oracle Autonomous Database on dedicated Exadata infrastructure as a user with administrator rights.
The Alerts page is a chronological log of messages and errors and is commonly used to learn whether the background processes have encountered errors. You can review the alert log periodically to verify that your database system is operating normally.

To navigate to the Alerts page, do either of the following:

- In the Launchpad page, click **Alerts**.
- Click **Selector** to display the navigation menu. Under Monitoring, select **Alerts**.

The alert log includes the following:

- Nondefault initialization parameters used at startup
- Administrative operations, such as STARTUP, SHUTDOWN, ARCHIVE LOG, RECOVER, and CREATE/ALTER/DROP DATABASE/TABLESPACE
- Messages and errors relating to the functions of certain background processes, such as LGWR
- Internal errors (ORA-600), block corruption errors (ORA-1578), and deadlock errors (ORA-60)

Click **Refresh** at the top right of the page to refresh the data.

You can search for a specific value in the log by selecting the display column in the first drop-down list, selecting the condition in the second drop-down list, entering the search value in the box, and clicking the search icon.

In the display table, if you right-click the header row, you see:

- **Columns**: Enables you to select columns to show or hide.
- **Sort**: Displays a dialog box for selecting columns to sort by. For each column, you can specify ascending or descending order, and you can specify that null values be displayed first.

If you right-click any other part of the display table, you see:

- **Count Rows**: Displays the number of rows in the table.
- **Single Record View**: Enables you to view data for a table or view, one record at a time.
- **Copy**: Enables you to copy data from a cell or a row or a range of rows. To copy from more than one row, select the rows you want to copy, right-click by pressing the SHIFT or CTRL key, and select **Copy**.

---

The Sessions Page

**Note:**

Available only if you signed in to an Oracle Autonomous Database on dedicated Exadata infrastructure as a user with administrator rights.

The Sessions page shows information about all currently open sessions in the database.
To navigate to the Sessions page, do either of the following:

- In the Launchpad page, click **Sessions**.
- Click **Selector** to display the navigation menu. Under Monitoring, select **Sessions**.

The data is automatically refreshed at intervals ranging from 10 seconds to 2 minutes. You can also refresh the data by clicking **Refresh** at the top right of the screen.

The table shows summarized data about each open session. Select a session in the table to see more detailed data in the Session Details table below, such as the last SQL statement, explain plan, waits, contention, and so on. You can use the Column, Operator and Value fields to search for the required sessions.

In the display table, if you right-click the header row, you see:

- **Columns**: Enables you to select columns to show or hide.
- **Sort**: Displays a dialog box for selecting columns to sort by. For each column, you can specify ascending or descending order, and you can specify that null values be displayed first.

If you right-click any other part of the display table, you see:

- **Count Rows**: Displays the number of rows in the table.
- **Single Record View**: Enables you to view data for a table or view, one record at a time.
- **Copy**: Enables you to copy data from a cell or a row or a range of rows. To copy from more than one row, select the rows you want to copy by pressing the SHIFT or CTRL key, right-click and select **Copy**.

---

**The Storage Page**

![Note]

Available only if you signed in to an Oracle Autonomous Database on dedicated Exadata infrastructure as a user with administrator rights.

The Storage page shows the storage used based on the current allocation of tablespaces along with additional drill-down capabilities to view segments.

To navigate to the Storage page, do either of the following:

- In the Launchpad page, click **Storage**.
- Click **Selector** to display the navigation menu. Under Monitoring, select **Storage**.

You can refine the list of segments shown by using the filter feature. Click **View Datafiles** to view the datafiles in each tablespace.

You can view tablespace and segment space usage.

To view space usage information

1. From the Database drop-down menu, click **Storage**.
The Storage page displays. If the Oracle database is version 12c or later, the Storage page shows the used and allocated storage space for tablespaces in any pluggable database. If the Oracle database is version 11g, the Storage page shows the used and allocated space for the entire database.

2. You can click a tablespace to view its storage information. An interactive report appears, showing the segments that exist within the tablespace. Most segments are user objects, and they include tables, LOBs, and indexes.

3. On the Segments page, you can refine the list of segments shown by using the filter feature.
   For example, you can search for all the segments for a specific owner (schema) by selecting OWNER from the first drop-down list, entering the owner (schema) name in the box, and clicking the search icon.

The Parameters Page

**Note:**

Available only if you signed in to an Oracle Autonomous Database on dedicated Exadata infrastructure as a user with administrator rights.

The Parameters pages displays initialization parameters, which are used to configure the database instance, including memory structures, and define locations for database files.

To navigate to the Parameters page, do either of the following:

- In the Launchpad page, click Parameters.
- Click Selector to display the navigation menu. Under Monitoring, select Parameters.

Values for initialization parameters are stored in a text-based initialization parameter file (PFILE) or binary server parameter file (SPFILE). The initialization parameter file is read at database instance startup.

Click Refresh at the top right of the page to refresh the data.

To perform a search, enter values in the search criteria columns and click the search icon to locate the initialization parameter.

In the display table, if you right-click the header row, you see:

**Columns:** Enables you to select columns to show or hide.

**Sort:** Displays a dialog box for selecting columns to sort by. For each column, you can specify ascending or descending order, and you can specify that null values be displayed first.

If you right-click any other part of the display table, you see:

**Count Rows:** Displays the number of rows in the table.

**Single Record View:** Enables you to view data for a table or view, one record at a time.
**Copy**: Enables you to copy data from a cell or a row or a range of rows. To copy data from more than one row, select the rows you want to copy by pressing Shift or Ctrl, right-click and select **Copy**.

---

## The Database Monitor Page

**Note:**

Available only if you signed in to an Oracle Autonomous Database as a user with administrator rights.

The Database Monitor page provides information about the performance of an Autonomous Database.

To navigate to the Database Monitor page, do either of the following:

- In the Launchpad page, click **Database Monitor**.
- Click **Selector** to display the navigation menu. Under Monitoring, select **Database Monitor**.

The Database Monitor page consists of the following tabs, which show real-time and historical information about the utilization of an Autonomous Database:

- **Overview**
- **Monitor**

### Database Monitor Overview

The **Overview** tab shows real-time and historical information about the Autonomous Database utilization.

The charts shown on this page include:

- **Storage**: This chart shows the provisioned, allocated, and used storage. The chart indicates what percentage of the space is currently in-use.
Provisioned storage is the amount of storage you select when you provision the instance or when you modify storage by scaling storage.

**Storage allocated** is the amount of storage physically allocated to all data tablespaces and temporary tablespaces and includes the free space in these tablespaces. This does not include storage for the sample schemas.

**Storage used** is the amount of storage actually used in all data and temporary tablespaces. This does not include storage for the sample schemas. The storage used is the storage in the Autonomous Database as follows:

- Storage used by all database objects. Note: the chart does not include storage for the sample schemas as they do not count against your storage.
- Storage for files users put in the file system.
- Storage used by temporary tablespaces.
- Used storage excludes the free space in the data and temporary tablespaces.

By default the chart does not show the used storage. Select **Storage used** to expand the chart to see used storage (the values are calculated when you open the chart).

For an Autonomous JSON Database the chart shows an additional field showing the percentage of storage used that is not storing JSON documents.
Note:

If you drop an object, the space continues to be consumed until you empty the recycle bin. See Purging Objects in the Recycle Bin for more information.

See Use Sample Data Sets in Autonomous Database for information on sample schemas SH and SSB.

- **CPU utilization (%)**: This chart shows the historical CPU utilization of the service:
  - OCPU auto scaling disabled: this chart shows hourly data. A data point shows the average CPU utilization for that hour. For example, a data point at 10:00 shows the average CPU utilization for 9:00-10:00.
    
    The utilization percentage is reported with respect to the number of CPUs the database is allowed to use which is two times the number of OCPUs. For example, if the database has four (4) OCPUs, the percentage in this graph is based on 8 CPUs.

- OCPU auto scaling enabled: For databases with OCPU auto scaling enabled the utilization percentage is reported with respect to the maximum number of CPUs the database is allowed to use, which is six times the number of OCPUs. For example, if the database has four OCPUs with auto scaling enabled the percentage in this graph is based on 24 CPUs.

- **Running SQL statements**: This chart shows the average number of running SQL statements historically. This chart shows hourly data. A data point shows the running
SQL statements for that hour. For example, a data point at 10:00 shows the average number of running SQL statements for 9:00-10:00.

- **Number of OCPUs allocated**

  Notes for display results:
  - OCPU auto scaling disabled: For databases with OCPU auto scaling disabled, for each hour the chart shows the number of OCPUs allocated to the database if the database is open for at least some part of the hour.
  - OCPU auto scaling enabled: For databases with OCPU auto scaling enabled, for each hour the chart shows the average number of OCPUs used during that hour if that value is higher than the number of OCPUs provisioned. If the number of OCPUs used is not higher than the number of OCPUs provisioned, then the chart shows the number of OCPUs allocated for that hour.
  - Stopped Database: If the database was stopped for the full hour the chart shows 0 OCPUs allocated for that hour.

  Click **Show details** for more information, including the number of OCPUs allocated to the database and to external resources, and the total allocated OCPUs.

  The Show details view includes separate values for database OCPU usage and external resource OCPU usage. External resources include: Cloud SQL, Graph, OML4PY, and others. The Total OCPUs are the total number of OCPUs in use on the Autonomous Database. The external OCPUs value shows how external OCPUs contribute to the total OCPU usage.
• **SQL statement response time (s):** This chart shows the average response time, in seconds, of SQL statements historically. This chart shows hourly data. A data point shows the average SQL statement response time for that hour. For example, a data point at 10:00 shows the average SQL statement response time, in seconds, for the hour from 9:00-10:00.

![SQL statement response time chart]

• **SQL statements executed per second**

  **Note:**
  Database Monitor does not show this chart when the Autonomous Database instance workload type is **Data Warehouse**.

![SQL statements executed per second chart]

The default retention period for performance data is thirty (30) days. The CPU utilization, running statements, and average SQL response time charts show data for the last eight (8) days by default.
Note:

You can change the retention period by modifying the Automatic Workload Repository retention setting with the PL/SQL procedure DBMS_WORKLOAD_REPOSITORY.MODIFY_SNAPSHOT_SETTINGS(). Be aware that increasing the retention period will result in more storage usage for performance data. See Oracle Database PL/SQL Packages and Types Reference.

Database Monitor Activity

The Monitor tab shows real-time and historical information about the Autonomous Database performance data, activity, and utilization.

Note:

The default view in the Monitor tab is real-time. This view shows performance data for the last hour.

The charts on this page are:

- **Database Activity**
  This chart shows the average number of sessions in the database using CPU or waiting on a wait event. See Oracle Database Reference for more information on wait events.

- **CPU Utilization**
  This chart shows the CPU utilization of each consumer group. The utilization percentage is reported with respect to the number of CPUs the database is allowed to use which is two times the number of OCPUs. For example, if the database has four (4) OCPUs, the percentage in this graph is based on 8 CPUs.

  For databases with OCPU auto scaling enabled the utilization percentage is reported with respect to the maximum number of CPUs the database is allowed to use, which is six times the number of OCPUs. For example, if the database has four OCPUs with auto scaling enabled the percentage in this graph is based on 24 CPUs.

  See Manage Concurrency and Priorities on Autonomous Database for detailed information on consumer groups.

- **Running Statements**
  This chart shows the average number of running SQL statements in each consumer group.

  See Manage Concurrency and Priorities on Autonomous Database for detailed information on consumer groups.

- **Queued Statements**
  This chart shows the average number of queued SQL statements in each consumer group.
See Manage Concurrency and Priorities on Autonomous Database for detailed information on consumer groups.

To see earlier data click **Time period**. The default retention period for performance data is thirty (30) days. By default in the Time Period view the charts show information for the last eight (8) days.

In the time period view you can use the calendar to look at a specific time period in the past 30 days. You can also use the time slider to change the period for which performance data is shown.

---

**Note:**

The retention time can be changed by changing the Automatic Workload Repository retention setting with the PL/SQL procedure `DBMS_WORKLOAD_REPOSITORY.MODIFY_SNAPSHOT_SETTINGS`. Be aware that increasing the retention time results in more storage usage for performance data. See *Oracle Database PL/SQL Packages and Types Reference*.

---

### The AWR Page

**Note:**

Available only if you signed in to an Oracle Autonomous Database on dedicated Exadata infrastructure as a user with administrator rights.

The AWR page in Database Actions enables you to generate, view and download Automatic Workload Repository (AWR) reports.

To navigate to the AWR page, do either of the following:

- In the Launchpad page, in the Monitoring category, click **AWR**.
- Click **Selector** to display the navigation menu. Under Monitoring, select **AWR**.

The Automatic Workload Repository collects, processes, and maintains performance statistics for the database. The gathered data can be displayed in reports and views.

To generate or view a report in the AWR page, select the range of the time period required using the **Start ID** and **End ID** fields. The snapshot dropdown list for each field is sorted in descending order and starts with the most recent database snapshot.

**Note:**

The **Generate Report** button is available only when the value in the Start ID field is lower than the value in the End ID field.
## WORKLOAD REPOSITORY PDB report (root snapshots)

<table>
<thead>
<tr>
<th>DB Name</th>
<th>DB Id</th>
<th>Unique Name</th>
<th>Paris</th>
<th>Edition</th>
<th>Release</th>
<th>RAC</th>
<th>CDB</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB13C</td>
<td>20B5474074</td>
<td>DB13C</td>
<td>PRIMARY</td>
<td>8E</td>
<td>21.0.0.0.0</td>
<td>NO</td>
<td>YES</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Instance</th>
<th>Inst Num</th>
<th>Starting Time</th>
<th>User Name</th>
<th>System Data Visible</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB13C</td>
<td>1</td>
<td>06-Jun-20 06:09</td>
<td>P0000A</td>
<td>YES</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Container DB Id</th>
<th>Container Name</th>
<th>Open Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3d5142155</td>
<td>DB13P</td>
<td>06-Jun-20 06:09</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Host Name</th>
<th>Platform</th>
<th>CPUs</th>
<th>Cores</th>
<th>Sockets</th>
<th>Memory (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>db213</td>
<td>Linux x86 64-bit</td>
<td>48</td>
<td>24</td>
<td>2</td>
<td>902</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Snap Id</th>
<th>Snap Time</th>
<th>Sessions</th>
<th>CPU Sessions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Begin Snap:</td>
<td>3</td>
<td>06-Jun-20 07:58:33</td>
<td>36</td>
</tr>
<tr>
<td>End Snap:</td>
<td>4</td>
<td>06-Jun-20 08:08:47</td>
<td>29</td>
</tr>
</tbody>
</table>
Supported SQL*Plus and SQLcl Commands

This following sections list the SQL*Plus and SQLcl commands supported in the worksheet.

Supported SQL*Plus Commands

The SQL worksheet supports most of the SQL*Plus commands except those statements that are related to formatting.

- / (slash)
- @ (url | file_name[.ext] ) [arg ...]
- ACC[CEPT] variable [NUM[BER] | CHAR | DATE | BINARY_FLOAT | BINARY_DOUBLE] [FOR[MAT] format] [DEF[AULT] default] [PROMPT text | NOPR[OMPT]] [HIDE]
- ARCHIVE LOG LIST
- BRE[AK] [ON report_element [action [action]]] ...
- BTI[TLE] [printspec [text | variable] ...] | [ON | OFF]
- C[HANGE] sepchar old [sepchar [new [sepchar]]]
- CL[EAR] option ...
- COL[UMN] [(column | expr) [option ...]]
- COMP[UTE] [function [LAB[EL] text] ... OF {expr | column | alias} ...ON {expr | column | alias | REPORT | ROW} ...]
- COPY {FROM database | TO database | FROM database TO database} {APPEND | CREATE | INSERT | REPLACE} destination_table[(column, column, column, ...)] USING query
- DEF[INE] [variable] | [variable = text]
- DESC[RIBE] {[schema.]object[@connect_identifier]}
- DISC[ONNECT]
- EXEC[UTE] statement
- (EXIT | QUIT) [SUCCESS | FAILURE | WARNING | n | variable | :BindVariable] [COMMIT | ROLLBACK]
- GET [FILE] file_name[.ext] [LIST | NOLIST]
- HO[ST] [command]
- L[IST] [n | n m | n * | n LAST | * | * n | * LAST | LAST]
- PAU[SE] [text]
- PRINT [variable ...]
- PRO[MPT] [text]
• {QUIT | EXIT} [SUCCESS | FAILURE | WARNING | n | variable 
  | :BindVariable] [COMMIT | ROLLBACK]
• R[UN]
• SAV[E] [FILE] file_name[.ext] [CRE[ATE] | REP[LACE] | APP[END]]
• SET system_variable value
• SHO[W] [option]
• SHUTDOWN [ABORT | IMMEDIATE | NORMAL | TRANSACTIONAL [LOCAL]]
• STA[RT] { url | file_name[.ext] } [arg ...]
• STARTUP db_options | cdb_options | upgrade_options
• TIMI[NG] [START text | SHOW | STOP]
• TTI[TLE] [printspec [text | variable] ...] | [ON | OFF]
• UNDEF[INE] variable ...
• VAR[IABLE] [variable [type][=value]]
• XQUERY xquery_statement

Supported SQLcl Commands

The SQL worksheet supports many of the SQLcl commands.

• ALIAS
• APEX
• BRIDGE
• CTAS
• DDL
• FORMAT
• INFORMATION
• LOAD
• NET
• OERR
• RESERVED_WORDS
• SCRIPT
• SERRORL
• SODA (See SODA Commands)
• TNSPING
• TOSUB
• WHICH
SODA Commands

SODA (Simple Oracle Document Access) commands are supported in the SQL code editor. SODA allows schemaless application development using the JSON data model. The commands are:

- **SODA create <collection_name>** — Creates a new collection
- **SODA list** — Lists all the collections
- **SODA get <collection_name> [-all | -f | -k | -klist] [{{<key>} | <k1> <k2> ... | <qbe>}]** — Lists documents in the collection. Optional arguments:
  - all: Lists the keys of all documents in the collection
  - k: Lists documents matching the specific <key>
  - klist: Lists documents matching the list of keys
  - f: Lists documents matching the <qbe>
- **SODA insert <collection_name> <json_str | filename>** — Inserts a new document within a collection
- **SODA drop <collection_name>** — Deletes existing collection
- **SODA count <collection_name> [qbe]]** — Counts number of documents inside collection. Optional parameter <qbe> returns number of matching documents
- **SODA replace <collection_name> <oldkey> <new_{str | doc}>** — Replaces one document with another
- **SODA remove <collection_name> [-k | -klist | -f] [{{<key>} | <k1> <k2> ... | <qbe>]}** — Removes documents from collection. Optional arguments:
  - k: Removes document in collection matching the specific <key>
  - klist: Removes document in collection matching the list <key1> <key2> ...
  - f: Removes document in collection matching <qbe>