<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run a Notebook</td>
<td>3-4</td>
</tr>
<tr>
<td>Export a Notebook</td>
<td>3-7</td>
</tr>
<tr>
<td>Import a Notebook</td>
<td>3-7</td>
</tr>
<tr>
<td>Create Text Input Forms in Notebooks</td>
<td>3-8</td>
</tr>
<tr>
<td>Create Select Forms in Notebooks</td>
<td>3-9</td>
</tr>
<tr>
<td>Create Check Box Forms in Notebooks</td>
<td>3-10</td>
</tr>
<tr>
<td>Save Notebooks as Templates</td>
<td>3-10</td>
</tr>
<tr>
<td>Version a Notebook</td>
<td>3-11</td>
</tr>
<tr>
<td>Set Output Format in Notebooks</td>
<td>3-11</td>
</tr>
<tr>
<td>Output Formats Supported by SET SQLFORMAT Command</td>
<td>3-12</td>
</tr>
<tr>
<td>About Interpreter Bindings and Notebooks</td>
<td>3-13</td>
</tr>
<tr>
<td>Set Interpreter Bindings for Notebooks</td>
<td>3-15</td>
</tr>
<tr>
<td>Change Interpreter Bindings for Specific Paragraphs in a Notebook</td>
<td>3-15</td>
</tr>
<tr>
<td>Validate Interpreter Bindings</td>
<td>3-19</td>
</tr>
<tr>
<td>Use the Scratchpad</td>
<td>3-20</td>
</tr>
<tr>
<td>Run SQL Scripts</td>
<td>3-20</td>
</tr>
<tr>
<td>Run SQL Statements</td>
<td>3-22</td>
</tr>
<tr>
<td>Collaborate in Oracle Machine Learning</td>
<td>3-22</td>
</tr>
</tbody>
</table>

## 4 Use Templates to Collaborate with Users

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use the Personal Templates</td>
<td>4-1</td>
</tr>
<tr>
<td>Create Notebooks from Templates</td>
<td>4-2</td>
</tr>
<tr>
<td>Share Notebook Templates</td>
<td>4-2</td>
</tr>
<tr>
<td>Edit Notebook Templates Settings</td>
<td>4-2</td>
</tr>
<tr>
<td>Use the Shared Templates</td>
<td>4-3</td>
</tr>
<tr>
<td>Create Notebooks from Templates</td>
<td>4-3</td>
</tr>
<tr>
<td>Edit Notebook Templates Settings</td>
<td>4-4</td>
</tr>
<tr>
<td>Use the Example Templates</td>
<td>4-4</td>
</tr>
<tr>
<td>Create a Notebook from the Example Templates</td>
<td>4-5</td>
</tr>
<tr>
<td>Example Templates</td>
<td>4-6</td>
</tr>
</tbody>
</table>

## 5 Get Started with Jobs

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>About Jobs</td>
<td>5-1</td>
</tr>
<tr>
<td>Create Jobs to Schedule Notebooks</td>
<td>5-2</td>
</tr>
<tr>
<td>View Job Logs</td>
<td>5-3</td>
</tr>
</tbody>
</table>

## 6 Get Started with Connection Groups

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>About Connection Groups</td>
<td>6-1</td>
</tr>
<tr>
<td>About Global Connection Group</td>
<td>6-2</td>
</tr>
</tbody>
</table>
7 Use Notebook Sessions
   About Notebook Sessions 7-1

8 Administer Oracle Machine Learning
   Typical Workflow for Managing Oracle Machine Learning 8-2
   Manage OML Users 8-3
      Create User 8-3
      Add Existing Database User Account to Oracle Machine Learning 8-4
   About User Data 8-5
      Reassign 8-6
   About Compute Resource 8-6
Preface

This document describes how to use Oracle Machine Learning and provides references to related documentation.

- Audience
  This document is intended for data scientists, developers, and business users.

- Conventions
- Documentation Accessibility
- Related Resources
  For more information, see these related resources.

Audience

This document is intended for data scientists, developers, and business users.

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><code>monospace</code></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>

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.
Related Resources

For more information, see these related resources.

- *Getting Started with Oracle Cloud*
- *Accessibility Guide for Oracle Cloud Services*
1

Get Started with Oracle Machine Learning Notebooks

This section discusses how to get started with Oracle Machine Learning notebooks, and use Apache Zeppelin-based data mining notebooks where you can perform data exploration and data visualization, data preparation and machine learning.

• About Oracle Machine Learning Notebooks
  Oracle Machine Learning Notebooks is an Apache Zeppelin-based collaborative web-based interface that provides a development environment to create data mining notebooks where you can perform data exploration and visualizations, data preparation and machine learning.

• Oracle Machine Learning Notebooks Home Page
  The Oracle Machine Learning Notebooks home page is the default landing page when you log in to Oracle Machine Learning. The home page provides you quick links to important interfaces, and the log of your high-level recent activities.

• Get Started with Oracle Machine Learning
  Here is how you can get started with Oracle Machine Learning.

• Access Oracle Machine Learning User Management Page
  From Autonomous Data Warehouse you can access the Oracle Machine Learning Manage Oracle ML Users page.

• Typical Workflow For Using Notebooks
  To begin with Oracle Machine Learning, refer to the tasks listed in the table as a guide.

• Quickstart Tutorial: Creating Projects and Workspaces in Oracle Machine Learning
  This tutorial shows you the steps to create projects and workspaces in Oracle Machine Learning.

• Quickstart Tutorial: Creating and Running Notebooks in Oracle Machine Learning
  This tutorial shows you how to create a notebook and run it in Oracle Machine Learning.

• Quickstart Tutorial: Collaborating in Oracle Machine Learning
  This tutorial shows you how to collaborate and share notebooks with other users in Oracle Machine Learning.

• Quickstart Tutorial: Creating SQL Scripts in Oracle Machine Learning
  This tutorial shows you how to create a SQL script and share it with other users in Oracle Machine Learning.

• Quickstart Tutorial: Running SQL Statements in Oracle Machine Learning
  This tutorial shows you how to run SQL statements in Oracle Machine Learning.

Related Topics
• Data Mining Algorithms
About Oracle Machine Learning Notebooks

Oracle Machine Learning Notebooks is an Apache Zeppelin-based collaborative web-based interface that provides a development environment to create data mining notebooks where you can perform data exploration and visualizations, data preparation and machine learning.

Key features of Oracle Machine Learning Notebooks:

- Allows collaboration among data scientists, business and data analysts, application and dashboard developers, Database Administrators, and IT professionals
- Leverages the scalability and performance of Oracle Platform and its Cloud Services

Oracle Machine Learning Notebooks Home Page

The Oracle Machine Learning Notebooks home page is the default landing page when you log in to Oracle Machine Learning. The home page provides you quick links to important interfaces, and the log of your high-level recent activities.

In the home page, you can access:

- The How Do I help links to:
  - Get Started
  - Create Notebooks
  - Create Jobs
- Quick Actions:
  - Run SQL Statements
    - Enter and run SQL statements
  - Run SQL Scripts
    - Enter and run SQL scripts
  - Notebooks
    - The place for data discovery and analytics
  - Jobs
    - Schedule notebooks to run at certain times
  - Examples
    - Check out some examples
- Recent Activities
  - Nothing to Display
- Manage Permissions
- Try It

• The Quick Actions links to:
  - Run SQL Statements
  - Run SQL Scripts
  - Notebooks
  - Jobs
  - Examples

• The log of your Recent Activities.

• The application navigation by clicking on the top left corner of the home page.

• The options to select and create new projects, access recent projects, manage workspace, and set workspace permissions by clicking on the top right corner of the home page.

• The Recent Notebooks on the right pane of the home page.

Related Topics

• Run SQL Statements
  SQL or Structured Query Language is the standard language for relational database management systems. You can use SQL statements to perform tasks such as retrieving data from a database, updating data on a database, and so on. Some examples of SQL statements are SELECT, INSERT, UPDATE, DELETE, CREATE, and DROP.

• Run SQL Scripts
  A SQL script is a set of SQL statements. You can create SQL scripts in Oracle Machine Learning Notebooks, save them as .json files in your system. You can share the notebook.json files with other users as well.

• Notebooks
  The Notebooks page lists all the notebooks associated with the selected project. You can create, edit, and run your notebooks here.

• Jobs
  Jobs allow you to create schedules to run notebooks. In the Jobs page, you can create and schedule jobs, monitor job status, and view job logs as read-only notebooks.

• Examples
  The Example Templates page lists the pre-populated Oracle Machine Learning notebook templates. You can view and use these templates to create your notebooks.

Get Started with Oracle Machine Learning

Here is how you can get started with Oracle Machine Learning.

1. Request access to Oracle Machine Learning. Contact your Service Administrator to provide access to your Oracle Machine Learning account.
2. Access the Oracle Machine Learning account by using your credentials. In case you forget your password, then request the Administrator to reset it.

**Note:**

Once you receive your new password, you must change it immediately. Refer to the Oracle Machine Learning password policy for more information.

3. Once you log in for the first time, a workspace and project will be created for you. You can start creating your notebook and assign it to the default project and workspace. You can also create your own project and workspace.

**Related Topics**

- **About Oracle Machine Learning Password Policy**
  All Oracle Machine Learning users must follow the password policy to create a strong and secured password.

**Access Oracle Machine Learning User Management Page**

From Autonomous Data Warehouse you can access the Oracle Machine Learning Manage Oracle ML Users page.

To access Oracle Machine Learning Manage Oracle ML Users page:

1. Sign in to your Oracle Cloud Account at cloud.oracle.com.

2. On the My Services page, click and select Compute.

3. From the Oracle Cloud Infrastructure page click in the top corner of the page and then click Autonomous Data Warehouse from the left navigation pane, and select a Compartment.

4. Select an Autonomous Data Warehouse instance and on the details page click Service Console.
5. Click Administration.

6. On the Administration page click Manage Oracle ML Users.
   - About Oracle Machine Learning Password Policy
     All Oracle Machine Learning users must follow the password policy to create a strong and secured password.

About Oracle Machine Learning Password Policy

All Oracle Machine Learning users must follow the password policy to create a strong and secured password.

When changing or modifying your Oracle Machine Learning password, ensure that you follow these conditions:

- The password must be between 12 and 30 characters long. It must include at least one uppercase letter, one lowercase letter, and one numeric character.
- The password cannot contain the username.
- The password cannot be one of the last 4 passwords used for the same username.
- The password cannot contain the double quote (" ) character.
- The password must not be the same password that is set less than 24 hours ago.
Typical Workflow For Using Notebooks

To begin with Oracle Machine Learning, refer to the tasks listed in the table as a guide.

<table>
<thead>
<tr>
<th>Tasks</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Oracle Machine Learning</td>
<td>Access Oracle Machine Learning</td>
</tr>
<tr>
<td>Create workspaces</td>
<td>Create Projects and Workspaces</td>
</tr>
<tr>
<td>Create projects</td>
<td>Create Projects and Workspaces</td>
</tr>
<tr>
<td>Create notebooks</td>
<td>Create a Notebook</td>
</tr>
<tr>
<td>Run SQL scripts</td>
<td>Run SQL Scripts</td>
</tr>
<tr>
<td>Run SQL statements</td>
<td>Run SQL Statements</td>
</tr>
<tr>
<td>Create jobs to schedule notebooks</td>
<td>Create Jobs to Schedule Notebook</td>
</tr>
</tbody>
</table>

Related Topics

- **Grant Workspace Permissions**
  You can collaborate with other users in Oracle Machine Learning by granting permissions to access your workspace. Your workspace contains your projects and notebooks.

Quickstart Tutorial: Creating Projects and Workspaces in Oracle Machine Learning

This tutorial shows you the steps to create projects and workspaces in Oracle Machine Learning.

- [Creating Projects and Workspaces in Oracle Machine Learning](#)

Quickstart Tutorial: Creating and Running Notebooks in Oracle Machine Learning

This tutorial shows you how to create a notebook and run it in Oracle Machine Learning.

- [Creating and Running Notebooks in Oracle Machine Learning](#)

Quickstart Tutorial: Collaborating in Oracle Machine Learning

This tutorial shows you how to collaborate and share notebooks with other users in Oracle Machine Learning.

- [Collaborating in Oracle Machine Learning](#)
Quickstart Tutorial: Creating SQL Scripts in Oracle Machine Learning

This tutorial shows you how to create a SQL script and share it with other users in Oracle Machine Learning.

Creating SQL Scripts in Oracle Machine Learning

Quickstart Tutorial: Running SQL Statements in Oracle Machine Learning

This tutorial shows you how to run SQL statements in Oracle Machine Learning.

Running SQL Statements in Oracle Machine Learning
Get Started with Project and Workspaces

A project is a container for storing your notebooks and other objects such as dashboards and so on. A workspace is a virtual space where your projects reside, and multiple users with the appropriate permission type can work on different projects.

- **Create Projects and Workspaces**
  A project is a container for your notebooks, and a workspace is a container for your projects. You can own multiple projects in a workspace.

- **Grant Workspace Permissions**
  You can collaborate with other users in Oracle Machine Learning by granting permissions to access your workspace. Your workspace contains your projects and notebooks.

### Create Projects and Workspaces

A project is a container for your notebooks, and a workspace is a container for your projects. You can own multiple projects in a workspace.

The initial workspace and the default project is created by the Oracle Machine Learning service automatically when you log in to Oracle Machine Learning for the first time. To create a new project and a workspace:

1. On the top right corner of Oracle Machine Learning home page, click the project workspace drop-down list. The project name and the workspace, in which the project resides, are displayed here. In this screenshot, the project name is Project A, and the workspace name is Admin. If a default project exists, then the default project name is displayed here. To choose a different project, click **Select Project**.
2. To create a new project, click **New Project**.
   The Create Project dialog box opens.

3. In the **Name** field, provide a name for your project.

4. In the **Comments** field, enter comments, if any.

5. In the **Select Workspace** field, select a workspace from the drop-down list. Your project is assigned to the selected workspace. If you want to create a new workspace, then click `+`.

6. In the Create Workspace dialog box, enter a name for the workspace in the **Name** field.

7. In the **Comments** field, enter comments, if any.

8. Click **OK**.
   This creates your workspace, and navigates back to the Create Project dialog box. The project that you are creating is now assigned to the newly created workspace.

9. Click **OK**.
   Your project is created and assigned to the workspace that you have selected or created.

---

## Grant Workspace Permissions

You can collaborate with other users in Oracle Machine Learning by granting permissions to access your workspace. Your workspace contains your projects and notebooks.

By granting different types of permissions such as **Manager**, **Developer**, and **Viewer**, you can allow another user to view your workspace and perform different tasks in your projects and notebooks such as edit, create, update, delete, run, view notebooks and so on. For more information about the permission types, see [About Workspace Permission Types](#).

---

### Caution:

If you grant the permission type **Manager** or **Developer**, then the user can also drop tables, create tables, and run any scripts at any time on your account. The user with **Viewer** permission type can only view your notebooks, and is not authorized to run or make any changes to your notebooks.

To grant permission to another user:
1. On the top right corner of Oracle Machine Learning home page, click the project and workspace drop-down list and select **Workspace Permission**. The Permissions dialog box opens.

2. In the Permissions dialog box, select a user from the **Username** drop-down list.

3. In the **Permissions** drop-down list, select the permission type you want to grant to the selected user.
   - **Manager**
   - **Developer**
   - **Viewer**

4. Click **Add**.
   
The selected user is granted the assigned permission. The user name is displayed in the Permissions dialog box, along with the permission type.
5. Click **OK**. This completes the task of granting permission to a user. To delete a user and the associated permission, select the user and click **Delete**.

- **About Workspace Permission Types**
  Oracle Machine Learning allows three types of permissions. Depending on the permission type, you can allow the user to view or perform different tasks in your workspace, projects, and notebooks.

- **Manage Workspaces**
  You can provide access to your workspace, manage permissions for users, and edit and delete workspace.

### About Workspace Permission Types

Oracle Machine Learning allows three types of permissions. Depending on the permission type, you can allow the user to view or perform different tasks in your workspace, projects, and notebooks.

The three types of permissions are listed in the following table along with the actions that are allowed.

<table>
<thead>
<tr>
<th>Permission Types</th>
<th>Actions based on permission</th>
</tr>
</thead>
</table>
| Manager          | • Project: Create, update, delete.  
|                  | • Workspace: View only.  
|                  | • Notebooks: Create, update, run, delete, and schedule jobs.  |
| Developer        | • Project: View only.  
|                  | • Workspace: View only.  
|                  | • Notebooks: Create, update, run, and delete notebooks that a developer creates only.  |
| Viewer           | • Project: View only.  
|                  | • Workspace: View only.  
|                  | • Notebooks: View only.  
|                  | • Jobs: View jobs and job runs of shared notebooks only.  |

**Note:**

A user with the **Manager** permission type can also drop tables and run any script at any time on the owner’s account.

**Note:**

A user with the **Developer** permission type can also drop tables and run any script at any time on the owner’s account.  

• Jobs: View and run jobs of shared notebooks only. A developer cannot create jobs for notebooks that are shared.
Manage Workspaces

You can provide access to your workspace, manage permissions for users, and edit and delete workspace.

To manage workspaces:

1. On the top right corner in your home page, click **Project**.
2. Select **Manage Workspaces**.
   The Manage Workspaces dialog box opens.
3. You can perform the following tasks:
   - Click **Permissions** to add and delete users
   - Click **Edit** to edit the selected workspace
   - Click **Delete** to delete the selected workspace
4. Once you have finished your tasks, click **Close**.
Get Started with Notebooks for Data Analysis and Data Visualization

Oracle Machine Learning Notebooks is based on Apache Zeppelin technology. You can write code, text, create visualizations, and perform data analytics including machine learning. Notebooks work with interpreters in the back-end. In Oracle Machine Learning, notebooks are available in a project, where you can create, edit, delete, and even save notebooks as templates.

- **About Notebooks**
  The Notebooks page lists all the notebooks associated with the selected project. You can create, edit, and run your notebooks here.

- **About Interpreter Bindings and Notebooks**
  An interpreter is a plug-in that allows you to use a specific data processing language at the backend. For the Zeppelin Notebooks in Oracle Machine Learning, you use the sql and pl/sql interpreters within an Oracle Database interpreter group and the md (MarkDown) interpreter for plain text formatting syntax so that it can be converted to HTML.

- **Use the Scratchpad**
  You can use the SQL scratchpad to run your SQL statements and SQL script without opening or creating a notebook.

- **Collaborate in Oracle Machine Learning**
  Two or more users can collaborate and share Oracle Machine Learning notebooks with other users.

### About Notebooks

The Notebooks page lists all the notebooks associated with the selected project. You can create, edit, and run your notebooks here.

You can perform the following tasks in the Notebooks page:

- To create a new notebook, select a project and click **Create**.
- To edit a notebook, select the notebook and click **Edit**. You can edit the notebook name, and add comments in the Edit Notebook dialog box.
- To create a copy of a notebook, select the notebook and click **Duplicate**. The duplicate copy of the selected notebook is created, and listed in the Notebooks page with the suffix `_1` in the notebook name.
- To save a notebook as a template, select the notebook and click **Save as Template**. You can save the template in **Personal** or **Shared** under Templates.
- To delete a notebook, select it and click **Delete**.
- To import a notebook as `.json` files, click **Import**. Select the project and workspace in which to import the notebook.
To create versions of a notebook, select it and click Version. You can experiment with your notebook by creating versions of it, and revert to an older version by clicking Revert Version.

To open a notebook and run it, click the notebook. The notebook opens in the edit mode.

Create a Notebook
A notebook is a web-based interface for data analysis, data discovery, data visualization and collaboration.

Edit Your Notebook
After creating a notebook, you must open the notebook and write commands to fetch data from the data source, and run it for data discovery and data visualization.

Save Notebooks as Templates
You can save an existing notebook as a template in Personal or in Shared.

Version a Notebook
You can version or create a backup a notebook, experiment on it, and revert to the original notebook, if required.

Set Output Format in Notebooks
Oracle Machine Learning Notebooks allow you to preformat query output in notebooks.

Related Topics
• About Interpreter Bindings and Notebooks
An interpreter is a plug-in that allows you to use a specific data processing language at the backend. For the Zeppelin Notebooks in Oracle Machine Learning, you use the sql and pl/sql interpreters within an Oracle Database interpreter group and the md (MarkDown) interpreter for plain text formatting syntax so that it can be converted to HTML.

Create a Notebook
A notebook is a web-based interface for data analysis, data discovery, data visualization and collaboration.

Before you create a notebook, you must assign it to a project, which resides inside a workspace.

Whenever you create a notebook, it has an interpreter settings specification. The notebook contains an internal list of bindings that determines the order of the interpreter bindings. To create a notebook:

1. In the Oracle Machine Learning home page, click Notebooks. The Notebooks page opens.
2. In the Notebooks page, click Create.
   
   The Create Notebook window appears.
3. In the Name field, provide a name for the notebook.
4. In the Comments field, enter comments, if any.
5. In the Connections field, select a connection in the drop-down list. By default, the Global Connection Group is assigned.
6. Click OK.
Your notebook is created. You can now use the notebook to fetch data for data
discovery and data analysis.

**Related Topics**

- **Set Output Format in Notebooks**
  Oracle Machine Learning Notebooks allow you to preformat query output in
  notebooks.

- **Edit your Notebook**
  After creating a notebook, you must open the notebook and write commands to
  fetch data from the data source, and run it for data discovery and data
  visualization.

- **About Interpreter Bindings and Notebooks**
  An interpreter is a plug-in that allows you to use a specific data processing
  language at the backend. For the Zeppelin Notebooks in Oracle Machine
  Learning, you use the sql and pl/sql interpreters within an Oracle Database
  interpreter group and the md (MarkDown) interpreter for plain text formatting syntax
  so that it can be converted to HTML.

**Edit Your Notebook**

After creating a notebook, you must open the notebook and write commands to fetch
data from the data source, and run it for data discovery and data visualization.

Set the context with a project with which your notebook is associated.

You can edit an existing notebook in your project. To edit an existing notebook:

1. In Oracle Machine Learning home page, select the project in which your notebook
   is available.

2. Go to the Oracle Machine Learning navigator, and select **Notebooks**.
   Alternatively, you can click the **Notebooks** quick link in the home page.
   In the right pane, all notebooks that are available in the project are listed.

3. Click the notebook that you want to open and edit.
   The selected notebook opens in edit mode.

4. In edit mode, you can perform the following tasks:
   - Write code to fetch data
   - Run paragraphs. Click the run icon 🔁 to run one or all paragraphs in the
     notebook.
   - Export notebooks. Click the export icon 🔄 to export the notebook.
   - Set order for interpreter bindings. Click the gear icon ⚙ to set the order for
     interpreter bindings for the notebook.
   - View list of shortcut keys. Click the keyboard shortcut icon ☰ to view the list
     of keyboard shortcuts.
   - Add dynamic forms such as the Text Input form, Select form, Check box form
     for easy selection of inputs and easy filtering of data in your notebook. Oracle
     Machine Learning supports the following Apache Zeppelin dynamic forms:
     - Text Input form — Allows you to create a simple form for text input.
– Select form — Allows you to create a form containing a range of values that the user can select.
– Check Box form — Allows you to insert check boxes for multiple selection of inputs.

**Note:**
The Apache Zeppelin dynamic forms are supported only on SQL interpreter notebooks.

5. Once you have finished editing the notebook, click **Back**. This takes you back to the Notebook page.

- **Run a Notebook**
  A notebook can contain many paragraphs. You can run one or all the paragraphs in a notebook.

- **Export a Notebook**
  You can export a notebook as a .json (JavaScript Object Notation) file, and later import it in to the same or a different environment.

- **Import a Notebook**
  You can import notebooks across Pluggable Databases (PDBs) into your workspace.

- **Create Text Input Forms in Notebooks**
  The Text Input form allows you to dynamically retrieve values as defined in the notebook.

- **Create Select Forms in Notebooks**
  The Select Form allows you to select input values from a list of values, and dynamically retrieve the selected values as defined in the paragraph.

- **Create Check Box Forms in Notebooks**
  The Check Box Form supports multiple selection of inputs in a paragraph. The inputs are available as check box options in the notebook.

### Run a Notebook

A notebook can contain many paragraphs. You can run one or all the paragraphs in a notebook.

There are two types of notebook paragraphs:

- **%sql** - Supports standard SQL statements. In %sql the results of a SELECT statement are directly displayed in a Zeppelin table viewer, with access to other visualization options. Use the options in the chart settings to perform groupings, summation, and other operations.

- **%script** - Supports both SQL statements and PL/SQL. In %script, the results of a SELECT statement are provided as text string output.
To run a **Group By** on all your data, then it is recommended to use SQL scripts to do the grouping in the database, and return the summary information for charting in the notebook. Grouping at the notebook level works well for small sets of data. If you pull too much data to the notebook, you may encounter issues due to insufficient memory. You can set the row limit for your notebook by using the option **Render Row Limit** in the Connections Group page.

To fetch and visualize data in a notebook:

1. In the Notebook page, click the notebook that you want to run.
   
The notebook opens in edit mode.
2. Type the SQL statement to fetch data from an Oracle Database. For example, type `SELECT * FROM TABLENAME` and click ![Execute]. Alternatively, press Shift+Enter keys to run the notebook.

   **Note:**
   
   Notebooks must be opened as a regular user, that is, a non-administrator user. The **Run** notebook option is not available to the Administrator.

   This fetches the data in the notebook.
3. The data is displayed in the output of the paragraph.
   
   A paragraph is a notebook component in which you can write SQL statements and run scripts. A paragraph has an input section and an output section. In the input section, you specify the interpreter to run along with the text. This information is sent to the interpreter to be executed. The results of the interpreter appear in the output section. In the SQL interpreter, the output section of the paragraph comprises a charting component that displays the results in graphical output. The chart interface allows you to interact with the output in the notebook paragraph. You have the option to run and edit single a paragraph or all paragraphs in a notebook.

   For Table Options, click **settings** and select:
   
   • **useFilter**: To enable filter for columns.
   • **showPagination**: To enable pagination for enhanced navigation.
   • **showAggregationFooter**: To enable a footer to display aggregated values.

   You can also sort the columns by clicking the down arrow next to the column name.
To visualize the tabular data, click the respective icons for each of the each graphical representation, as shown here:

- Click [ ] to represent the data in a Bar Chart.
- Click [ ] to represent the data in a Pie Chart.
- Click [ ] to represent the data in an Area Chart.
- Click [ ] to represent the data in a Line Chart.
- Click [ ] to represent the data in a Scatter Chart.

Related Topics

- **Edit Oracle Database Interpreter Connection**
  When defining an Oracle Database interpreter connection, a reference to a compute resource is created. This reference contains all connection-related information about the interpreter.
• **Edit Oracle Database Interpreter Connection**

When defining an Oracle Database interpreter connection, a reference to a compute resource is created. This reference contains all connection-related information about the interpreter.

---

**Export a Notebook**

You can export a notebook as a `.json` (JavaScript Object Notation) file, and later import it into the same or a different environment.

To export a notebook:

1. In the Notebooks page, click the notebook that you want to export.
   
The notebook opens in the notebook editor.

2. In the top panel of the notebook editor, click `.json`. The notebook is saved to your local folder as a `.json` file.

**Related Topics**

• **About Interpreter Bindings and Notebooks**

An interpreter is a plug-in that allows you to use a specific data processing language at the backend. For the Zeppelin Notebooks in Oracle Machine Learning, you use the `sql` and `pl/sql` interpreters within an Oracle Database interpreter group and the `md` (MarkDown) interpreter for plain text formatting syntax so that it can be converted to HTML.

---

**Import a Notebook**

You can import notebooks across Pluggable Databases (PDBs) into your workspace.

---

**Note:**

Starting in Oracle Database 20c, "database" refers specifically to the data files of a multitenant container database (CDB), pluggable database (PDB), or application container.

To import a notebook:

1. In Oracle Machine Learning home page, click **Notebooks**.
2. In the Notebooks page, click **Import**.

   This opens the **File Upload** dialog. Browse and select the notebook that you want to import.

---

**Note:**

You must have the notebook saved as a `.json` file to import it. You can import notebooks exported from non-Oracle Apache Zeppelin environments, but only paragraphs types that are supported may be run.

3. In the File Upload dialog, browse and select the `.json` file and click **Open**.

   This imports the notebook file into your workspace.
4. Click the imported notebook to open it. In the notebook page, click the gear icon to view the interpreter bindings.

Related Topics
- About Interpreter Bindings and Notebooks
  An interpreter is a plug-in that allows you to use a specific data processing language at the backend. For the Zeppelin Notebooks in Oracle Machine Learning, you use the sql and pl/sql interpreters within an Oracle Database interpreter group and the md (MarkDown) interpreter for plain text formatting syntax so that it can be converted to HTML.
- Database and Instance

Create Text Input Forms in Notebooks

The Text Input form allows you to dynamically retrieve values as defined in the notebook.

To create a Text Input form:
1. Open the notebook in which you want to add the Text Input form.
2. In a SQL statement, define the Text Input form by using the syntax:
   
   ```
   ${formName}
   ```

   For example, run the SQL statement:
   ```
   SELECT * FROM ALL_OBJECTS WHERE OBJECT_TYPE = '${OBJ}';
   ```

   In this example,
   - The form name is `obj`
   - The table name is ALL_OBJECTS
   - The column name is OBJECT_TYPE

   Here, the text form `obj` is created for the column OBJECT_TYPE in the table ALL_OBJECTS. You can enter different values in the form field `obj` and run the notebook to retrieve the corresponding values in the column OBJECT_TYPE.

3. Run the paragraph. The notebook now displays the text input form field `obj`, as shown in the screenshot. You can enter values in the `obj` field, and run the notebook to retrieve the corresponding values for the column OBJECT_TYPE in the table ALL_OBJECTS.

   - If you enter `TABLE` in the `obj` field, and run the notebook, then the notebook retrieves `TABLE` in the column OBJECT_TYPE, as shown in the screenshot.

   - If you enter `VIEW` in the `obj` form field and run the notebook, then the notebook retrieves the value VIEW in the column OBJECT_TYPE, as shown
4. You can also assign default values in the form by using the syntax:

${formName=defaultValue}

To assign a default value to the Text Input form, modify the SQL statement to:

SELECT * FROM ALL_OBJECTS WHERE OBJECT_TYPE = '(${obj=TABLE})'

Here, the default value assigned to the form is TABLE. Once you run the paragraph, the default value TABLE will be retrieved in the column OBJECT_TYPE, as shown in the screenshot.

Create Select Forms in Notebooks

The Select Form allows you to select input values from a list of values, and

dynamically retrieve the selected values as defined in the paragraph.

To create a Select form:

1. Open the notebook in which you want to add the text input form.

2. In a SQL statement, define the Select form by using the syntax:

   ${formName=defaultValue,option1|option2...}.

   For example, run the SQL statement:

   SELECT * FROM ALL_OBJECTS WHERE OBJECT_TYPE = '(${OBJ=INDEX,INDEX|TABLE|VIEW|SYNONYM})';

   In this example,

   - The form name is obj
   - The list of available values are INDEX, TABLE, VIEW, SYNONYM.
• The table name is ALL_OBJECTS
• The column name is OBJECT_TYPE

Select any values from the drop-down list in the obj form. The selected value will be retrieved in the OBJECT_TYPE column in the ALL_OBJECTS table.

Create Check Box Forms in Notebooks

The Check Box Form supports multiple selection of inputs in a paragraph. The inputs are available as check box options in the notebook.

To create a Check Box Form:
1. Open the notebook in which you want to add the Check Box Form.
2. In a SQL statement, define the Check Box form by using the syntax:
   `${checkbox:formName=defaultValue1|defaultValue2...,option1|option2...}`

   For example, run the SQL statement:
   ```sql
   SELECT ${checkbox:whichcolumn=OWNER|OBJECT_TYPE, OWNER|OBJECT_NAME|OBJECT_TYPE|CREATED|STATUS} FROM ALL_OBJECTS WHERE OBJECT_TYPE IN ('VIEW', 'TABLE', 'INDEX', 'SYNONYM');
   ```

   In this example,
   • The Check Box form is WhichColumn
   • The multiple selection options available in the check boxes are OWNER, OBJECT_NAME, OBJECT_TYPE, CREATED, and STATUS
   • The fields OWNER and OBJECT_TYPE are defined as default
   • The table name is ALL_OBJECTS
   • The columns that are configured for display are OWNER, OBJECT_NAME, OBJECT_TYPE, CREATED, and STATUS
3. Run the notebook. The Check Box form called WhichForm is available in the notebook, as shown in the screenshot.

Save Notebooks as Templates

You can save an existing notebook as a template in Personal or in Shared.

To save a notebook as a template:
1. In the Notebooks page, select the notebook that you want to save as template and click Save as Template.
   The Save as Template dialog box opens.
2. In the Name field, enter a name for the notebook template.
3. In the **Comments** field, enter comments, if any.
4. In the **Tags** field, enter tags for the template.
5. In Save To, select:
   - **Personal**: If you want to save this notebook template to Personal.
   - **Shared**: If you want to save and share this notebook template.

**Version a Notebook**

You can version or create a backup a notebook, experiment on it, and revert to the original notebook, if required.

**To version a notebook:**

1. In the Notebooks page, select the notebook that you want to version and click **Version**.
   The Versions page opens.
2. In the Versions page for the selected notebook, click + **Version**.
   The Create Versions dialog box opens.
3. In the Create Versions dialog box, enter comments for the specific version of your notebook, and click **OK**.
4. The versioned notebook is now listed in the Versions page. You can perform the following tasks:
   - Click **Revert Version** to restore the older version of your notebook.
   - Click **Delete** to delete the selected version of your notebook.
   - Click **New Notebook** to create a new notebook from the selected notebook version.

**Set Output Format in Notebooks**

Oracle Machine Learning Notebooks allow you to preformat query output in notebooks.

**To preformat query output, you must use the command** `SET SQLFORMAT` **as follows:**

1. Open a notebook in Oracle Machine Learning.
2. Type the command:
   ```
   %script
   SET SQLFORMAT format_option
   ```
   For example, if you want the output in ansiconsole format, then type the command followed by the SQL query as:
   ```
   SET SQLFORMAT ansiconsole;
   SELECT * FROM HR.EMPLOYEES;
   ```
   Here, the output format is **ansiconsole**, and the table name is **HR.EMPLOYEES**.
Output Formats Supported by SET SQLFORMAT Command

By using the `SET SQLFORMAT` command, you can generate the query output in a variety for formats.

The available output formats are:

- **CSV** — The CSV format produces standard comma-separated variable output, with string values enclosed in double quotes. The syntax is:
  ```
  %script
  SET SQLFORMAT CSV
  ```

- **HTML** — The HTML format produces the HTML for a responsive table. The content of the table changes dynamically to match the search string entered in the text field. The syntax is:
  ```
  %script
  SET SQLFORMAT HTML
  ```

- **XML** — The XML format produces a tag based XML document. All data is presented as `CDATA` tags. The syntax is:
  ```
  %script
  SET SQLFORMAT XML
  ```

- **JSON** — The JSON format produces a JSON document containing the definitions of the columns along with the data that it contains. The syntax is:
  ```
  %script
  SET SQLFORMAT JSON
  ```

- **ANSICONSOLE** — The ANSICONSOLE format resizes the columns to the width of the data to save space. It also underlines the columns, instead of separate line of output. The syntax is:
  ```
  %script
  SET SQLFORMAT ANSICONSOLE
  ```
• INSERT — The INSERT format produces the INSERT statements that could be used to recreate the rows in a table. The syntax is:

```
%script
SET SQLFORMAT INSERT
```

• LOADER — The LOADER format produces pipe delimited output with string values enclosed in double quotes. The column names are not included in the output. The syntax is:

```
%script
SET SQLFORMAT LOADER
```

• FIXED — The FIXED format produces fixed width columns with all data enclosed in double-quotes. The syntax is:

```
%script
SET SQLFORMAT FIXED
```

• DEFAULT — The DEFAULT option clears all previous SQLFORMAT settings, and returns to the default output. The syntax is:

```
%script
SET SQLFORMAT DEFAULT
```

**Note:**
You can also execute this command without the format name `DEFAULT` by simply typing `SET SQLFORMAT`.

• DELIMITED — The DELIMITED format allows you to manually define the delimiter string, and the characters that are enclosed in the string values. The syntax is:

```
%script
SQLFORMAT DELIMITED delimiter left_enclosure right_enclosure
```

For example,

```
%script
SET SQLFORMAT DELIMITED ~del~ " "
SELECT * FROM emp WHERE deptno = 20;
```

**Output:**

"EMPNO"~del~"ENAME"~del~"JOB"~del~"MGR"~del~"HIREDATE"~del~"SAL"~del~"COMM"~del~"DEPTNO"

In this example, the delimiter string is `~del~` and string values such as `EMPNO`, `ENAME`, `JOB` and so on, are enclosed in double quotes.

### About Interpreter Bindings and Notebooks

An interpreter is a plug-in that allows you to use a specific data processing language at the backend. For the Zeppelin Notebooks in Oracle Machine Learning, you use the `sql` and `pl/sql` interpreters within an Oracle Database interpreter group and the `md`
Notebooks contain an internal list of bindings that define the order of the interpreter bindings in an interpreter group. The default order of interpreter bindings in the Oracle Database interpreter group is:

- **Low**: It provides the least level of resources to each SQL statement, but supports the maximum number of concurrent SQL statements. The interpreter with low priority is listed on the top of the interpreter list, and hence, is the default.
- **Medium**: It provides a lower level of resources to each SQL statement potentially resulting in a lower level of performance, but supports more concurrent SQL statements.
- **High**: It provides the highest level of resources to each SQL statement resulting in the highest performance, but supports the minimum number of concurrent SQL statements.

With respect to interpreter bindings, you can perform the following tasks:

- **Bind and unbind interpreters**: If you do not bind any specific interpreter to your notebook, then you get the error message:

  ```
  databasename_servicename not found
  ```

- **Set and re-order interpreter bindings**: You may want to set and re-order interpreter bindings if you want to use a specific interpreter for a specific paragraph in a notebook. In that case, you have to select the specific interpreter for that paragraph.

- **Change the interpreter binding for any specific paragraph in a notebook**

You must note the interpreter binding order in the following scenarios:

- **Notebook creation**: When you create a notebook, the notebook inherits the initial interpreter binding order, which is low (default), medium, high.

- **Notebook import**: When importing a notebook, the notebook inherits the defined interpreter bindings. However, after you import a notebook, ensure to check the order of the interpreter bindings.

- **Notebook export**: When exporting a notebook, the notebook inherits the defined interpreter bindings.

- **Notebook creation from templates**: When you create a notebook from templates, the notebook inherits the default order of interpreter bindings.

- **Set Interpreter Bindings for Notebooks**

  You must bind a notebook to an interpreter to fetch data from the database or any data source. A default set of interpreter bindings is available.

- **Change Interpreter Bindings for Specific Paragraphs in a Notebook**

  The interpreter binding order that is set for a notebook applies to all the paragraphs in that notebook. However, you can change the interpreter binding order for any specific paragraph in the notebook.

- **Validate Interpreter Bindings**

  You can use a SQL statement to fetch the information about interpreter binding.
Set Interpreter Bindings for Notebooks

You must bind a notebook to an interpreter to fetch data from the database or any data source. A default set of interpreter bindings is available.

You can set the order of interpreter bindings if you have more than one set available. To set the order of interpreter bindings:

1. In the Notebook page, click the notebook for which you want to set the interpreter bindings.
   The notebook opens in edit mode.
2. Click the gear icon at the top panel.

   The Settings pane opens listing the interpreter bindings for the notebook.
3. Drag and drop the interpreters to reorder the interpreter bindings. The first interpreter on the list is the default. The order of interpreter bindings is:
   - Low (Default): It provides the least level of resources to each SQL statement, but supports the maximum number of concurrent SQL statements. The interpreter with low priority is listed on the top of the interpreter list, and hence, is the default.
   - Medium: It provides a lower level of resources to each SQL statement potentially resulting in a lower level of performance, but supports more concurrent SQL statements.
   - High: It provides the highest level of resources to each SQL statement resulting in the highest performance, but supports the minimum number of concurrent SQL statements.

   This is the initial binding order of the interpreters. You can change the order of the interpreter bindings.
4. Click Save.

   Related Topics
   - Manage Concurrency and Priorities on Autonomous Data Warehouse Cloud
   - Predefined Database Service Names for Autonomous Data Warehouse

Change Interpreter Bindings for Specific Paragraphs in a Notebook

The interpreter binding order that is set for a notebook applies to all the paragraphs in that notebook. However, you can change the interpreter binding order for any specific paragraph in the notebook.

To change the interpreter binding for a specific paragraph in a notebook:

1. Open the notebook and click the gear icon to view the interpreter bindings and its order.
In this example, all the three SQL interpreters are bound to the notebook, and the interpreter with low resource allocation `cwdp_low %sql` is the default, as it is the first interpreter on the list. The MarkDown interpreter is not bound to the notebook.

**Note:**

The names of the interpreters are in the format `databasename_servicename, databasename_servicename, and databasename_servicename` which is the same as the interpreter binding order name. In this example, the interpreter names are `cwdp_low % sql`, `cwdp_medium % sql`, and `cwdp_high % sql`. 
2. To change the interpreter bindings order for a particular paragraph in the notebook:

   - Scroll down to the paragraph for which you want to change the interpreter
   - Invoke the interpreter with the specific binding
   - Run the paragraph

For example, invoke the interpreter with medium resource allocation by typing
%cwdp_medium for the first paragraph in the notebook, and run the paragraph. In
this example, cwdp is the database name.
Notice that the first paragraph runs without any error after changing the interpreter binding. The second paragraph in this notebook has the default binding.

3. Validate the interpreter binding for first paragraph of this notebook by typing the SQL statement `SELECT SYS_CONTEXT ('USERENV', 'SERVICE_NAME') FROM DUAL;` 

The SQL statement returns the following information about the interpreter with medium binding:
In this example, with reference to the screenshot:

- **LGKFDTOOBOQK48I** is the tenant name
- **CWDP** is the database name
- **medium** is the service name
- **adwc.oraclecloud.com** is the domain

**Note:**

For the rest of the paragraphs in this notebook, the interpreter binding is the default. You may validate the bindings for each paragraph by running step 3.

This completes the task of changing the interpreter binding for a particular paragraph in the notebook. The rest of the paragraphs in the notebook have the default binding for the interpreter.

### Validate Interpreter Bindings

You can use a SQL statement to fetch the information about interpreter binding.

To validate the interpreter binding of a notebook:

1. Open the notebook for which you want to check the interpreter binding.
2. Run the following SQL statement:

   ```sql
   SELECT SYS_CONTEXT ('USERENV', 'SERVICE_NAME') FROM DUAL;
   ```
The SQL statement queries the service name that is used to run the query.

3. Click Run.

The query returns the information about the interpreter, the priority of the binding, and the service name. The result is displayed in the following format

\[
\text{tenantname__databasename__servicename.domain}
\]

as shown in the screenshot. Here, HDY7RUSKGDMPHN2 is the tenant name, PDB2 is the database name, low is the service name, and adwc.oraclecloud.com is the domain name.

```sql
%script
SELECT SYS_CONTEXT ('USERENV', 'SERVICE_NAME') FROM DUAL;
```

Here, `HDY7RUSKGDMPHN2_PDB1_low.adwc.oraclecloud.com` is the result.

Use the Scratchpad

You can use the SQL scratchpad to run your SQL statements and SQL script without opening or creating a notebook.

To open and use the scratchpad:

- Click Run SQL Scripts. This opens the SQL Script Scratchpad where you can create and run SQL scripts.
- Click Run SQL Statements. This opens the SQL Query Scratchpad where you can run SQL statements.
- Run SQL Scripts
  A SQL script is a set of SQL statements. You can create SQL scripts in Oracle Machine Learning Notebooks, save them as `.json` files in your system. You can share the notebook.json files with other users as well.
- Run SQL Statements
  SQL or Structured Query Language is the standard language for relational database management systems. You can use SQL statements to perform tasks such as retrieving data from a database, updating data on a database, and so on. Some examples of SQL statements are `SELECT`, `INSERT`, `UPDATE`, `DELETE`, `CREATE`, and `DROP`.

Run SQL Scripts

A SQL script is a set of SQL statements. You can create SQL scripts in Oracle Machine Learning Notebooks, save them as `.json` files in your system. You can share the notebook.json files with other users as well.

To run SQL scripts in Oracle Machine Learning:
1. Log in to Oracle Machine Learning using your credentials provided by your Oracle Machine Learning Administrator.

2. In the Oracle Machine Learning home page, click Run SQL Script. The SQL Query Scratchpad opens.

3. In the SQL Query Scratchpad, after %script, press enter.

4. In the next line in the SQL Query Scratchpad, type the SQL script as shown in the screenshot. The SQL script in this example, contains two parts:

```
CREATE TABLE small_table
(
NAME VARCHAR(200),
ID1 INTEGER,
ID2 VARCHAR(200),
ID3 VARCHAR(200),
ID4 VARCHAR(200),
TEXT VARCHAR(200)
);
BEGIN
FOR i IN 1..100 LOOP
   INSERT INTO small_table VALUES ('Name____'||i, i, 'ID1_'||i,'ID2_||i,'ID3_||i,'ID4_||i,'TEXT_||i);
END LOOP;
COMMIT;
END;
```

- The first part of the script contains the SQL statement CREATE TABLE to create a table named small_table. It defines the table name, table column, data types, and size. In this example, the column names are NAME, ID1, ID2, ID3, ID4, and TEXT.
- The second part of the script begins with the keyword BEGIN. It inserts 100 rows in to the table small_table.

5. Once you have finished writing the SQL script, click Run.

6. After you have created the SQL script, you can share it with other users:
   - **Export SQL script:** In the SQL Query Scratchpad, click the export icon to save it as a .json file in your system.
   - **Import SQL script:** In the Notebooks page, click Import to import the SQL file saved as a .json file in to your notebook.
After the SQL script is successfully imported, it is listed in the Notebooks page. Click the SQL script that is listed here to open and edit it in the Notebook editor. You can also save the SQL script as a notebook template in Personal Templates, Shared, or in the Examples.

**Related Topics**
- Restrictions on SQL Commands
- Restrictions for Database Options
- Restrictions for Database Initialization Parameters

**Run SQL Statements**

SQL or Structured Query Language is the standard language for relational database management systems. You can use SQL statements to perform tasks such as retrieving data from a database, updating data on a database, and so on. Some examples of SQL statements are `SELECT`, `INSERT`, `UPDATE`, `DELETE`, `CREATE`, and `DROP`.

To run SQL statements:

1. In the Oracle Machine Learning home page, click **Run SQL Statement**. The SQL Query Scratchpad opens.
2. In the SQL Query Scratchpad, type `%sql` and press enter.
3. In the next line, enter the SQL statement that you want to run. If you want to fetch data about product sales, then type `SELECT * FROM tablename`. For example, `SELECT * FROM SH.SALES;` where, `SH` is the schema name and `SALES` is the table that contains the data about product sales in the database.
4. To run the SQL statement, click or press Shift+Enter. Once the statement runs successfully, Oracle Machine Learning fetches the data from the database and displays it in a tabular format.
5. If you want to visualize the data in a different graphical output, then click the icon for the corresponding graph.

**Collaborate in Oracle Machine Learning**

Two or more users can collaborate and share Oracle Machine Learning notebooks with other users.

You can collaborate by:
- Granting Access to Projects of Another User
- Using the Export Option
- Using Oracle Machine Learning Notebook Templates

[Collaborating in Oracle Machine Learning]
Use Templates to Collaborate with Users

In the Oracle Machine Learning Templates UI, you can collaborate with other users by sharing your work, publishing your work as reports, and by creating notebooks from templates. You can store your notebooks as templates, share notebooks, and provide sample templates to other users.

Note:
You can also collaborate with other Oracle Machine Learning Notebook users by providing access to your workspace. The authenticated user can then access the projects in your workspace, and access your notebooks. The access level depends on the permission type granted - Manager, Developer, or Viewer. For more information about collaboration among users, see Quickstart Tutorial: Collaborating in Oracle Machine Learning

- **Use the Personal Templates**
  Personal Templates lists the notebook templates that you have created.

- **Use the Shared Templates**
  In the Shared Templates, you can share notebook templates with all authenticated users the notebook templates you create from existing notebooks available in Templates.

- **Use the Example Templates**
  The Example Templates page lists the pre-populated Oracle Machine Learning notebook templates. You can view and use these templates to create your notebooks.

Use the Personal Templates

Personal Templates lists the notebook templates that you have created.

You can perform the following tasks:

- View selected templates in read-only mode.
- Create new notebooks from selected templates.
- Edit selected templates.
- Share selected notebook templates in Shared Templates.
- Delete selected notebook templates.
- **Create Notebooks from Templates**
  You can create new notebooks from an existing template, and store them in Personal Templates for later use.
Create Notebooks from Templates

You can create new notebooks from an existing template, and store them in Personal Templates for later use.

You must select a notebook template.

To create a new notebook from a template:

1. In the Personal Templates page, select the template based on which you want to create the notebook, and click **New Notebook**.
   
   The Create Notebook dialog box opens.

2. In the **Name** field, provide a name for the notebook.

3. In the **Comments** field, enter comments, if any.

4. In the **Project** field, select the project in which you want to save your notebook.

5. In the **Connection** field, the default connection is selected.

6. Click **OK**.

   The notebook is created, and is available in the Notebooks page.

Share Notebook Templates

You can share templates from Personal Templates. You can also share templates for editing.

To share a template:

1. Select the notebook template in Personal Templates and click **Share**.

   The Save to Shared Templates dialog box opens.

2. In the **Name** field, enter a new name for the template.

3. In the **Comments** field, provide comments, if any.

4. In the **Tags** field, enter tags separated by commas. To enable easy searching, use descriptive tags.

5. Click **OK**.

   Once the template is successfully created and shared, a message appears stating that the template is created in Shared.

Edit Notebook Templates Settings

You can modify the settings of an existing notebook template in Personal Templates.

To edit notebook template settings:

1. Select the notebook template in Personal Templates and click **Edit Settings**.
The Edit Template dialog box opens.

2. In the **Name** field, edit the name, as applicable.
3. In the **Comments** field, edit the comments, if any.
4. In the **Tags** field, edit the tags, as applicable.
5. Click **OK**.

### Use the Shared Templates

In the Shared Templates, you can share notebook templates with all authenticated users the notebook templates you create from existing notebooks available in Templates.

The Shared Templates page tracks notebook templates when you perform the following:

- Like templates
- Create notebooks from templates
- View templates

The Shared Templates page displays the following information about the templates:

- Template name
- Description
- Number of likes
- Number of creations
- Number of static views

You can perform the following tasks:

- Create templates by clicking **New Notebook**
- Edit template settings by clicking **Edit Settings**
- Delete any selected template by clicking **Delete**
- Search templates by Name, Tag, Author
- Sort templates by Name, Date, Author, Liked, Viewed, Used
- View templates by clicking **Show Liked Only** or **Show My Items Only**
- **Create Notebooks from Templates**
  You can create new notebooks from an existing template, and store them in Personal Templates for later use.
- **Edit Notebook Templates Settings**
  You can modify the settings of an existing notebook template in Personal Templates.

### Create Notebooks from Templates

You can create new notebooks from an existing template, and store them in Personal Templates for later use.

You must select a notebook template.
To create a new notebook from a template:

1. In the Personal Templates page, select the template based on which you want to create the notebook, and click **New Notebook**.
   The Create Notebook dialog box opens.
2. In the **Name** field, provide a name for the notebook.
3. In the **Comments** field, enter comments, if any.
4. In the **Project** field, select the project in which you want to save your notebook.
5. In the **Connection** field, the default connection is selected.
6. Click **OK**.
   The notebook is created, and is available in the Notebooks page.

---

**Edit Notebook Templates Settings**

You can modify the settings of an existing notebook template in Personal Templates.

To edit notebook template settings:

1. Select the notebook template in Personal Templates and click **Edit Settings**.
   The Edit Template dialog box opens.
2. In the **Name** field, edit the name, as applicable.
3. In the **Comments** field, edit the comments, if any.
4. In the **Tags** field, edit the tags, as applicable.
5. Click **OK**.

---

**Use the Example Templates**

The Example Templates page lists the pre-populated Oracle Machine Learning notebook templates. You can view and use these templates to create your notebooks.

The Example Templates page displays the following information about the templates:

- Template name
- Description
- Number of likes. Click **Likes** to mark it as liked.
- Number of static views
- Number of uses

You cannot alter any templates in the Example Templates page. The search options are:

- Search templates by Name, Tag, Author
- Sort templates by Name, Date, Author, Liked, Viewed, Used
- View templates that are liked by clicking **Show Liked only**
- **Create a Notebook from the Example Templates**
  In Oracle Machine Learning Example Templates, you can create a notebook from the available templates.
Create a Notebook from the Example Templates

In Oracle Machine Learning Example Templates, you can create a notebook from the available templates.

To create a notebook:

1. In the Example Templates page, select the template based on which you want to create a notebook.
2. Click **New Notebook**.
   
   The Create Notebook dialog box opens.
3. In the **Create Notebook** window, the name of the selected template appears. In the **Name** field, you can change the notebook name.
4. In the **Comment** field, if any comment is available for the template, then it is displayed. You can edit the comment.
5. In the **Project** field, click the edit icon.
6. Select the project in which you want to save the notebook.
7. In the **Connection** field, the default connection is selected.
8. Click **OK**.

The notebook is created and is available in the Notebooks page.

**Example Templates**
Oracle Machine Learning Notebooks provide you the following notebook Example templates that are based on different machine learning algorithms. The Example templates are processed in Oracle Autonomous Database.
Example Templates

Oracle Machine Learning Notebooks provide you the following notebook Example templates that are based on different machine learning algorithms. The Example templates are processed in Oracle Autonomous Database.

You can create your notebook based on any of these templates:

Figure 4-1    Example Templates

- **Anomaly Detection**: Use the Anomaly Detection Example Template notebook to detect anomalous records, customers or transactions in your data. This template uses the unsupervised learning algorithm 1-Class Support Vector Machine. The notebook template builds a 1-Class Support Vector Machine (SVM) model.

- **Association Rules**: Use the Association Rules Example Template notebook for market basket analysis of your data, or to detect co-occurring items, failures or events in your data. This template uses the apriori Association Rules model using the `SH.SALES` schema data.

- **Attribute Importance**: Use the Attribute Importance Example Template notebook to identify key attributes that have maximum influence over the target attribute. The target attribute in the build data of a supervised model is the attribute that you want to predict. The template builds an Attribute Importance model using the `SH` schema data.

- **Classification Prediction Model**: Use the Classification Prediction Model Example Template notebook for predicting customer behavior and similar predictions. The template builds and applies the classification algorithm Decision Tree to build a Classification model based on the relationships between the predictor values and the target values. The template uses the `SH` schema data.

- **Clustering**: Use the Clustering Example Template notebook to identify natural clusters in your data. The notebook template uses the unsupervised learning k-Means algorithm on the `SH` schema data.

- **My First Notebook**: Use the My First Notebook Example Template notebook for basic machine learning functions, data selection and data viewing. This template uses the `SH` schema data.
• **Regression**: Use the Regression Example Template notebook to predict numerical values. This template uses multiple regression algorithms such as Generalized Linear Models (GLM).

• **Statistical Function**: Use the Statistical Function Example Template notebook for descriptive and comparative statistical functions. The notebook template uses SH schema data.

• **Time Series Forecasting**: Use the Time Series Forecasting Example Template notebook to build time series models on your time series data for forecasting. This Example notebook is based on the SH.SALES data. All computations are done inside Oracle Autonomous Database.
Get Started with Jobs

Jobs allow you to create schedules to run notebooks. In the Jobs page, you can create and schedule jobs, monitor job status, and view job logs as read-only notebooks.

- **About Jobs**
  The Jobs page lists all the jobs created, along with the job name, notebook, owner of the job, last start date, next run date, status, and schedule.

- **Create Jobs to Schedule Notebooks**
  You can create jobs to schedule your notebook with preferred scheduling settings.

- **View Job Logs**
  You can view the historical logs of any particular job in the Job Log interface.

About Jobs

The Jobs page lists all the jobs created, along with the job name, notebook, owner of the job, last start date, next run date, status, and schedule.

You can perform the following tasks:

**Figure 5-1  Jobs**

- **Edit jobs**: You can edit the metadata of any job listed in the Jobs page. Click **Edit** to edit the selected job.
- **Create jobs**: Click **Create** to create a new job to schedule your Notebook.
- **Duplicate jobs**: You can create a copy of an existing job listed in the Jobs page. Click **Duplicate** to make a copy of the selected job.
- **Stop jobs**: Click **Stop** to terminate a job that is currently running.
- **Start jobs**: The **Start** button is enabled only for jobs that are in **Scheduled** status. Click **Start** to start a scheduled job. The **Start** option is not applicable for the following conditions:
  - Jobs that have already completed its scheduled run cannot be re-started.
  - Jobs that have failed more than the allowed number of times, and are currently in **Broken** status, cannot be re-started.
- **Delete jobs**: Click **Delete** to delete any job listed in the Jobs page.
Create Jobs to Schedule Notebooks

You can create jobs to schedule your notebook with preferred scheduling settings.

To create jobs, enter the following details in the Create Jobs dialog box:

1. In the Jobs page, click **Create**. The Create Jobs dialog box opens.

2. In the **Name** field, enter a name for the job. The number of characters in the job name must not exceed 128 bytes.

3. In the **Notebook** field, click the search icon to select a notebook to create a job.
Note:

Only notebooks that are owned by the user or shared are available for selection.

4. In the **Start Date** field, click the date-time editor to set the date and time for your job to commence. Based on the selected date and time, the next run date is computed.

5. Optionally, in the **Repeat** section, select:
   - **Frequency**: To set the repeat settings and frequency. You can set the frequency in minutes, hours, days, week, and month.
   - **Custom**: To customize the job settings.

6. Optionally, in **Advanced Settings**, select one or more of the following options:
   - **Maximum Number of Runs**: To specify the maximum number of times the job must run before it is stopped. When the job reaches the maximum run limit, it will stop.
   - **Maximum Failures Allowed**: To specify the maximum number of times a job can fail on consecutive scheduled runs. When the maximum number of failures is reached, the next run date column in the Jobs UI will show an empty value to indicate the job is no longer scheduled to run. The Status column may show the status as *Failed*.
   - **Timeout in Minutes**: To specify the maximum amount of time a job should be allowed to run.

7. Click **OK**.

**View Job Logs**

You can view the historical logs of any particular job in the Job Log interface.

You can view a log in the read-only Notebook. To view job logs:

1. To view the history of a job, select the job and click ![View](View.png).
2. To delete a particular job log, select it and click ![Delete](Delete.png).
Get Started with Connection Groups

A connection group, also known as a Zeppelin interpreter set, is a collection of database connections.

- **About Connection Groups**
  In the Connection Group page, a user with Administrator role can manage your connections that constitute the connection group.

- **About Global Connection Group**
  The Global Connection Group is created automatically when a new database is provisioned.

- **Edit Oracle Database Interpreter Connection**
  When defining an Oracle Database interpreter connection, a reference to a compute resource is created. This reference contains all connection-related information about the interpreter.

### About Connection Groups

In the Connection Group page, a user with Administrator role can manage your connections that constitute the connection group.

You can **Edit**, and **Stop** one or more connections that are listed under a connection group in this page.

#### Note:

Only an Administrator user can manage connection groups.

The following information about the connections are available:

- **Name**: This is the name of the interpreter.
- **Default**: A check mark indicates whether the connection is the default connection or not.
- **Scope**: Indicates the scope of the connection.
- **Comment**: Displays any comment related to the interpreter.
- **Owner**: Displays the name of the user who created the connection.
- **Last Updated**: Indicates the date and time when the connection was last updated.

You can perform the following tasks:

- **Edit**: To edit the interpreter connection, select the connection and click **Edit**.
- **Stop**: To stop the interpreter connection, select the connection and click **Stop**.
- **Reset**: To reset the interpreter connection, click the connection group name. The connection group opens in a separate page, listing all the interpreter connections.
in the group. Select the connection you want to reset and click **Reset**. When you click **Reset**, then all connections supported by the interpreter are closed, and all notebooks using that connection are cancelled.

**Note:**
The **Reset** option is available to the Administrator only.

### About Global Connection Group

The Global Connection Group is created automatically when a new database is provisioned.

The Global Connection Group comprises the following:

- **Compute Resource definition** — Compute Resources are added or removed automatically when databases are provisioned for a tenant. Only one Compute Resource definition is provided per Oracle Machine Learning database installation.

**Note:**
The Compute Resource definition can be edited by the Administrator only.

- **Connection Group definition** — The Global Connection Group comprises a single connection of type **Global**. Only one Global Connection Group is allowed per tenant.

**Note:**
A Global Connection Group can be edited by the Administrator only.

### Edit Oracle Database Interpreter Connection

When defining an Oracle Database interpreter connection, a reference to a compute resource is created. This reference contains all connection-related information about the interpreter.

Compute Resources for an Oracle Database interpreter is defined by your service. You can edit the following:

**Note:**
You must have the Administrator role to edit these fields.

1. **Name** — You can edit the name of the interpreter editor here. This is useful if you have several definitions of the same interpreter type in the same interpreter set. By specifying a name, you can turn on or turn off the specific binding to a notebook.
2. **Type**: This is a non-editable field. It indicates the connection type.

3. **Binding Mode**: This is a non-editable field. It defines the behavior of the interpreter instance in memory, and how the resources are shared. By default, the Binding Mode of the Global Connection Group is set to Scoped. It ensures that each notebook creates a new interpreter instance in the same interpreter process.

4. **Row Render Limit**: This determines the number of rows to be displayed in the paragraph results when fetching a data structure that can be presented as a table or graph using the Zeppelin built-in plotting service. You must consider the browser capabilities when modifying this setting. The default limit is 1000.

   **Note:**
   Zeppelin plotting service works with data that is fetched previously to the client-side for a snapper UI.

5. **Comments**: Enter any information related to the interpreter not exceeding 1000 characters.

   **Note:**
   You must have Administrator role to edit this field.

6. In the Compute Resource section, the **Resources** field indicates the priority of the compute resource. This is a non-editable field.

7. In the Database section, you can specify additional settings related to PL/SQL DBMS output. Select **Enabled** to allow the PL/SQL interpreter to display the messages sent to the DBMS_OUTPUT in the paragraph results.

8. Click **Save**.
Use Notebook Sessions

The Notebook Sessions page provides you with an overview of your notebooks, and allows you to manage notebook sessions from your workspace or in a workspace where you have collaboration rights.

- **About Notebook Sessions**
  The Notebook Sessions page provides you with an overview of your notebooks, and allows you to manage notebook sessions from your workspace or in a workspace where you have collaboration rights.

### About Notebook Sessions

The Notebook Sessions page provides you with an overview of your notebooks, and allows you to manage notebook sessions from your workspace or in a workspace where you have collaboration rights.

In the Notebook Sessions page, you can stop and unload notebook sessions. The page displays the following information about notebooks:

- **Notebook**: The name of the notebook.
- **Project**: The project in which the notebook resides.
- **Workspace**: The workspace in which the project is available.
- **Connection**: The connection name.
- **Owner**: The owner of the notebook.
- **Status**: The statuses of a notebook are:
  - **Loaded**: Indicates that the notebook is loaded but not tied to the websocket or running.
  - **Active**: Indicates that the notebook is tied to the websocket but is not running.
  - **Running**: Indicates that the notebook paragraph is queued to run or is running.

You can perform the following tasks:

- **Stop**: Select the notebook that is running, and click **Stop**. This stops the selected notebook in the server.
- **Unload**: Select the notebook that is loaded, and click **Unload**. This removes the selected notebook from memory on the server.
Administer Oracle Machine Learning

Oracle Machine Learning is managed at the system level and at the application level by an administrator.

- **Administrator** — Creates, edits, and deletes Oracle Machine Learning user accounts. The Administrator reassigns user workspace.

  Note:
  The Administrator is not authorized to run notebooks. The Administrator can only read notebooks.

- **Developer** — This is the default user role that allows you to create, and run notebooks, run SQL Statements, create SQL scripts, create jobs to schedule and run notebooks.
**Typical Workflow for Managing Oracle Machine Learning**

To manage Oracle Machine Learning and other administrative tasks, refer to the tasks listed in the table as a guide.

- **Manage OML Users**
  An administrator manages new user account and user credentials creation for Oracle Machine Learning in the User Management interface.

- **About User Data**
  In the User Data page in Oracle Machine Learning, you can view existing user data, reassign, and delete it.

- **About Compute Resource**
  The term Compute Resource refers to services such as a database, or any other backend service to which a Zeppelin Interpreter connects.

### Table: Typical Workflow for Managing Oracle Machine Learning

<table>
<thead>
<tr>
<th>Tasks</th>
<th>Oracle Machine Learning Interface</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>User account and password creation</td>
<td>Oracle Machine Learning User Management interface</td>
<td>Create Users for Oracle Machine Learning</td>
</tr>
<tr>
<td>Connection Groups — View and Reset</td>
<td>Oracle Machine Learning</td>
<td>Work with Connection Groups</td>
</tr>
<tr>
<td>Compute Resource — View</td>
<td>Oracle Machine Learning</td>
<td>About Compute Resource</td>
</tr>
<tr>
<td>User Data administration — Delete all users, all user related objects such as workspace, projects, and notebooks, and workspace reassignment</td>
<td>Oracle Machine Learning</td>
<td>About User Data</td>
</tr>
</tbody>
</table>
Note:
The tasks listed here can be performed by an administrator only.

Manage OML Users

An administrator manages new user account and user credentials creation for Oracle Machine Learning in the User Management interface.

- **Create User**
  An administrator creates a new user account and user credentials for Oracle Machine Learning in the User Management interface.

- **Add Existing Database User Account to Oracle Machine Learning**
  An administrator adds an existing database user account for Oracle Machine Learning in the User Management interface.

Create User

An administrator creates a new user account and user credentials for Oracle Machine Learning in the User Management interface.

Note:
You must have the administrator role to access the Oracle Machine Learning User Management interface.

To create a user account:
1. Select an Autonomous Data Warehouse instance and on the details page click **Service Console**.
2. On the Service Console click **Administration**.
3. Click **Manage OML Users** to open the Oracle Machine Learning User Administration page.
4. Click **Create** on the Oracle Machine Learning User Administration page.
5. In the **Username** field, enter a username for the account. Using the username, the user will log in to an Oracle Machine Learning instance.

6. Enter a name in the **First Name** field.

7. Enter a name in the **Last Name** field.

8. In the **Email Address** field, enter the email ID of the user.

9. Select the option **Generate password and email account details to user. User will be required to reset the password on first sign in.** to auto generate a temporary password and send an email with the account credentials to the user.
   
   If you select this option, you need not enter values in the **Password** and **Confirm Password** fields; the fields are grayed out.

10. In the **Password** field, enter a password for the user, if you choose to create a password for the user.

   This option is disabled if you select the **Generate password...** option to auto generate a temporary password for the user.

11. In the **Confirm Password** field, enter a password to confirm the value that you entered in the **Password** field.

   By doing so, you create the password for the user. The user can change the password when first logging in.

12. Click **Create**.

    This creates a new database user and grants the required privileges to use Oracle Machine Learning.

---

**Note:**

With a new database user, an administrator needs to issue grant commands on the database to grant table access to the new user for the tables associated with the user's Oracle Machine Learning notebooks.

---

**Add Existing Database User Account to Oracle Machine Learning**

An administrator adds an existing database user account for Oracle Machine Learning in the User Management interface.

---

**Note:**

You must have the administrator role to access the Oracle Machine Learning User Management interface.

---

To add an existing database user account:

1. Select an Autonomous Database instance, and on the details page click **Service Console**.

2. On the Service Console, click **Administration**.

3. Click **Manage OML Users** to add Oracle Machine Learning users.
4. Click **Show All Users** to display the existing database users.

![Users page](image)

**Note:**
Initially, the **Role** field shows the role **None** for existing database users. After adding a user the role **Developer** is assigned to the user.

5. Select a user. To select a user select a name in the **User Name** column. For example, select **ANALYST1**.

Selecting the user shows the Oracle Machine Learning **Edit User** page.

6. Enter a name in the **First Name** field. (Optional)

7. Enter the last name of the user in the **Last Name** field. (Optional)

8. In the **Email Address** field, enter the email ID of the user.

Making any change on this page adds the existing database user with the required privileges as a Oracle Machine Learning user.

9. Click **Save**.

This grants the required privileges to use the Oracle Machine Learning application. In Oracle Machine Learning this user can then access any tables the user has privileges to access in the database.

### About User Data

In the User Data page in Oracle Machine Learning, you can view existing user data, reassign, and delete it.

The User Data page lists details of the Oracle Machine Learning user such as the name, role, comments, last updated date. You can perform the following tasks:

- **Delete User Data:** To delete a user, select the user to delete and click **Delete User Data**.
- **Reassign:** To reassign workspace and templates from one user to another.
- **Reassign**

The **Reassign** option allows you to reassign workspaces, along with templates, from one user to another.
Reassign

The Reassign option allows you to reassign workspaces, along with templates, from one user to another.

To reassign workspaces:

1. On the User Data page, select the user from whom you want to reassign workspace and click Reassign.
   The Reassign page opens.
2. In the Target User field, select the user to whom you want to reassign workspace.
3. Select All Templates if you want to reassign all the templates associated with the user selected in the User Data page.
4. Select:
   • Reassign all workspaces: To reassign all the workspaces associated with the selected user.
   • Select workspaces to reassign: To reassign particular workspaces associated with the selected user.
5. Click Reassign.

After the templates and workspaces are reassigned successfully, a notification message is displayed on the User Data page with the number of templates and workspaces reassigned.

About Compute Resource

The term Compute Resource refers to services such as a database, or any other backend service to which a Zeppelin Interpreter connects.

Note:

You must have the Administrator role to access the Compute Resources page.

The Compute Resources page displays the list of compute resources along with the name of each resource, its type, comments, and last updated details. To view details of each Compute Resource, click the Compute Resource name. The following connection details are displayed:

• Name
• Comment
• Host
• Port