

Oracle® WebDB™

Creating and Managing Components - Field-Level Help

Release 2.2

October, 1999

Part No: A77055-01

ORACLE®

Oracle WebDB Creating and Managing Components - Field-Level Help, Release 2.2

Part No: A77055-01

Copyright © 1999, Oracle Corporation. All rights reserved.

Authors: David Mathews, Susan Léveillé, Carol Menzigian

Editor: Susan Barton

Contributors: Haranadh Abburu, Mark Clark, Thiagarajan Natarajan, Frank Rovitto, Todd Vender, Hui Zeng

The programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of such applications if the programs are used for such purposes, and Oracle Corporation disclaims liability for any damages caused by such use of the programs.

The programs (which include both the software and documentation) contain proprietary information of Oracle Corporation; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the programs is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. Oracle Corporation does not warrant that this document is error free. Except as may be expressly permitted in your license agreement for these programs, no part of these programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Oracle Corporation.

If the programs are delivered to the U.S. Government or anyone licensing or using the programs on behalf of the U.S. Government, the following notice is applicable:

Restricted Rights Notice Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the programs, including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication, and disclosure of the programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

Oracle is a registered trademark, and [Insert trademark(s). Check trademarks list available from the Writers' Resource Center] is a [are] trademark[s] or registered trademark[s] of Oracle Corporation. All other company or product names mentioned are used for identification purposes only and may be trademarks of their respective owners.

Table of Contents

SEND US YOUR COMMENTS	XIII
ABOUT THIS BOOK	XIV
BROWSING	17
Browse Database	17
Database [objects]	19
Browse Database Schemas	20
Browse Object Type	21
All Objects with Name Containing [Object name]	22
Query and Update Table	23
Query options	27
Execute (Function, Procedure)	28
Execute Procedure	32
Find an Existing Component	35
Component status codes	37
BUILD COMPONENTS	38
Finish Component	38

Calendars	39
Calendar Building	39
Calendars: Calendar Name and Schema	40
Calendars: SQL Query	41
Calendars: Display Options	43
Calendars: Parameter Entry Form Display Options	46
Calendars: Text Options	49
Calendars: Advanced PL/SQL Code	51
 Charts	 52
Chart Building	52
Charts: Chart Name and Schema	54
Charts: Tables or Views	55
Charts: Table/View Columns	56
Charts: Column Conditions	59
Charts: Display Options	61
Charts: Parameter Entry Form Display Options	64
Charts: Text Options	67
Charts: Advanced PL/SQL Code	69
 SQL-based Charts	 70
SQL-based Query Charts: Chart Name and Schema	70
SQL-based Query Charts: SQL Query	71
SQL-based Query Charts: Display Options	73
SQL-based Query Charts: Parameter Entry Form Display Options	75
Chart summary options	78
SQL-based Query Charts: Text Options	79
Chart parameter options	81
SQL-based Chart Wizard: Advanced PL/SQL Code	82
 Dynamic Pages	 83
Dynamic Page Building	83
Dynamic Pages: Dynamic Page Name	84
Dynamic Pages: Dynamic Page Content	85
Dynamic Page Wizard: Refine Page Content	86
Dynamic Page Wizard: Log Activity Option	87
 Forms	 88
Form Building	88
Forms from Tables	90
Forms (from tables): Form Name and Schema	90
Forms (from tables): Tables or Views	91

Forms (from tables): Form Layout Selection	92
Forms (from tables): Field-level Formatting and Validation	93
Forms (from procedures) Form-level Formatting and Validation	100
Forms (from tables): Button Options	103
Forms (from tables): Text Options	104
Forms (from table): Advanced PL/SQL Code	106
Forms from Stored Procedures	107
Forms (from procedure): Form Name and Schema	107
Forms (from procedure): Procedure Name	108
Forms (from procedures): Form Layout Selection	109
Forms (from procedures): Form Layout Editor - Unstructured	110
Forms (from procedures): Field-level Formatting and Validation	111
Forms (from procedures): Button Options	117
Forms (from procedures): Text Options	118
Form (from procedures): Advanced PL/SQL Code	120
Master-Detail Forms	121
Master-Detail Forms: Master-Detail Form Name and Schema	121
Master-Detail Forms: Tables or Views	122
Master-Detail Forms: Join Conditions	123
Master-Detail Forms: Master Row Formatting	124
Master-Detail Forms: Detail Row Formatting	127
Master-Detail Forms: Display Options	131
Master-Detail Forms: Button Options	134
Master-Detail Forms: Master Row Finder	135
Master-Detail Forms: Text Options	136
Master-Detail Forms: Advanced PL/SQL Code	138
Query by Example Forms	139
Query by Example Forms: QBE Name and Schema	139
Query by Example Forms: Table or View	140
Query by Example Forms: Table/View Columns	141
Query by Example Forms: QBE Results Formatting	142
Query by Example Forms: Display Options	143
Query by Example Forms: Button Options	146
Query by Example Forms: Text Options	147
Query by Example Forms: Advanced PL/SQL Code	149
Frame Drivers	150
Frame Driver Building	150
Frame Drivers: Frame Driver Name and Schema	151
Frame Drivers: SQL Query for Frame Driver	152
Frame Drivers: Parameter Entry Form Display Options	154
Frame Drivers: Initial Page in Target Frame	155

Frame Drivers: Display Options	157
Frame Drivers: Text Options	160
Frame Drivers: Advanced PL/SQL Code	162
Hierarchies	163
Hierarchy Building	163
Hierarchies: Hierarchy Name and Schema	164
Hierarchies: Tables or Views	165
Hierarchies: Table/View Columns	166
Hierarchies: Column Conditions	168
Hierarchies: Display Options	170
Hierarchies: Parameter Entry Form Display Options	173
Hierarchies: Parameter options	176
Hierarchies: Text Options	177
Hierarchies Wizard: Advanced PL/SQL Code	179
Menus	180
Menu Building	180
Menus: Menu Name and Schema	182
Menus: Menu Options - Menu	183
Menus: Menu Options - Submenu	188
Menus: Menu Options - Links	194
Reports	200
Report Building	200
Reports: Name and Schema	202
Reports: Table and Views	203
Reports: Join Conditions	204
Reports: Table/View Columns	205
Reports: Column Conditions	206
Reports: Column Formatting	208
Reports: Display Options	210
Reports: Parameter Entry Form Display Options	214
Reports: Text Options	218
Reports: Advanced PL/SQL Code	220
Advanced PL/SQL code examples	221
Example reports formatted with column breaks	222
SQL-based Reports	224
SQL-based Reports: Report Name and Schema	224
SQL-based Reports: SQL Statement	225
SQL-based Reports: Column Formatting	227

SQL-based Reports: Display Options	229
SQL-based Reports: Parameter Entry Form Display Options	233
SQL-based Reports: Text Options	237
SQL-based Reports: Advanced PL/SQL Code	239

MANAGE COMPONENTS 240

Manage Component	240
Manage Component Privileges	242
Manage Component	244
Copy Component	246
Rename Component	247
Drop Component	248

BUILD SHARED COMPONENTS 249

Manage Colors	249
Edit Color Definition	251
Manage Images	252
Edit Image Definition	254
Manage Fonts	255
Edit Font Definition	256
Manage U/I Templates	257
Create/Define Structured U/I Template	259
Create Unstructured UI Template	263
Manage Links	265
Links: Link Name and Schema	267
Links: Link Target Type and Name	268
Links: Link Target Inputs	270
Links: Finish Link	273
Manage List of Values	274
Create Dynamic List of Values	276
Edit Dynamic List of Values	279
Create Static List of Values	282
Edit Static List of Values	284

Manage JavaScripts	286
Create JavaScript	288
Guidelines for writing JavaScripts	289
Edit JavaScript	291
Test JavaScript	292
 BUILD DATABASE OBJECTS	 293
 Build Database Objects	 293
 Database object finish	 294
 Functions	 295
Create Function: Schema, Function Name, and Return Datatype	295
Create Function: Arguments	296
Create Function: Function Body	297
 Indexes	 298
Create Index: Schema and Index Name	298
Create Index: Table Columns	300
Create Index: Storage Parameters	301
 Packages	 302
Create Package: Schema and Package Name	302
Create Spec and Body	303
 Procedures	 304
Create Procedure: Schema and Procedure Name	304
Create Procedure: Arguments	305
 Sequences	 306
Create Sequence: Schema and Sequence Name	306
 Synonyms	 307
Create Synonym: Schema and Synonym Name	307
Create Synonym: Object Name	308
 Tables	 309
Create Table: Schema and Table Name	309
Create Table: Columns	310
Create Table: Storage	311

Triggers	312
Create Trigger: Schema and Trigger Name	312
Create Trigger: Trigger Table and Options	313
Create Trigger: Trigger Script	315
Views	316
Create View: Schema and View Name	316
Create View: View Body	317
 CREATE SITES	 318
Manage Sites page	318
Site Name	319
Select Owning Schema	320
Language	322
Select Tablespaces	323
Select Demos	324
Create Site	325
 ADMINISTRATION	 328
Role Manager	328
Manage Roles	330
WebDB Grant Manager	331
Grant Manager	332
Report WebDB Developer Privileges	334
Change Your Password	335
User Manager	336
User Manager: User	339

User Manager: Grants	342
User Manager: Roles	344
User Manager: Build Privileges	346
User Manager: Browse Privileges	347
Listener Settings	349
Oracle WebDB PL/SQL Gateway Settings	349
Oracle WebDB Listener Settings	353
Stored Results	356
Stored Results Manager	356
Batch Results	359
Edit Stored Results	360
Oracle Reports Security	361
Report Definition File Access	361
Report Definition File Access: Report Name and Schema	363
Report Definition File Access: Users and Roles	364
Report Definition File Access: Availability Calendar	365
Report Definition File Access: Required Parameters	366
Report Definition File Access: Optional Parameters	369
Report Definition File Access: Validation Trigger	371
Report Definition File Access: Create Report Definition File	372
Server Access	372
Server Access: Server Definition	374
Server Access: Users and Roles	376
Server Access: Availability Calendar	377
Server Access: Add Reports Server	378
Printer Access	379
Printer Access: Printer Name	380
Printer Access: Users and Roles	381
Printer Access: Availability Calendar	382
Printer Access: Add Printer	383
Availability Calendars	384
Simple Availability Calendar: Simple Availability Calendar	386
Simple Availability Calendar: Date/Time Availability	387
Simple Availability Calendar: Summary	390
Simple Availability Calendar: Create Simple Availability Calendar	390
Combined Availability Calendar	390
Combined Availability Calendar: Availability Calendars	391
Combined Availability Calendar: Exclude Availability Calendars	392
Combined Availability Calendar: Summary	393

Combined Availability Calendar: Create Combined Availability Calendar	393
Oracle Reports Parameter Form	394

MONITORING	395
-------------------	------------

Browse Activity Log	395
----------------------------	------------

Set Log Attribute	398
--------------------------	------------

INDEX	398
--------------	------------

Send Us Your Comments

Oracle WebDB Creating and Managing Components - Field-Level Help, Release 2.2

Part No: A77055-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the part number, chapter, section, and page number (if available). You can send comments to us by electronic mail to webdbdoc@us.oracle.com.

If you have any problems with the software, please contact your local Oracle World Wide Support Center.

About This Book

Oracle WebDB Creating and Managing Components - Field-Level Help, Release 2.2 (Part No: A77055-01) is a hardcopy version of the online Field-Level Help for the component building features of WebDB. Field-Level Help describes each field and button on each page of WebDB.

To display online Field-Level Help in WebDB, click the small help button  located on the upper right of every WebDB page.

The contents of this book are also available in printable Adobe Acrobat PDF format on your WebDB product CD at:

`\doc\us\compref.pdf`

The following table lists the other information units in the WebDB documentation library:

Information Unit	Description	Location in \doc\us directory on WebDB product CD
Release Notes (Part No: A77054-01)	Describes last minute changes to the product or documentation. Read the Release Notes before installing or using Oracle WebDB.	relnotes.htm
Quick Tour	Provides a high-level introduction to Oracle WebDB. Accessible by clicking "Quick Tour" in the WebDB or Site Builder task help Table of Contents.	\QuickTour\wbstart.htm

Information Unit	Description	Location in \doc\us directory on WebDB product CD
Oracle WebDB Tutorial Guide (Part No: A77075-01)	Provides step-by-step lessons that teaches you the basics of WebDB.	PDF version: \Tutorial\tutorial.pdf HTML version: \Tutorial\html\toc.htm
Oracle WebDB Installation Guide (Part No: A77053-01)	Provides complete Oracle WebDB 2.2 installation instructions for both Windows NT or Solaris environments. Also, provides upgrade and site migration details, WebDB Listener installation and configuration, <i>interMedia</i> Text set up information, and WebDB CGI executable and WebDB cartridge to Oracle Application Server setup information.	PDF version: \Installation\installation.pdf HTML version: \Installation\html\toc.htm
Oracle WebDB Creating and Managing Components - Task Help (Part No: A77057-01)	Hardcopy-formatted version of the online Task Help for the component building features of WebDB.	comptask.pdf
Oracle WebDB Creating and Managing Sites - Task Help (Part No: A77056-01)	Hardcopy-formatted version of the online Task Help for the Web Site building features of WebDB.	sitetask.pdf
Oracle WebDB Creating and Managing Sites - Field-Level Help (Part No: A77058-01)	Hardcopy-formatted version of the online Field-Level Help for the Web Site building features of WebDB.	siteref.pdf

Browsing

Browse Database

Description Use this page to search for objects stored in the database.

This page contains:

Schema

Type the name of the schema that owns the database object you want to find, for example, SCOTT.

If you leave this field blank, WebDB searches all schemas you have privileges to browse.

If you don't know the name of the schema, click  to the right of the text box and type search criteria in the search dialog box. You can use the % wildcard in your search criteria. For example, typing SC0% might locate the schemas named SCOTT and SCOUT.

Type

Choose the database object type you want to find. For example, choose **Tables** to search for all tables in the database. Choose **All Objects** to search for all database objects.

Object Name

Type the name of the object you want to find.

You can use the % wildcard in your search statement; for example, typing %EMP% might locate a table named Employee and a function named Calc_Employee_Salary

Browse Database

Browse

Click to search the database for objects based on the search criteria you entered in the **Schema**, **Type**, and **Object Name** fields, or any combination of these fields.

Notes

- You have the option of specifying search criteria in one, two, or all three fields on this page. For example, to search for all tables in the database, choose **Tables** in the **Object Type** drop-down list, and leave the other fields on this page blank.
- WebDB returns a maximum of 1,000 objects in a search for any object type. If a search returns more than 1,000 objects, you need to narrow your browse criteria.

You can narrow your search by specifying values in the other fields. For example, you can search for all tables in the SCOTT schema by specifying **Tables & Views** as the **Object Type** and **Scott** as the **Schema**.

Database [objects]

Description The Database [objects] page displays all objects matching the object type (for example, tables, views, triggers, procedures, etc.) and schema you specified. The text at the top of the page indicates the schema you are currently searching. The page title indicates the object type you are searching.

Click an object to perform an action on it such as:

- query and run tables and views
- run packages, procedures, and functions
- view additional information about other object types.

This page contains:

Show: All Objects	Click to display all objects owned by the schema.
Show: Tables & Views	Click to display tables and views owned by the schema.
Show: Packages, Procedures & Functions	Click to display packages, procedures and functions owned by the schema.

Notes

- The  icon indicates objects created within the last 7 days.
- An object becomes invalid if based on a database table that has been dropped. Invalid objects often automatically become valid when referenced. For example, clicking some invalid objects in a browser can automatically make them valid.

Browse Database Schemas

- The contents of this page depend on the object type you are browsing. It can contain synonyms, views, procedures - any object type you can browse.

Browse Database Schemas

Description The Browse Database Schemas page displays all schemas you have privileges to browse. Click a schema to display the object types owned by it.

Note

- If a schema you want to browse is not shown on this page, contact your DBA (a user with the DBA role) to obtain permission to browse it.

Browse Object Type

Description The Browse Object Type page displays the object types owned by the schema you specified (shown at the top of the page). Click an object type to display all objects of the type owned by the schema.

This page contains:

Show: All Objects	Click to display all objects owned by the schema.
Show: Tables & Views	Click to display tables and views owned by the schema.
Show: Packages, Procedures & Functions	Click to display packages, procedures and functions owned by the schema.

Notes

- The  icon indicates objects created within the last 7 days.
- An object becomes invalid if based on a database table that has been dropped. Invalid objects often automatically become valid when referenced. For example, clicking on some invalid objects in a browser can automatically make them valid.
- When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

All Objects with Name Containing [Object name]

Description The All Objects with Name Containing [Object name] page displays all objects matching the object name and schema you specified. The upper left corner of each section on the page displays each schema.

Click an object to perform an action on it such as:

- query and run tables and views.
- run packages, procedures, and functions.
- view additional information about other object types.

Notes

- The **NEW** icon indicates objects created within the last 7 days.
- An object becomes invalid if based on a database table that has been dropped. Invalid objects often automatically become valid when referenced. For example, clicking on some invalid objects in a browser can automatically make them valid.

Query and Update Table

Description Use the Query and Update Table page to query, update, and insert rows in the table or view identified in the upper left corner of the page. The page contains fields that correspond to each column in the table or view.

This page contains:

Show Object Information	Click to display a report of information about the table or view. This includes information such as the schema that owns it, and the names of any indexes, triggers, or constraints based on the table or view.
Query	Click to display data based on the criteria you specified on this page.
Query and Update	Click to display a report containing data based on the criteria you specified on this page. The report enables you to update the data you selected.
Insert New Row	Click to insert a new row into the table using the values you specified on this page.
Reset	Click to clear or reset all fields on this page to their default values.
Column display check box	Check to display the table column in your query results.
Column fields	Type values that will be used to query, insert a new row into, or update a row in the table identified in the upper left corner of the page. The names of the column fields are derived from the column names in the table or view.

Query and Update Table

You can use wildcards and conditions such as <, >, or IN in your search criteria.

Column data type icon

Displays the datatype for values in the column. Click an icon (for example, ) to view column information or search for distinct column values.

Column Alignment

Choose whether to display data to the left, right, or center of the column in query results.

Format Mask

Choose an Oracle display format for columns containing date and number data types. For example, if you choose **DD/MM/RR**, the date January 2nd, 1999 displays as 02/01/99.

If you choose **999G999G999G999D99999999**, the display format for numbers is 12 digits to the left of the decimal point and 8 to the right.

Note Refer to the Oracle documentation for additional information about date formatting options.

Where Clause

Type or paste a SQL WHERE clause to further restrict the data returned by a query.

Order by

Choose column values that will be used to order table rows returned by a query. This choice is equivalent to specifying a SQL ORDER BY clause.

Choose **Ascending** to sort query results according to the alphabetic (A-Z) or numeric (starting with the lowest number) order of column values, depending on the datatype for the column. Choose **Descending** to sort query results in the reverse order.

Sum Columns

Choose columns whose values you want to sum and display in the query results.

Note This list displays only columns containing numeric data.

Break on Columns

Choose up to three columns to divide the query results into groups based on the values of those columns. If you specify one or more columns in **Sum Columns**, the sum of the values is calculated for each group.

Output Format

Choose a display format for query results:

- HTML Format

Formats the results using HTML tables and displays output on a new page in the web browser. Components that contain large amounts of data may take longer to display in this format.

- Microsoft Excel

Downloads the results for display in Microsoft Excel.

- ASCII text

Formats the results using the HTML PRE tag to display headings and values in the report as ASCII text. This option is useful for displaying large amounts of data.

Maximum Rows

Type the maximum number of rows you want to display in the report.

Note If the query returns more rows than the value you specify in **Maximum Rows**, you can choose **Show Paging Buttons** in the **Query Options** list box to add a button on the results page that enables you to view more rows.

Query and Update Table

Show Null as

Type text that you want to display in the place of null values in your query results. For example, you can type a dash "-", "(null)", or leave the text box blank.

Query Options

Choose one or more options for formatting your query output.

Note Windows users can choose more than one option by clicking it while pressing the Ctrl key.

Notes

- When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

Query options

Description Query options allow you to specify how the results of your query of the table or view display.

Options

Show SQL	Displays at the top of the report the SQL query that was used to create it.
Display Results in Table with Borders	Adds a border around the table containing your query results.
Show Total Row Count	Displays at the bottom of your query results the total number of rows returned by the query.
Count Rows Only	Displays only the total number of rows returned by your query. The query results do not display.
Show Paging Buttons	Displays buttons labeled Next and Previous that enable you to view additional rows if more are returned by the query than specified in Maximum Rows .
Show HR Between Rows	Displays a line between rows in the query results.
Replace ASCII New Lines with HTML Breaks	If you display the query results as HTML-formatted text, WebDB substitutes the HTML tag, <code>
</code> , for every ASCII new line. This produces HTML-formatted results with the same line breaks as those formatted as ASCII text.

Execute (Function, Procedure)

Description Use this page to specify arguments that will be passed to a procedure or function when you execute it.

The procedure you are executing could display a WebDB component (for example, SCOTT.EXAMPLE_FORM.SHOW) or perform some other task (for example: SCOTT.GIVE_RAISE). If the procedure, displays a WebDB component, you will see three arguments defined for the procedure:

- P_REQUEST
- P_ARG_NAMES
- P_ARG_VALUES

See the notes below for an explanation of these arguments.

If the procedure performs a task other than displaying a WebDB component, the **Argument** text boxes on this page contain any arguments defined for the procedure. You can click the data type icon next to an argument to obtain additional information about it.

This page contains:

Show Object Information	Click to display a report of information about the procedure or function. This information includes the schema that owns the function, the creation date, and status.
Execute	Click to execute the procedure or function with the arguments you specified on this page.
Reset	Click to clear all fields on this page.
Argument text boxes	Type any argument you want to pass to the procedure or function.

Argument data type icon

Displays the argument's datatype. Click the icon to view more information about the argument.

Notes

- Use a colon (:) to pass multiple elements to an array argument.
- Three arguments are typically passed to a procedure that displays a WebDB component: P_REQUEST, P_ARG_NAMES, P_ARG_VALUES. You can pass one or more argument names in P_ARG_NAMES and one or more argument values in P_ARG_VALUES.

For example, when executing the SCOTT.EXAMPLE_FORM.SHOW procedure, which displays a form based on the SCOTT.EMP table, you could type `NAME` in P_ARG_NAMES, and `CLARK` in P_ARG_VALUES. `NAME` is an argument that corresponding to an entry field on the form. The form that displays would contain `CLARK` as a default value in the `NAME` entry field.

Or you could specify `DEPTNO:JOB` in P_ARG_NAMES, and `10:MANAGER` in P_ARG_VALUES to display the following form. Note that `manager` and `Accounting (10)` are selected in this form by default.

?

Example Form on EMP Table

Company Name/Logo

Query

Insert

Update

Delete

Reset

Empno

Name

Job

manager

Mgr

Hiredate

Salary

Comm

Deptno

☐ %

☒ ACCOUNTING (10)

☐ OPERATIONS (40)

☐ RESEARCH (20)

☐ SALES (30)

- If you are unsure which arguments are accepted by a procedure that displays a WebDB component:
1. Click  at the bottom of any WebDB page to display the Find an Existing Component page.
 2. In **Schema**, choose the schema that owns the procedure. In **Name Contains**, type the WebDB component created by the procedure. For example, choose the SCOTT schema and type EXAMPLE_FORM in **Names Contains** if you are searching for arguments accepted by the SCOTT.EXAMPLE_FORM.SHOW procedure.
 3. Click **Find**.

4. Click the WebDB **Component Name** that matches your query at the bottom of the page.
5. On the Manage Component page, click **Manage**.
6. Click **Show Call Interface**. A table displays P_ARG_NAMES and default P_ARG_VALUES defined for the procedure.

Execute Procedure

Description Use the Execute Procedure page to pass arguments to a procedure that displays a WebDB component; for example, SCOTT.EXAMPLE_FORM.SHOW.

Three arguments are typically passed to a procedure that displays a WebDB component:

- P_REQUEST
- P_ARG_NAMES
- P_ARG_VALUES.

You can pass one or more argument names in P_ARG_NAMES and one or more argument values in P_ARG_VALUES.

For example, when executing the SCOTT.EXAMPLE_FORM.SHOW procedure, which displays a form based on the SCOTT.EMP table, you could type `NAME` in P_ARG_NAMES, and `CLARK` in P_ARG_VALUES. `NAME` is an argument that corresponding to an entry field on the form. The form that displays would contain `CLARK` as a default value in the `NAME` entry field.

Or you could specify `DEPTNO:JOB` in P_ARG_NAMES, and `10:MANAGER` in P_ARG_VALUES to display the following form. You use a colon (:) in this example to pass multiple elements to an array argument. Note that `manager` and `Accounting (10)` are selected in this form by default.

Example Form on EMP Table
?

Company Name/Logo

Query
Insert
Update
Delete
Reset

Empno

Name

Job

Mgr

Hiredate

Salary

Comm

Deptno

☐ %

☒ ACCOUNTING (10)

☐ OPERATIONS (40)

☐ RESEARCH (20)

☐ SALES (30)

If you are unsure which arguments are accepted by a procedure that displays a WebDB component:

1. Click  at the bottom of any WebDB page to display the Find an Existing Component page.
2. In **Schema**, choose the schema that owns the procedure. In **Name Contains**, type the WebDB component created by the procedure. For example, choose the SCOTT schema and type EXAMPLE_FORM in **Names Contains** if you are searching for arguments accepted by the SCOTT.EXAMPLE_FORM.SHOW procedure.
3. Click **Find**.

Execute Procedure

4. Click the WebDB **Component Name** that matches your query at the bottom of the page.
5. On the Manage Component page, click **Manage**.
6. Click **Show Call Interface**. A table displays P_ARG_NAMES and default P_ARG_VALUES defined for the procedure.

This page contains:

Execute	Click to execute the procedure with the arguments you specified on this page.
Reset	Click to clear all text boxes on this page.
Argument text boxes	Type any argument you want to pass to the procedure.
Argument data type	Displays the argument's data type. Click the data type icon to view more information about the argument.

Note

- Use a colon (:) to pass multiple elements to an array argument.

Find an Existing Component

Description Use the Find an Existing Component page to search for WebDB components and shared components. The search returns the following information:

- the component's type; for example, Chart, Report, or Hierarchy.
- the name of the schema that owns it.
- the name of the component.
- when it was created or last edited.
- the name of the user who created or last edited the component.
- its current status (for example, Production or Archive).

After you locate the component, you can click its name to manage, run, or edit it.

This page contains:

Schema	Choose the schema that owns the component or shared component you want to find. The Schema drop-down list displays all schemas you have privileges to browse.
Name Contains	Type one or more characters in the name of the component you want to find. For example, you could type <code>sales_d</code> or <code>depart</code> to search for a chart named <code>SALES_DEPARTMENT</code> . Note The Name Contains text box is not case sensitive.
U/I Components	Check next to each component type you are trying to find. You can check more than one component type. Note Check SQL Workshop to search for database objects.

Find an Existing Component

Shared Components

Check next to each shared component type you are trying to find. You can check more than one component type.

Note You can find WebDB components and shared components in a single search by checking the **U/I Components** and **Shared Components** check boxes.

Sort By

Choose a method for sorting the columns in the report returned by the search. You can order the report by:

Name - the component's name.

Created by - the name of the user who created or last edited the component.

Date Created - the date the component was created or last edited (columns display from oldest to most recent date).

Date Created Desc - the date the component was created or last edited (columns display from most recent to oldest date).

Schema - the name of the schema that owns the component.

Status, Schema, Name - the component's status; for example, PRODUCTION or ARCHIVE. After sorting by status, columns are next sorted according to the schema that owns the component, then the component's name.

Type, name - the component's type; for example, Hierarchy, Chart, or Report. After sorting by status, columns are next sorted according to the component's name.

Status

Choose the status code of the component you want to find.

Find

Click to search for components or shared components based on your selections on this page.

Component status codes

WebDB uses the following codes to indicate the status of each version of a component:

Code	Meaning
ARCHIVE	Indicates an old version of the component. When you edit a component, the version you save after making changes becomes a PRODUCTION version. The version you opened to edit is automatically saved as an ARCHIVE version. Higher version numbers indicate more recent ARCHIVE versions.
CREATE	Indicates that the component is currently being created in the build wizard by another user.
EDIT	Indicates that the most current version of the component is now being edited by another WebDB user. The component is locked until the other user finishes editing it.
PRODUCTION with INVALID PACKAGE	Indicates the most recent version of the component. The component will not run because it contains errors. You must edit the component to fix the errors before it will run.
PRODUCTION with VALID PACKAGE	Indicates the most recent version of the component. The component runs without errors.

Build Components

Finish Component

Description Use the Finish Component page to create a new WebDB component based on your selections in the previous pages of this build wizard. If you are satisfied with the choices you made in the wizard, click **OK** to create a packaged procedure that, when executed, displays the component.

If you aren't satisfied with your choices, use the  button to navigate back to any wizard page where you want to make changes. After you make changes, click  to return to the Finish Component page. Then, click **OK** to create the component's package. WebDB does not create the component package until you click **OK** on this page.

Always use the  and  buttons to navigate to and from previous wizard pages. Don't use your web browser's Back and Forward buttons. After you click **OK** on the Finish Component page, you can still edit options in the component's Edit dialog box, but not in the build wizard itself. At this point, you will get an error if you use your web browser's Back button to return to previous wizard pages.

After you click **OK** on the Finish Component page, the Manage Component page displays and indicates whether you created a valid or invalid package. A valid package executes without errors. An invalid package contains error that prevent it from executing. To fix the errors, you must edit the finished component using the Edit dialog box for the component.

You can use the Edit dialog box to change most options you selected when building the original component. In general, you can change any build option except the table or procedure on which the component was based.

Calendars

Calendar Building

Description Use the Calendar Building page to create a new calendar, or find an existing or recently edited calendar. After you find a calendar, you can edit it using the Edit Calendar dialog box.

This page contains:

Create a New Calendar

Create

Click to create a new calendar. You must know how to write a SQL query to build a calendar using this wizard.

Find an Existing Calendar

Find in Schema

Choose the schema that owns the calendar you want to find. The **Find in Schema** list displays all schemas you have privileges to browse.

Find

Click to search for calendars in the schema you specified in the **Find in Schema** drop-down list.

Select a Recently Edited Calendar

Name

Displays the names of the five most recently created or edited calendars. Click a **Name** to edit the calendar.

Schema

Displays the schema that owns the calendar.

Type

Displays the component type, i.e. **Calendar**.

Changed

Displays in days, hours, minutes, or seconds how long ago the calendar was created or last edited.

By

Displays the name of the developer who created or last edited the calendar.

Calendars: Calendar Name and Schema

Description Use this page to choose a name for the calendar and the name of the database schema in which the finished calendar will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished calendar. Only schemas that you are allowed to build in are listed in the drop-down list.
Calendar Name	Type the name you want to use to identify the database package containing the finished calendar; for example, MY_CALENDAR.

Notes

- The **Schema** becomes part of the URL that end users can specify to display the component.
- Follow these guidelines when typing a **Calendar Name**:
 - You must specify a calendar name (null is not allowed).
 - The name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a calendar MY_CALENDAR, but not MY CALENDAR.
 - You cannot name a calendar with a PL/SQL reserved word; for example, COLUMN, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.
- Do not insert a colon in the bind variable itself. For example, the report fails if your bind variable is:

```
select * from WEBDB.test_time
where time_stanp = :timevar
```


Calendars: SQL Query

Description Use this page to type or paste the SQL statement that selects the table or view data to display in the chart. You can specify up to five columns from a table or view to include the SELECT statement for the calendar:

Column	Specifies	Function
1 (always appears first in the SQL Query)	the_date (required)	Displays text on the calendar on the dates contained in this table or view column. Values in this column must have the DATE datatype.
2 (always appears second in the SQL Query)	the_name (required)	Displays cell text from this table or view column according to the dates in column 1.
3	the_name_link (optional)	Specifies a link from the values in column 2 to another WebDB component or URL. If you don't want to display links from calendar names, specify <code>null</code> for this column.
4	the_date_link (optional)	Specifies a link from the values in column 1 to another WebDB component or URL. If you don't want to display links from calendar dates, specify <code>null</code> for this column.
5	the_target_frame (optional)	Specifies the URL of a frame in a web page. Specify this column if you want to link to a specific frame in a URL. If you don't want to link to a target frame, specify <code>null</code> for this column.

Calendars

For example,

```
select hiredate, ename, null, null, null from  
scott.emp
```

You do not need to specify all five columns. Instead of the above query, for example, you can specify:

```
select hiredate, ename from scott.emp
```

This page contains:

Enter the SQL SELECT
Statement

Type or paste a SQL SELECT statement.

Notes

Use alphanumeric strings preceded by colons as bind variables (for example, `:var1`, `:var2`, `:var3...`). All bind variables must **begin** with an alpha character (a, b, c, etc.) For each bind variable you specify, WebDB displays a parameter entry field in the chart's parameter entry form. The entry field prompts end users to choose conditions for displaying data in the chart.

Do not add single or double quotes around bind variables. For example, do not specify `:var1'` or `" :var1"` as a bind variable.

- Do not insert a colon in the bind variable itself. For example, the report fails if your bind variable is:

```
select * from WEBDB.test_time  
where time_stanp = :timevar
```

- If you specify a bind variable in this page, you can associate a List of Values with it in the Calendars: Parameter Entry Fields page.
- Preface table names with their owning schema (for example, SCOTT.EMP for a table owned by SCOTT) if your statement includes tables owned by a schema other than the one that will own the finished chart.
- You can include relative HTML links by coding them into the SELECT list.
- Use aliases for long columns names (greater than 32 characters).

Calendars: Display Options

Description Use this page to choose options that control the appearance of the finished calendar.

This page contains:

Run Options

Maximum Months	Type the maximum number of months you want to display in the calendar.
Show Monday-Friday Only	Check to display only Mondays through Fridays on the calendar. If you check this option, Saturdays and Sundays, as well as any table or view data that falls on those days, do not display.
Page Width (%)	Type the size of the calendar relative to the web page on which it appears. Choosing 100% displays a full-size calendar. Smaller percentages display smaller sized calendars.
Show Query Conditions	Check to display at the bottom of the calendar all user-specified parameters passed to the query that created the calendar, and the time when the calendar was created.
Paginate	Check to display on the calendar a button labeled Next. Clicking the button allows the end user to see additional calendar months The maximum number of calendar months that the end user can see is set by the Maximum Rows option.
Log Activity	Check to log in the WebDB activity log the names of end users who request the calendar as well as other performance information.
Show Timing	Check to display at the bottom of the calendar the time from when the server received the request to generate the calendar to when the HTML for the calendar was generated.

Calendars

Default Output Format

Choose a display format for the calendar:

- **HTML Format**
Formats the calendar using HTML tables and displays output on a new page in the web browser. Components that contain large amounts of data may take longer to display in this format.
- **Excel**
Downloads the calendar for display in Microsoft Excel.
- **ASCII text**
Formats the calendar using the HTML PRE tag to display heading and values in the calendar as ASCII text. This option is useful for displaying large amounts of data.

Look and Feel Options

Month Type Face

Choose a font for displaying months and years on the calendar.

Month Font Size

Choose the size of the text that displays months and years on the calendar.

Month Font Color

Choose the color for displaying months and years on the calendar.

Day Type Face

Choose a font for displaying the days (**Monday**, **Tuesday**, **Wednesday**, etc.) and the dates (**01**, **02**, **03**, etc.) on the calendar.

Day Font Size

Choose the size of the text that displays the names of days (**Monday**, **Tuesday**, **Wednesday**, etc.) and the dates (**01**, **02**, **03**, etc.) on the calendar.

Day Font Color

Choose the color for displaying the days (**Monday**, **Tuesday**, **Wednesday**, etc.) and the dates (**01**, **02**, **03**, etc.) on the calendar.

Cell Type Face

Choose a font for displaying text that appears in calendar cells.

Cell Font Size	Choose the size of the text that appears in calendar cells. Specify Font Size as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.
Cell Font Color	Choose the color for displaying text that appears in calendar cells.
Heading Background Color	Choose a background color for the calendar heading (the areas on the calendar in which the days of the months appear).
Table Background Color	Choose a color for all cells that appear in the calendar.

Calendars: Parameter Entry Form Display Options

Description Use this page to optionally display parameter entry fields in the calendar's parameter entry form. WebDB displays a parameter entry field for each table or view column for which you specified a bind variable on the SQL-based Calendars: SQL Statement page. The entry field enables end users to choose conditions for displaying data in the calendar.

For example, if you specified a bind variable for the DEPTNO column of the SCOTT.EMP table, the calendar's parameter form displays an entry field for the column. End users can type a department number in the field to display only data about employees from that department on the calendar.

You can optionally add a List of Values to the entry field. In the previous example, instead of requiring end users to type a numeric value, you could add a List of Values that enables them to choose 10, 20, or 30.

Other options on this page let you choose which buttons and options are displayed to the end user of the parameter entry form. For example, you can choose whether to display a Batch button that allows the end user to run the calendar in batch mode, or an option that allows the end user to choose how many months to display on the calendar.

This page contains:

Bind Variable	Displays the bind variables that you specified on the Calendars: SQL Query page. For each bind variable you specified, WebDB displays a parameter entry field in the calendar's parameter entry form. The entry form enables end users to choose conditions for displaying data in the calendar.
Prompt	Type the prompt text you want to display next to the entry field. The prompt text tells end users what to enter in the field; for example: Select the range of dates you want to display.

Value	Type a default value for the column's entry field. End users have the option of accepting this value or specifying their own
LOV	Type the name of the List of Values you want to use for the parameter entry field. End users of the parameter entry form can choose values from this list to limit data displayed in the calendar. For example, you could add a List of Values containing months of the year. Click  to the right of the text box to search for Lists of Values.
Display LOV As	Choose a format for displaying the List of Values to end users.
Monday to Friday Only	Check to enable the end user to display only Mondays through Fridays on the calendar. If the end user chooses this option, Saturdays and Sundays, as well as any table or view data that fall on those days, do not display.
Cell Font Size	Check to enable the end user to choose the size of the text that appears in calendar cells.
Maximum Months/Page	Check to enable the end user to choose how many calendar months to display on a web page.
Run	Check to display a Run button on the calendar's parameter entry form. End users can click the Run button to display the calendar with the options they specified in the parameter entry form.
Save	Check to display a Save button on the calendar's parameter entry form. End users can click the Save button to save their option selections.
Batch	Check to display a Batch button on the calendar's parameter entry form. End users can click the Batch button to run the report in batch mode and save the results in the database.

Calendars

Reset	Check to display a Reset button on the calendar's parameter entry form. End users can click the Reset button to reset all entry fields to their default values.
Name	Type the label you want to display on the button; for example, <code>Display Calendar</code> . Tip Keep the Name short to avoid displaying large buttons.
Location	Choose the vertical location of the button on the parameter entry form. Note Choose Don't Show if you don't want the button added to the parameter entry form.
Alignment	Choose whether to display the button on the left, center, or right margin of the parameter entry form.

Calendars: Text Options

Description Use this page to choose text that you want to display at the top or bottom of the calendar page or its parameter entry form. its calendar or parameter entry form. The end user clicks a ? button on page to view the text.

This page contains:

Template	Choose a template to set the look and feel of calendar and parameter entry form elements such as background colors and images, and the image that appears in the upper left corner of the calendar page.
Preview Template	Click to view the appearance of the template currently selected in the Template list.
Title	Type text you want to display in the banner at the top of the calendar or parameter entry form.
Header Text	Type any introductory text that you want to display at the top of the calendar or parameter entry form, just below the title.
Footer Text	Type any text that you want to display at the bottom of the calendar or parameter entry form.
Help Text	Type any text that you want to display in a help page for the calendar or parameter entry form. If you type text in this text box, WebDB automatically adds a help button (?) to the calendar or parameter entry form. End users can click this button to link to a page displaying the help text.

Calendars

Notes

- You can choose a template in addition to any options you choose in the Calendars: Display Options page. Templates control the look and feel of the page on which the calendar appears, whereas display options control the look and feel of the calendar itself.
- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** entry fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

Calendars: Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different points during the execution of the HTML code that creates the calendar or its parameter entry form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Enter the PL/SQL code to execute before displaying the header

Enter the PL/SQL code to execute after displaying the footer

Enter the PL/SQL code to run after displaying the page

Type or paste a PL/SQL procedure that will execute before the page containing the calendar or parameter entry form displays.

Type or paste a PL/SQL procedure that will execute before the calendar or parameter entry form header displays.

Type or paste a PL/SQL procedure that will execute after the calendar or parameter entry form footer displays.

Type or paste a PL/SQL procedure that will run after the page containing the calendar or parameter entry form displays.

Charts

Chart Building

Description Use the Chart Building page to create a new chart, or find an existing or recently edited chart. After you find a chart, you can edit it using the Edit Chart dialog box.

This page contains:

Create a New Chart

Chart from Query Wizard	Click to create a new chart using the Chart Build Wizard. The build wizard guides you through all pages for creating a chart, including creating the SQL query that selects the data displayed in the chart.
Chart from SQL Query	Click to create a chart by writing your own SQL query that selects the data displayed in the chart.
Create	Click to create a new chart using either the SQL query build wizard or by writing your own SQL query.

Find an Existing Chart

Find in Schema	Choose the schema that owns the chart you want to find. The Find in Schema list displays all schemas you have privileges to browse.
Find	Click to find all charts owned by the schema you specified.

Select a Recently Edited Chart

Name	Displays the name of the five most recently created or edited charts. Click a Name to edit the chart.
------	--

Schema	Displays the schema that owns the chart.
Type	Displays the chart type, either: • A chart created using the SQL query build wizard. • A chart created using a hand-coded SQL query.
Changed	Displays in days, hours, minutes, and seconds how long ago the chart was created or last edited.
By	Displays the developer who created or last edited the chart.

Charts: Chart Name and Schema

Description Use this page to choose a name for the chart and the name of the database schema in which the finished chart will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished chart. Only schemas that you are allowed to build in are listed in the drop-down list.
Chart Name	Type the name you want to use to identify the database package containing the finished chart; for example, <code>MY_CHART</code> .

Notes

- The **Schema** becomes part of the URL that end users specify to display the component.
- Follow these guidelines when typing a **Chart Name**:
 - You must specify a name (null is not allowed).
 - The chart name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a chart `MY_CHART`, but not `MY CHART`.
 - You cannot name a chart with a PL/SQL reserved word; for example, `COLUMNS`, `PACKAGE`, `VARCHAR`. Refer to the Oracle documentation for more information about reserved words.

Charts: Tables or Views

Description Use this page to choose the database table or view on which you want to base the chart.

This page contains:

Tables/Views

Choose the table or view containing the data that will be used to create the chart. The drop-down list contains all tables and views on which you have SELECT, INSERT, UPDATE, or DELETE privileges.

The table name in the drop-down list is prefixed by the schema that owns it. For example, a table named EMP in the SCOTT schema appears in the list as SCOTT.EMP.

Notes

- You can choose only one table or view using this wizard. If you want to base the chart on more than one table or view, you must write your own SQL to create the chart. To do so, navigate to the Chart Building page, and click the **Chart from SQL Query** option. A wizard will guide you through the pages for creating your own SQL query.
- You automatically have SELECT, INSERT, UPDATE, or DELETE privileges on a table or view if it is owned by a schema in which you have Build In privileges.

Charts: Table/View Columns

Description Use this page to choose the table or view columns whose data you want to display as Labels, Links, and Values in the chart.

This page contains:

Label	Choose the table or view column containing values that you want to use as labels for the bars on the chart. Labels are displayed along the bottom of a vertical chart and along to the left edge of a horizontal bar chart. They identify the bars on the chart. To create a chart that displays the salaries of each employee from the SCOTT.EMP table, for example, you could choose ENAME as the label column, and SAL as a value column.
Link	Choose a link from the chart's labels or values to another WebDB component or URL. Note You can specify a link only if one has been created for the Label or Value column and stored in the database. If a link does not exist, you can create the chart without the link, create a link, then edit the chart to include the link you created.
Value	Choose the table or view column containing values that will be used to calculate the relative size of the bars in the chart. Value columns always contain numeric data. To create a chart that displays the salaries of each employee from the SCOTT.EMP table, for example, you could choose ENAME as the label column, and SAL as a value column.

Tips

- The default for **Value** is 1. Specifying a **Value** of 1 is useful if you also choose a group function. For example, you can choose the JOB column from SCOTT.EMP as the **Label**, 1 as the **Value**, and COUNT as the **Group Function**. This creates a chart that displays the number of employees in each job classification.
- You can choose more than one value column. For example, you can specify SAL+COMM to display in the chart employee salaries added to their commissions.

Group Function

A group function calculates a single summary value (SUM, MINIMUM, MAXIMUM, AVERAGE, or COUNT) from groups of numeric values in the **Values** column. WebDB uses values in the **Label** column to determine these groupings.

For example, you can choose the JOB column from SCOTT.EMP as the **Label**, the SALARY column as the **Value**, and AVG as the **Group Function**. This creates a chart that displays the average salary for each job classification.

Order by

Choose a method for ordering the chart's data:

- Order by Label - the bars in the chart are shown in the same order as values in the table column that you specified in the Label entry field.
- Order by Label Desc - the bars in the chart are shown in the reverse order as values in the table column
- Order by Value - the bars in the chart are shown in the same numeric order as values in the table column you specified in the Value entry field.
- Order by Value Desc - uses the same criteria to order the bars as Order by **Value**, but displays them in reverse order.

Charts

Treat Null Values As

Type the value you want to use for all null table or view Value column values.

Include Null Values

Check to display null values in the chart.

Charts: Column Conditions

Description Use this page to optionally specify conditions that limit the data displayed in the chart. To specify a condition, choose a **Column Name**, a **Condition**, and a **Value**. For example, to display in the chart data for all employees in Department 10, choose Department from **Column Name**, like from **Condition**, and 10 from **Value**.

This page contains:

Conditions

Column Name

Choose columns whose values will be used to limit the data displayed in the chart.

For example, if you want to display in the chart values greater than 3000 from the EMPNO column of the SCOTT.EMP table, choose EMPNO as the **Column Name**.

Note After you choose all columns you want to include, make sure % is specified for any unused **Column Name** rows on this page.

Condition

Choose a condition that selects which column values will be used to display chart data.

For example, if you want to display in the chart values greater than 3000 from the EMPNO column of the SCOTT.EMP table, choose > as the **Condition**.

Value

Type a column value that will be used to limit the data displayed in the chart.

For example, if you want to display in the chart values greater than 3000 from the EMPNO column of the SCOTT.EMP table, type 3000 as the **Value**.

More Conditions

Click to display more fields on this page that allow you to specify additional conditions for limiting data displayed on the chart.

Charts

Note

- To specify multiple values after an IN or NOT IN condition, type a colon between each value. For example, to display in the chart 3 departments from the SCOTT.EMP table, you could choose EMP.DEPTNO as the **Column Name**, IN as the **Condition**, and 10:20:30 as the **Value**.

Charts: Display Options

Description Use this page to choose options that control the appearance of the chart.

This page contains:

Run Options

Maximum Rows	Type the maximum number of bars you want to display in the chart.
Show Query Conditions	Check to display at the bottom of the chart all user-specified parameters passed to the query that created the chart, and the time when the chart was created.
Paginate	Check to display on the chart a button labeled Next . An end user clicks the button to see more bars on the chart. The maximum number of chart bars that the end user can see is set by the Maximum Rows option.
Log Activity	Check to log in the WebDB activity log the names of end users who request the chart as well as other performance information.
Show Timing	Check to display at the bottom of the chart the time from when the server received the request to generate the chart to when the HTML for the chart was generated.

Summary Options

Select one or more options that display summary information about the chart. Each option you select is included in the summary information box at the bottom of the chart.

Note Windows users can choose more than one option by clicking it while pressing the Ctrl key.

Look and Feel Options

Type Face	Choose a font for displaying chart text. Specify Font Size as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.
Font Size	Choose the size of the font for displaying chart text.
Font Color	Choose the color for displaying chart text.
Type	Choose whether to display the chart bars in a horizontal or vertical orientation.
Axis	Choose a method for displaying chart bars relative to the value of chart's axis. For example, if you choose Zero, the value of the axis is set at 0. If you choose Average Value, the axis is set at the average of all values in the Value column of the table on which the chart is based. Bars with values higher than average display above the axis; those with lower values display below the axis.
Bar Image	Choose an image that will be used to fill in the bars on the chart. Tip Choose MULTI to display each bar in a different color.
Chart Scale	Choose a % value to set the size of chart bars relative to the web page. Higher percentages display larger bars.
Bar Width	Choose a width in pixels for bars on the chart.
Bar Height	This option applies to bars in both horizontally and vertically oriented charts Type a height in pixels for bars on the chart. This option applies to bars in both horizontally and vertically oriented charts

Value Format Mask

Type a format for numeric values or dates that appear on the chart.

Charts: Parameter Entry Form Display Options

Description Use this page to optionally display for each table or view column an entry field on the chart's parameter entry form. The entry field enables end users to choose their own condition for displaying data from the column on the chart.

For example, if you choose the DEPTNO column from the SCOTT.EMP table as a **Column Name** on this page, WebDB adds a entry field for the column to the chart's parameter entry form. End users can type a department number in the field to display only data about employees from that department on the chart.

You can optionally add a List of Values to the entry field. In the previous example, instead of requiring end users to type a numeric value, you could add a List of Values that enables them to choose 10, 20, or 30.

Other options on this page enable you to choose which buttons and options are displayed to the end user of the parameter entry form. For example, you can choose whether to display a Batch button that allows the end user to run the chart in batch mode, or an option that allows the end user to choose whether to display null values on the chart.

This page contains:

Value Required	Check to require the end user to specify a value for the column's entry field on the chart's parameter entry form. If you do not check this box, the end user is not required to specify a value.
Column Name	Choose a table or view column. An entry field will be added to the chart's parameter entry form that allows end users to specify values that limit the column's data displayed in the chart. If you do not choose a table or view column, an entry field for it does not appear on the parameter entry form.
Prompt	Type the prompt text you want to display next to the entry field. The prompt text tells end users what to enter in the field; for example

	Display all employees with IDs greater than this number:
LOV	Type the name of the List of Values you want to use for the column's entry field. End users of the parameter entry form can choose values from this list to limit data displayed in the chart. For example, you could add a List of Values containing ranges of Employee ID numbers: 1000-2000, 2000-3000, etc. Click  to the right of the text box to search for Lists of Values.
Display LOV As	Choose the List of Values format you want to provide the end user for selecting parameters for the column's entry field.
More Parameters	Click to display more fields on this page that allow you to add additional entry fields on the parameter entry form.
Axis	Check to enable end users of the parameter entry form to choose a method for displaying chart bars relative to the value of chart's axis. For example, if the end user chooses Zero, the value of the axis is set at 0. If the end user chooses Average Value, the axis is set at the average of all values in the Value column of the table on which the chart is based. Bars with values greater than average display above the axis; those with lower values display below the axis.
Include Nulls	Check to enable end users of the parameter entry form to specify whether to display null values in the chart.
Maximum Rows/Page	Check to enable end users of the parameter entry form to specify the maximum number of bars to display in the chart.
Summary	Check to enable end users of the parameter entry form to choose one or more options that display summary information about the chart.

Charts

	Each option the end user chooses is included in the summary information box at the bottom of the chart.
Type	Check to enable end users of the parameter entry form to choose a font for displaying chart text.
Run	Check to display a Run button on the chart's parameter entry form. End users can click the Run button to display the chart with the options they have specified in the parameter entry form.
Save	Check to display a Save button on the chart's parameter entry form. End users can click the Save button to save their option selections.
Batch	Check to display a Batch button on the chart's parameter entry form. End users can click the Batch button to run the chart in batch mode and save the results in the database.
Reset	Check to display a Reset button on the chart's parameter entry form. End users can click the Reset button to reset all entry fields to their default values.
Name	Type the label you want to display on the button; for example, Display Sales Chart . Tip Keep the Name short to avoid displaying large buttons.
Location	Choose the vertical location of the button on the parameter entry form. Tip Choose Don't Show if you don't want the button added to the parameter entry form.
Alignment	Choose whether to display the button on the left, center, or right margin of the parameter entry form.

Charts: Text Options

Description Use this page to choose text that you want to display at the top or bottom of the chart page or parameter entry form. You can also add help text for the chart or its parameter entry form. The end user clicks a ? button on the chart or parameter entry form to view the text.

This page contains:

Template	Choose a template to set the look and feel of chart and parameter entry form elements such as background colors and images and the image that appears in the upper left corner of the page.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the chart or parameter entry form.
Header Text	Type any introductory text that you want to display at the top of the chart or parameter entry form, just below the title.
Footer Text	Type any text that you want to display at the bottom of the chart or parameter entry form.
Help Text	Type any text that you want to display in a help page for the chart or parameter entry form. If you type text in this text box, WebDB adds a help button to the chart or parameter entry form. End users can click this button to link to a page displaying the help text.

Notes

- You can choose a template in addition to any options you choose in the Charts: Display Options page. Templates control the look and feel of the page on which the chart appears, whereas display options control the look and feel of the chart itself.

Charts

- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** entry fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

Charts: Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different points during the execution of the HTML code that creates the chart or parameter entry form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Enter the PL/SQL code to execute after displaying the header

Enter the PL/SQL code to execute after displaying the footer

Enter the PL/SQL code to run after displaying the page

Type or paste a PL/SQL procedure that will execute before the page containing the chart or parameter entry form displays.

Type or paste a PL/SQL procedure that will execute after the chart or parameter entry form header displays.

Type or paste a PL/SQL procedure that will execute after the chart or parameter entry form footer displays.

Type or paste a PL/SQL procedure that will run after the page containing the chart or parameter entry form displays.

SQL-based Charts

SQL-based Query Charts: Chart Name and Schema

Description Use this page to choose a name for the chart and the name of the database schema in which the finished chart will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished chart. Only schemas that you are allowed to build in are listed in the drop-down list.
Chart Name	Type the name you want to use to identify the database package containing the finished chart; for example, MY_CHART.

Notes

- The **Schema** becomes part of the URL that end users specify to display the component.
- Follow these guidelines when typing a **Chart Name**:
 - You must specify a name (null is not allowed).
 - The chart name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a chart MY_CHART, but not MY CHART.
 - You cannot name a chart with a PL/SQL reserved word; for example, COLUMNS, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.
- Do not insert a colon in the bind variable itself. For example, the report fails if your bind variable is:

```
select * from WEBDB.test_time
where time_stanp = :timevar
```

SQL-based Query Charts: SQL Query

Description Use this page to type or paste the SQL query that selects the table or view data to display in the chart. You specify up to three columns from a table or view to include the SELECT statement for the chart.

Column	Specifies	Function
1 (appears first in the SQL Query)	the _link (optional)	The chart uses data in this column to create links from the chart's labels to another WebDB component or URL. This column is optional. If you don't want to display links, you must specify null for the column.
2	the _label (required)	The chart displays data in this table or view column as labels for the bars on the chart. Labels are displayed along the bottom of a vertical chart and along to the left edge of a horizontal bar chart. They identify the bars on the chart.
3	the _value (required)	The chart uses data in this table or view column to calculate the relative size of the bars in the chart. The data in this column must be numeric.

This page contains:

Enter the SQL SELECT Statement

Type or paste a SQL SELECT statement.

Notes

Use alphanumeric strings preceded by colons as bind variables (for example, :var1, :var2, :var3...). All must **begin** with an alpha character (a, b, c, etc.) For each bind variable you specify, WebDB displays a parameter entry field in the chart's parameter entry form. The entry field prompts end users to choose conditions for displaying data in the chart.

SQL-based Charts

Do not add single or double quotes around bind variables. For example, do not specify `' :var1 '` or `" :var1 "` as a bind variable.

- Do not insert a colon in the bind variable itself. For example, the report fails if your bind variable is:

```
select * from WEBDB.test_time
where time_stanp = :timevar
```

- If you specify a bind variable in this page, you can associate a List of Values with it in the SQL-based Query Charts: Parameter Entry Fields page.
- Preface table names with their owning schema (for example, SCOTT.EMP for a table owned by SCOTT) if your statement includes tables owned by a schema other than the one that will own the finished chart.
- You can include relative HTML links by coding them into the SELECT list.
- Use aliases for long columns names (greater than 32 characters).

SQL-based Query Charts: Display Options

Description Use this page to choose options that control the appearance of the finished chart.

This page contains:

Run Options

Maximum Rows	Type the maximum number of bars you want to display in the chart.
Show Query Conditions	Check to display at the bottom of the chart all user-specified parameters passed to the query that created the chart and the time when the chart was created.
Paginate	Check to display on the chart a button labeled Next. Clicking the button allows the end user to see more bars on the chart. Note The maximum number of charts bars that the end user can see is set by the Maximum Rows option.
Log Activity	Check to log in the WebDB activity log the names of end users who request the chart as well as other performance information.
Show Timing	Check to display at the bottom of the chart the time from when the server received the request to generate the chart to when the HTML for the chart was generated.
Summary Options	Select one or more options that display summary information about the chart. Each option you select is included in the summary information box at the bottom of the chart. Note Windows users can choose more than one option by clicking it while pressing the Ctrl key.

Look and Feel Options

Type Face	Choose a font for displaying chart text.
-----------	--

SQL-based Charts

Font Size	Choose the size of the font for displaying chart text. Specify Font Size as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.
Font Color	Choose the color for displaying chart text.
Type	Choose whether to display the chart bars in a horizontal or vertical orientation.
Axis	Choose a method for displaying chart bars relative to the value of chart's axis. For example, if you choose Zero, the value of the axis is set at 0. If you choose Average Value, the axis is set at the average of all values in the Value column of the table on which the chart is based. Bars with values higher than average display above the axis; those with lower values display below the axis.
Bar Image	Choose an image that will be used to fill in the bars on the chart. Tip Choose MULTI to display each bar in a different color.
Chart Scale	Choose a % value to set the size of chart bars relative to the web page. Higher percentages display larger bars.
Bar Width	Choose a width in pixels for bars on the chart. This option applies to bars in both horizontally and vertically oriented charts.
Bar Height	Type a height in pixels for bars on the chart. This option applies to bars in both horizontally and vertically oriented charts
Value Format Mask	Type a format for numeric values or dates that appear on the chart.

SQL-based Query Charts: Parameter Entry Form Display Options

Description Use this page to optionally display parameter entry fields in the chart's parameter entry form. WebDB displays a parameter entry field for each table or view column for which you specified a bind variable on the SQL-based Charts: SQL Statement page. The entry field enables end users to choose conditions for displaying data in the chart.

For example, if you specified a bind variable for the DEPTNO column of the SCOTT.EMP table, the chart's parameter entry form displays an entry field for the column. End users can type a department number in the field to display only data about employees from that department on the chart.

You can optionally add a List of Values to the entry field. In the previous example, instead of requiring end users to type a numeric value, you could add a List of Values that enables them to choose 10, 20, or 30.

Other options on this page enable you to choose which buttons and options are displayed to the end user of the parameter entry form. For example, you can choose whether to display a Batch button that allows the end user to run the chart in batch mode, or an option that allows the end user to choose whether to display null values on the chart.

This page contains:

Bind Variable

Displays each bind variable you specified on the SQL Statement page. For each bind variable you specified, WebDB displays a parameter entry field in the calendar's parameter entry form. The entry form enables end users to choose conditions for displaying data in the calendar.

Prompt

Type the prompt text you want to display next to the entry field. The prompt text tells end users what to enter in the field; for example

Display all employees with
IDs greater than this number:

LOV

Type the name of the List of Values you want to use for the column's entry field. End users of the parameter entry form can choose values from this list to limit data displayed in the chart.

SQL-based Charts

For example, you could add a List of Values containing ranges of Employee ID numbers: 1000-2000, 2000-3000, etc.

Click  to the right of the text box to search for Lists of Values.

Display LOV As

Choose the List of Values format you want to provide the end user for selecting parameters for the column's entry field.

Show Axis

Check to enable end users of the parameter entry form to choose a method for displaying chart bars relative to the value of chart's axis. For example, if the end user chooses **Zero**, the value of the axis is set at 0.

If the end user chooses **Average Value**, the axis is set at the average of all values in the Value column of the table on which the chart is based. Bars with values greater than average display above the axis; those with lower values display below the axis.

Maximum Rows

Check to enable end users of the parameter entry form to specify the maximum number of bars to display in the chart.

Summary

Check to enable end users of the parameter entry form to choose one or more options that display summary information about the chart. Each option the end user chooses is included in the summary information box at the bottom of the chart.

Type

Check to enable end users of the parameter entry form to choose a font for displaying chart text.

Run

Check to display a Run button on the chart's parameter entry form. End users can click the Run button to display the chart with the options they have specified in the parameter entry form.

Save

Check to display a Save button on the chart's parameter entry form. End users can click the Save button to save their option selections.

Batch	Check to display a Batch button on the chart's parameter entry form. End users can click the Batch button to run the chart in batch mode and save the results in the database.
Reset	Check to display a Reset button on the chart's parameter entry form. End users can click the Reset button to reset all entry fields to their default values.
Name	Type the label you want to display on the button; for example, <code>Display Sales Chart</code> . Tip Keep the Name short to avoid displaying large buttons.
Location	Choose the vertical location of the button on the parameter entry form. Tip Choose Don't Show if you don't want the button added to the parameter entry form.
Alignment	Choose whether to display the button on the left, center, or right margin of the parameter entry form.

Chart summary options

Description Summary options summarize the data represented by the chart. Each option you choose is included in the summary information at the bottom of the chart. These options apply to charts you create using the Query Wizard as well as those you create by hand-coding a SQL query.

Average value	Choose to display the average of all values in the table or view Value column.
Axis name	Choose to display the option that the end user specified in the Chart Axis drop-down list on the chart's parameter entry form. Chart Axis allows the end user to select a method for displaying chart bars relative to the value of chart's axis.
Count of values	Choose to display the total number of values in the table or view Value column.
First value	Choose to display the first value in the table or view Value column.
Last value	Choose to display the last value in the table or view Value column.
Maximum value	Choose to display the maximum value in the table or view Value column.
Sum of values	Choose to display the sum of all values in the table or view Value column.

SQL-based Query Charts: Text Options

Description Use this page to choose text that you want to display at the top or bottom of the chart page or parameter entry form. You can also add help text for the chart or its parameter entry form. The end user clicks a **?** button to view the text.

This page contains:

Template	Choose a template to set the look and feel of chart and parameter entry form elements such as background colors and images and the image that appears in the upper left corner of the page.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the chart or parameter entry form.
Header Text	Type any introductory text that you want to display at the top of the chart or parameter entry form, just below the title.
Footer Text	Type any text that you want to display at the bottom of the chart or parameter entry form.
Help Text	Type any text that you want to display in a help page for the chart or parameter entry form. If you type text in this text box, WebDB adds a help button to the chart or parameter entry form. End users can click this button to link to a page displaying the help text.

Notes

- You can choose a template in addition to any options you choose in the Chart: Display Options page. Templates control the look and feel of the page on which the chart appears, whereas display options control the look and feel of the chart itself.

SQL-based Charts

- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** entry fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```


Chart parameter options

Description Parameter options enable you to choose which options appear on the parameter entry form for the chart. End users can choose these options when running the chart. These options apply to charts you create using the Query Wizard as well as those you create by hand-coding a SQL query.

Axis	Choose to enable the end user to specify a method for displaying chart bars relative to the value of chart's axis. For example, If the end user chooses Zero, the value of the axis is set to 0. Choose Average Value to set the axis to the average of all values in the Value column of the table on which the chart is based. Bars with values higher than average display above the axis; those with lower values display below the axis.
Include nulls	Allows the end user to display in the chart null Label column values.
Maximum rows/page	Choose to enable the end user to specify the maximum number of chart bars that display on the page. If there are more bars to display than are allowed on the page, the end user can click a button labeled Next to view more pages containing chart bars.
Summary	Choose to enable the end user to specify whether to display summary information about the chart, such as average chart value, sum of values, or count of value.
Type	Choose to enable the end user to specify whether to display the chart bars in a horizontal or vertical orientation.

SQL-based Chart Wizard: Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different points during the execution of the HTML code that creates the chart or parameter entry form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Type or paste a PL/SQL procedure that will execute before the page containing the chart or parameter entry form displays.

Enter the PL/SQL code to execute before displaying the chart

Type or paste a PL/SQL procedure that will execute before the chart or parameter entry form header displays.

Enter the PL/SQL code to execute after displaying the footer

Type or paste a PL/SQL procedure that will execute after the chart or parameter entry form footer displays.

Enter the PL/SQL code to run after displaying the chart

Type or paste a PL/SQL procedure that will run after the page containing the chart or parameter entry form displays.

Dynamic Pages

Dynamic Page Building

Description Use the Dynamic Page Building page to create a new dynamic page, or find an existing or recently edited dynamic page. After you find a dynamic page, you can edit it using the Edit Dynamic Page dialog box.

This page contains:

Create a New Dynamic Page

Create Click to create a new dynamic page.

Find an Existing Dynamic Page

Find in Schema Choose the schema that owns the dynamic page you want to find. The **Find in Schema** drop-down list displays all schemas you have privileges to browse.

Find Click to find all dynamic pages owned by the schema you specified.

Select a Recently Edited Dynamic Page

Name Displays the name of the five most recently created or edited dynamic pages. Click a **Name** to edit the dynamic page.

Schema Displays the schema that owns the dynamic page.

Type Displays the component type, i.e., **Dynamic page**

Changed Displays in days, hours, minutes, and seconds how long ago the dynamic page was created or last edited.

By Displays the developer who created or last edited the dynamic page.

Dynamic Pages: Dynamic Page Name

Description Use this page to choose a name for the dynamic page and the name of the database schema in which the finished dynamic page will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished dynamic page. Only schemas that you are allowed to build in are in the list.
Dynamic Page Name	Type the name you want to use to identify the database package containing the finished dynamic page; for example, MY_PAGE.

Notes

- The **Schema** becomes part of the URL that end users specify to display the component.
- Follow these guidelines when typing a **Dynamic Page Name**:
 - You must specify a name (null is not allowed).
 - The name must be unique within the schema.

Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a dynamic page MY_DYNAMIC_PAGE, but not MY DYNAMIC PAGE.

You cannot name a dynamic page with a PL/SQL reserved word; for example, COLUMN, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

Dynamic Pages: Dynamic Page Content

Description Use this page to type or paste HTML code that creates a web page. Then, add any PL/SQL code segments whose executed results display on the page. The PL/SQL code automatically runs every time an end user requests the page. Enclose each PL/SQL code segment in `<ORACLE>` `</ORACLE>` tags.

For example, you can create a page that automatically runs a SQL `SELECT` query on a database table, then displays the query results as the page content. You can also set up pages to automatically run functions and procedures, then display the results on the page.

If you have a large number of PL/SQL code segments and don't want to edit them all on this screen, the wizard provides the option of separately editing each segment you enclose between `<ORACLE>` `</ORACLE>` tags. In the example provided when the Dynamic Page Content page first displays, you have the option of editing the PL/SQL segment

```
select * from scott.emp
```

in the next page of this wizard.

Dynamic Page Wizard: Refine Page Content

Description Use this page to edit or add to any PL/SQL code segment you specified between `<ORACLE>` `</ORACLE>` tags in the Dynamic Page Contents page.

This page contains:

PL/SQL Code Segments	Displays the PL/SQL code segments you specified between the <code><ORACLE></code> <code></ORACLE></code> tags. You can edit this code.
----------------------	--

Dynamic Page Wizard: Log Activity Option

Description Use this page to choose whether you want to log performance and end user information in the WebDB activity log.

This page contains:

Log Activity

Check to log in the WebDB activity log the names of end users who request the page as well as other performance information.

Forms

Form Building

Description Use the Form Building page to create a new form, or find an existing or recently edited form. After you find a form, you can edit it using the Edit Forms dialog box.

This page contains:

Create a New Form

Forms on Stored Procedures	Click to create a form based on a PL/SQL procedure stored in the database.
Forms on Tables/Views	Click to create a form based on a database table or view.
Master-Detail Forms	Click to create a form that displays a master row and multiple detail rows.
Query by Example (QBE) Forms	Click to create a Query by Example form that provides end users options to query, delete, update, or insert values into a database table or view. Note: When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.
Create	Click to create the type of form you specified.

Find an Existing Form

Find in Schema	Choose the schema that owns the form you want to find. The Find in Schema drop-down list displays all schemas you have privileges to browse.
----------------	---

Find Click to find all forms owned by the schema you specified.

Select a Recently Edited Form

Name Displays the name of the five most recently created or edited forms. Click a **Name** to edit the form

Schema Displays the schema that owns the form.

Type Displays the form type, either:

- **Form (table)** - a form based on a table or view
- **Form (proc)** - a form based on a stored procedure
- **MD** - a Master-Detail form
- **QBE** - a Query by Example (QBE) form

Changed Displays in days, hours, minutes or seconds how long ago the form was created or last edited.

By Displays the name of the developer who created or last edited the form.

Note

- The WebDB Forms text areas can contain up to 32K.

Forms from Tables

Forms (from tables): Form Name and Schema

Description Use this page to choose a name for the form and the name of the database schema in which the finished form will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished form. Only schemas that you are allowed to build in are shown in the list.
Form Name	Type the name you want to use to identify the database package containing the finished form; for example, MY_FORM.

Notes

- The **Schema** name becomes part of the URL that end users can specify to display the component.
- Follow these guidelines when typing a **Form Name**:
 - You must specify a form name (null is not allowed).
 - The form name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a form MY_FORM, but not MY FORM.
 - You cannot name a form with a PL/SQL reserved word; for example, COLUMNS, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

Forms (from tables): Tables or Views

Description Use this page to choose the database tables or views on which you want to base the form.

This page contains:

Table/View

Choose the table or view whose data will be used to create the form. The list contains all tables and views on which you have SELECT, INSERT, UPDATE, or DELETE privileges.

The table name in the list is prefixed by the schema that owns it. For example, a table named EMP in the SCOTT schema appears in the list as SCOTT.EMP.

Notes

- You automatically have SELECT, INSERT, UPDATE, or DELETE privileges on a table or view if it is owned by a schema in which you have Build In privileges.
- When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

Forms (from tables): Form Layout Selection

Description Use this page to choose a structured or unstructured layout for the form:

- Structured layouts are based on options you choose in the Form Building wizard.
- Unstructured layouts are based on HTML code that you supply on a page in this wizard. Because you are specifying your own HTML code, you can create a more elaborate and sophisticated layout than you could using a structured layout.

This page contains:

Choose a layout for your form

Choose either a **Structured** or **Unstructured** layout.

Forms (from tables): Field-level Formatting and Validation

Description Use this page to choose the table or view columns you want to display in the form. WebDB adds an entry field to the form for each column you choose. For example, if you choose the EMPNO column from the SCOTT.EMP table, WebDB adds a field to the form that enables end users to update the SCOTT.EMP table based on values in the EMPNO column.

You can also specify options on this page that control the appearance of the entry field for each selected table or view column, and perform validation on the values that an end user specifies in an entry field.

The page you are looking at is divided into two frames. The left frame contains a list of the columns from the table or view you selected in the previous wizard step. The example shown below displays columns from the SCOTT.EMP table.

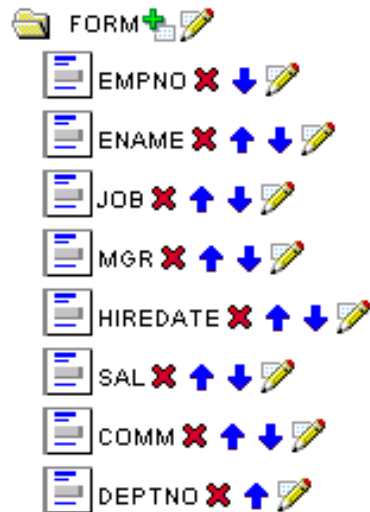


Table or view columns in the list are indicated by . All columns that appear in the list will display as fields in the finished form. At the top of the list is a folder  item

Forms

representing the finished form. The buttons that appear to the right of each item in the list indicate the actions you can perform on the form and columns.

Button	Allows you to:
	Add a field to the form.
	Delete a column from the list. Deleting a column from the list means that there won't be a corresponding entry field for the column in the finished form.
	Change the order in which the column entry fields display on the form. In general, the first column in the list displays at the top of the form, the last at the bottom.
	Edit options for the form, or individual entry fields within the form itself. If you click  next to a form item (), the right frame of this page updates with options that affect the entire form. If you click  next to a field item (), the right frame of this page updates with options that affect only the entry field for that column.

Field level options**Appearance**

Display Name	Type text that will appear as a label on the column's field on the form. For example, you can add a label next to the field for the EMPNO column called Employee Identification Number .
Display As	Choose a display format for the column's entry field on the form; for example, a combo box or pop-up list.
Width	Type a width for the field in characters.
Height	Type a height for the field in character rows.
Max Length	Type the maximum number of characters the end user can type in the field. Note If you specify a Max Length larger than the Width, existing text scrolls past the left edge of the field as the user types.
Font Color	Choose the color for the Display Name text.
Font Face	Choose a font for the Display Name text.
Font Size	Choose a font size for the Display Name text. Specify Font Size as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.

Forms

Link

Specify a link to another WebDB component or URL. If you specify this option, the **Display Name** appears on the form as a hypertext link.

- If you want to link the form to another URL, type its location in the **Link** text box; for example,

`http://www.oracle.com`

- If you want to link to a WebDB component, you can type the name of the package containing the component; for example,

`SCHEMA.COMPONENT.SHOW`

where **SCHEMA** is the name of the schema that owns the component,

COMPONENT is the component name,

and **SHOW** is the procedure used to display the component (you can also specify **SHOW_PARMs** to display the parameter entry form for the component).

OR you can click  next to the text box to choose from a list of links to components that have been defined in the Link Manager.

Layout

Row Span

Type how many HTML cells can be used to display the field vertically on the browser page

Column Span Type how many HTML cells can be used to display the field horizontally on the browser page. For example, specifying a Column Span of 3 tells the HTML browser to make the field occupy the same horizontal space as three cells above or below it.

Begin on New Line? Choose **Yes** to display the field for the column on a new line on the form. Choose **No** to display the column field on the same line as the previous column field.

Validation

Mandatory Choose **Yes** to require that the end user specify a value in the field before submitting the form.

Updateable Choose **Yes** to enable the end user to update the column. Choose **No** to prevent updates.

Tip This feature is useful if you want to allow end users to update some table columns in the form but only view others. For example, by setting **Updateable** to **No** for employee names and id numbers, and **Yes** for the employee's department, you can create a form that enables end users to update an employee's department when the employee is transferred, but never the employee name and id.

Note

- When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

Forms

LOV

Type the name of the List of Values you want to use for the field. End users of the form can choose values from this list instead of having to type the value in the field.

Note If you choose an LOV, you must also choose one of the following **Display As** types:

- Check box
- Combo box
- Pop up
- Radio Group

For example, you could add a List of Values containing ranges of Employee ID numbers: 1000-2000, 2000-3000, etc. and display this List of Values as a check box, combo box, pop-up list or radio group. Click  to the right of the text box to search for Lists of Values.

Default Value

Type a default value for the column's field. End users have the option of accepting this value or specifying their own.

Format Mask

Type an Oracle display format for columns containing numeric and date datatypes. For example, you could type DD/MM/YYYY to display dates according to this pattern, or 999,999,999.99 to place commas and decimals according to the pattern.

Note Refer to the Oracle documentation for additional information about date and numeric formatting options.

Field-level Validation

Choose a JavaScript validation application that verifies whether the end user enters a valid value in the field. For example, you could choose a JavaScript application called **IsNumber** that verifies that a number has been typed in a SALARY field.

Note Field validation applications are implemented in JavaScript and run when the OnBlur condition occurs; for example, when the end user presses the Return key after typing a value in the field.

Form-level Validation

Choose a JavaScript validation application that verifies whether the end user enters a valid value in the field.

Note Form validation applications run when the end user submits the information on the page to WebDB; for example, after clicking an Insert button on the form.

Note

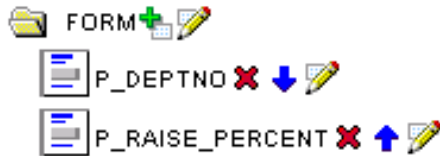
- The WebDB Forms text areas can contain up to 32K.

Forms (from procedures) Form-level Formatting and Validation

Description Use this page to choose options that control the look and feel of the form elements such as its background image and color, or text that appears at its top and bottom. You can also specify a PL/SQL procedure that executes after an end user enters values in the finished form; for example, a success message.

The page you are looking at is divided into two frames. The left frame contains a list of the arguments from the procedure you selected in the previous wizard step. Arguments in the list are indicated by . All arguments that appear in the list will display as fields in

the finished form. At the top of the list is a folder  item representing the finished form. The example shown below displays arguments from the GIVE_RAISE procedure in the SCOTT schema.



The buttons that appear to the right of each item in the list indicate the actions you can perform on the form and its arguments.

Button	Allows you to:
	Add an argument to the list. Adding an argument to the list adds a corresponding entry field on the form.
	Delete an argument from the list. Deleting an argument from the list means that there won't be a corresponding entry field for the argument in the finished form.



Change the order in which the argument entry fields display on the form. In general, the first argument in the list displays at the top of the form; the last at the bottom.



Edit options for the form, or individual entry fields within the form itself.

If you click  next to a form item (), the right frame of this page updates with options that affect the entire form.

If you click  next to a field item (), the right frame of this page updates with options that affect only the entry field for that argument.

The options shown below are for editing form-level options.

Form-level options

Log Activity	Check to log in the WebDB activity log the names of end users who request the form as well as other performance information.
Show Timing	Check to display at the bottom of the form the time from when the server received the request to generate the form to when the HTML for the form was generated.
Alternate ROWID	Optionally choose a table or view column that will be used as a primary key.
Box Background Color	Choose the background color of the form.
Box Background Image	Choose an image that will appear in the background of the form.

Forms

Box Border	Choose a style for the border around the background box.
Type Face	Choose a font type for displaying text on the form.
Font Color	Choose the color for displaying text on the form.
Font Size	Choose the size of the font for displaying text on the form. Specify Font Size as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.
On Successful Submission of a Form Execute this Code	Type optional PL/SQL code that will execute after an end user clicks a button on the form. The button must cause an operation such as INSERT to be performed on the table or view on which the form is based. For example, you might type PL/SQL code in this field that displays a message to the end user when a table row is successfully updated.

Notes

- The **Type Face**, **Font Color**, and **Font Size** options apply to all text on the form except labels next to entry fields. If you want to change the options for a label, click  next to a column in the left frame of this page. You can then choose field-level options in the right frame of this page that apply to the entry field label (i.e., the text that you specified in the **Display Name** field).
- The WebDB Forms text areas can contain up to 32K.

Forms (from tables): Button Options

Description Use this page to choose the buttons you want to display on the form. You can optionally choose up to four buttons that enable end users to perform the following actions:

- Insert - Insert a new row into the table or view on which the form is based, using the values that the end user specifies in the form's entry fields.
- Update - Update the table or view with the values the end user specifies in the form's entry fields.
- Delete - Delete a table or view row based on values the end user specifies in the form's entry fields.
- Query - Query the table or view on which the form is based.
- Reset - Reset all entry fields to their default values.

This page contains:

Name	Type the label you want to display next to the button; for example, <code>Insert New Table Row</code> . Tip Keep the Name short to avoid displaying large buttons.
Location	Choose the vertical location of the button on the form. Tip Choose Don't Show if you don't want the button added to the form.
Alignment	Choose whether to display the button on the left or right margin of the form.

Forms (from tables): Text Options

Description Use this page to choose text that you want to display at the top or bottom of the form.

This page contains:

Template	Choose a template to set the look and feel of form elements such as background colors and images and the image that appears in the upper left corner of the page.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the form.
Header Text	Type any introductory text that you want to display at the top of the form, just below the title.
Footer Text	Type any text that you want to display at the bottom of the form.
Help Text	Type any text that you want to display in a help page for the form. If you type text in this text box, WebDB adds a help button to the form. End users can click this button to link to a page displaying the help text.

Notes

- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

- The WebDB Forms text areas can contain up to 32K.

Forms (from table): Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different points during the execution of the HTML code that creates the form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Type or paste a PL/SQL procedure that will execute before the page containing the form displays.

Enter the PL/SQL code to execute before displaying the header

Type or paste a PL/SQL procedure that will execute before the form header displays.

Enter the PL/SQL code to execute after displaying the footer

Type or paste a PL/SQL procedure that will execute after the form footer displays.

Enter the PL/SQL code to run after displaying the page

Type or paste a PL/SQL procedure that will run after the page containing the form displays.

Enter the PL/SQL code to execute before processing the form

Type or paste a PL/SQL procedure that will execute before the form is processed.

Enter the PL/SQL code to run after processing the form

Type or paste a PL/SQL procedure that will run after the form is processed.

Note

- The WebDB Forms text areas can contain up to 32K.

Forms from Stored Procedures

Forms (from procedure): Form Name and Schema

Description Use this page to choose a name for the form and the name of the database schema in which the finished form will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished form. Only schemas that you are allowed to build in are listed in the drop-down list.
Form Name	Type the name you want to use to identify the database package containing the finished form; for example, MY _FORM.

Notes

- The **Schema** becomes part of the URL that end users specify to display the component.
- Follow these guidelines when typing a **Form Name**:
 - You must specify a form name (null is not allowed).
 - The form name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a form MY_FORM, but not MY FORM.
 - You cannot name a form with a PL/SQL reserved word; for example, COLUMNS, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

Forms (from procedure): Procedure Name

Description Use this page to choose the database stored procedure on which you want to base the form.

This page contains:

Procedure

Choose the procedure that will be used to create the form. The drop-down list contains all procedures on which you have EXECUTE privileges.

The procedure name in the list is prefixed by the schema that owns it. For example, a procedure named CALC-TAX in the SCOTT schema appears in the list as SCOTT . CALC - TAX.

Forms (from procedures): Form Layout Selection

Description Use this page to choose a structured or unstructured layout for the form:

- Structured layouts are based on options you choose in the Form Building wizard.
- Unstructured layouts are based on HTML code that you supply on a page in this wizard. Because you specify your own HTML code, you can create a more elaborate and sophisticated layout than you could using a structured layout.

This page contains:

Choose a layout for your form

Choose either a **Structured** or **Unstructured** layout.

Forms (from procedures): Form Layout Editor - Unstructured

Description Use this page to specify HTML code that will control the layout of your form. The text box on this page contains sample HTML code that creates a table. Each argument in the procedure you selected in the **Procedure Name** step of this wizard is formatted to display in a table row.

Arguments names are specified by the suffix .FIELD in the sample code; for example,

```
#P_DEPTNO.FIELD#
```

Labels to identify each argument name are specified by the suffix .LABEL in the sample code; for example,

```
#P_DEPTNO.LABEL#
```

You can update the sample code with any HTML of your own provided you do not change the argument names or labels (.LABEL suffix). You can delete argument names.

Note

- The WebDB Forms text areas can contain up to 32K.

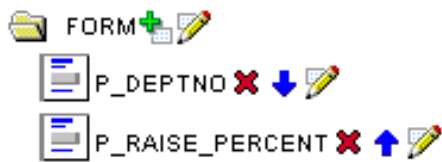
Forms (from procedures): Field-level Formatting and Validation

Description Use this page to choose the stored procedure arguments you want to display in the form. WebDB adds a field to the form for each argument you choose. For example, you could base the form on the SCOTT.GIVE_RAISE procedure. The procedure receives as input arguments a department number and a percentage to calculate a raise for each member of the department.

You can also specify a selectable List of Values, a display format, and code for validating each entry field on the form.

The page you are looking at is divided into two frames. The left frame contains a list of the arguments in the procedure you selected in the previous wizard step. Arguments in the list are indicated by . All arguments that appear in the list will display as fields in the finished form.

At the top of the list is a folder  item representing the finished form. The example shown below displays arguments from the GIVE_RAISE procedure in the SCOTT schema.



The buttons that appear to the right of each item in the list indicate the actions you can perform on the form and procedure arguments.

Button	Allows you to:
	Add an argument to the list. Adding an argument to the list adds a corresponding entry field on the form.

Forms



Delete an argument from the list. Deleting an argument from the list means that there won't be a corresponding entry field for the argument in the finished form.



Change the order in which the argument entry fields display on the form. In general, the first argument in the list displays at the top of the form, the last at the bottom.



Edit options for the form, or individual entry fields within the form itself.

If you click  next to a form item (), the right frame of this page updates with options that affect the entire form.

If you click  next to a field item (), the right frame of this page updates with options that affect only the entry field for that argument.

The options shown below are for editing field-level options.

Field-level options

Appearance

Display Name

Type text that will appear as a label on the argument's entry field on the form. For example, you can add a label next to the field for the P_DEPTNO argument column called `Enter Department Number`.

Display As

Choose a display format for the argument's entry field on the form; for example, a combo box or pop-up list.

Width

Type a width for the field in characters.

Height	Type a height for the field in character rows.
Max Length	Type the maximum number of characters the end user can type in the field. Note If you specify a Max Length larger than the Width, existing text scrolls past the left edge of the field as the user types.
Font Color	Choose the color for the Display Name text.
Font Face	Choose a font for the Display Name text.
Font Size	Choose a font size for the Display Name text. Specify Font Size as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.
Link	Specify a link to another WebDB component or URL. If you specify this option, the Display Name appears on the form as a hypertext link to the component or URL. · If you want to link the form to another URL, type its location in the Link text box; for example, <code>http://www.oracle.com</code> · If you want to link to a WebDB component, you can type the name of the package containing the component; for example, <code>SCHEMA . COMPONENT . SHOW</code> where SCHEMA is the name of the schema that owns the component,

COMPONENT is the component name,

and SHOW is the procedure used to display the component (you can also specify SHOW_PARMS to display the parameter entry form for the component).

OR you can click  next to the text box to choose from a list of links to components that have been defined in the Link Manager.

Layout

Row Span Type how many HTML cells can be used to display the field vertically on the browser page.

Column Span Type how many HTML cells can be used to display the field horizontally on the browser page. For example, specifying a Column Span of 3 tells the HTML browser to make the field occupy the same horizontal space as three cells above or below it.

Begin on New Line? Choose **Yes** to display the field for the argument on a new line on the form. Choose **No** to display the argument's field on the same line as the previous field.

Validation

Mandatory Choose **Yes** to require that the end user specify a value in the field before submitting the form.

Updateable Choose **Yes** to enable the end user to update the argument. Choose **No** to prevent updates.

Note When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

LOV

Type the name of the List of Values you want to use for the field. End users of the form can choose values from this list instead of having to type a value in the field.

Note If you choose an LOV, you must also choose one of the following **Display As** types:

- Check box
- Combo box
- Pop up
- Radio Group

For example, you could add a List of Values to the field for P_DEPTNO that displays 10, 20, 30, etc..

Click  to the right of the text box to search for Lists of Values.

Default Value

Type a default value that will appear in the argument's field on the form. End users have the option of accepting this value or specifying their own.

Format Mask

Type an Oracle display format for arguments containing numeric and date datatypes. For example, you could type DD/MM/YYYY to display dates according to this pattern, or 999,999,999.99 to place commas and decimals according to the pattern.

Note Refer to the Oracle documentation for additional information about date and numeric formatting options.

Forms

Field-level Validation

Choose a JavaScript validation application that verifies whether the end user enters a valid value in the field. For example, you could choose a JavaScript application called **IsNumber** that verifies that a number has been typed in a P_SALARY field.

Note Field validation applications are implemented in JavaScript and run when the OnBlur condition occurs; for example, when the end user presses the Return key after typing a value in the field.

Form-level Validation

Choose a JavaScript validation application that verifies whether the end user enters a valid value in the field.

Note Form validation applications run when the end user submits the information on the page to WebDB; for example, after clicking an Insert button on the form.

Forms (from procedures): Button Options

Description Use this page to choose the buttons you want to display on the form. You can optionally choose up to four buttons that enable end users to perform the following actions:

- Submit - Runs the procedure with the arguments the end user has specified on the form.
- Save - Saves the end-user's argument selections.
- Batch - Runs the procedure in batch mode and saves the results.
- Reset - Resets all argument entry fields to their default values.

This page contains:

Name	Type the label you want to display next to the button; for example, <code>Calculate tax</code> . Tip Keep the Name short to avoid displaying large buttons.
Location	Choose the vertical location of the button on the form. Note Choose Don't Show if you don't want the button added to the form.
Alignment	Choose whether to display the button on the left or right margin of the form.

Forms (from procedures): Text Options

Description Use this page to choose text that you want to display at the top or bottom of the form. You can also add help text for the form or its parameter entry form. The end user clicks a ? button to view the text.

This page contains:

Template	Choose a template to set the look and feel of form elements such as background colors and images and the image that appears in the upper left corner of the page.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the form.
Header Text	Type any introductory text that you want to display at the top of the form, just below the title.
Footer Text	Type any text that you want to display at the bottom of the entry form.
Help Text	Type any text that you want to display in a help page for the form. If you type text in this text box, WebDB adds a help button to the form. End users can click this button to link to a page displaying the help text.

Notes

- You can choose a template in addition to any options you choose in the Display Options page. Templates control the look and feel of the page on which the form appears, whereas display options control the look and feel of the form itself.
- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

- The WebDB Forms text areas can contain up to 32K.

Form (from procedures): Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different points during the execution of the HTML code that creates the form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Type or paste a PL/SQL procedure that will execute before the page containing the form displays.

Enter the PL/SQL code to execute before displaying the form

Type or paste a PL/SQL procedure that will execute before the form displays.

Enter the PL/SQL code to execute after displaying the footer

Type or paste a PL/SQL procedure that will execute after the form footer displays.

Enter the PL/SQL code to run after displaying the form

Type or paste a PL/SQL procedure that will run after the page containing the form displays.

Enter the PL/SQL code to execute before processing the form

Type or paste a PL/SQL procedure that will execute before the form is processed.

Enter the PL/SQL code to run after processing the form

Type or paste a PL/SQL procedure that will run after the form is processed.

Note

- The WebDB Forms text areas can contain up to 32K.

Master-Detail Forms

Master-Detail Forms: Master-Detail Form Name and Schema

Description Use this page to choose a name for the master-detail form and the name of the database schema in which the finished form will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished master-detail form. Only schemas that you are allowed to build in are listed in the drop-down list.
Form Name	Type the name you want to use to identify the database package containing the finished master-detail form; for example, MY_MD_FORM.

Notes

- The **Schema** becomes part of the URL that end users specify to display the component.
- Follow these guidelines when typing a **Form Name**:
 - You must specify a form name (null is not allowed).
The form name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a form MY_MD_FORM, but not MY MD FORM.
 - You cannot name a form with a PL/SQL reserved word; for example, COLUMNS, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

Master-Detail Forms: Tables or Views

Description Use this page to choose the database tables or views on which you want to base the master and detail rows of the form

This page contains:

Master Table/View	Choose the table or view on which the master rows of the form will be based.
Detail Table/View	Choose the table or view on which the detail rows of the form will be based.

Notes

- The drop-down lists contain all tables and views on which you have SELECT, INSERT, UPDATE, or DELETE privileges. You automatically have SELECT, INSERT, UPDATE, or DELETE privileges on a table or view if it is owned by a schema in which you have Build In privileges.
- Table names in the drop-down lists are prefixed by the schema that owns them. For example, SCOTT.EMP is a table named EMP in the SCOTT schema. By default, the SCOTT. DEPT master table and the SCOTT.EMP detail table display, but you can choose different tables.
- When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

Master-Detail Forms: Join Conditions

Description Use this page to choose join conditions between the master table or view and the detail table or view that you selected on the Table or Views page.

The columns shown in the two sets of combo boxes in this page are contained in the tables or views you selected. For example, a department number master column named DEPTNO in the DEPT table appears in the combo box as DEPT.DEPTNO.

To specify a join condition, choose a column from one of the left **Master Column** drop-down lists and a column in the **Detail Column** combo box directly to the right of it.

This page contains:

Master Column	Choose the master table or view column that you want to include in the JOIN condition of the SQL query used to build the master-detail form.
Detail Column	Choose the detail table or view column that you want to include in the JOIN condition of the SQL query used to build the master-detail form.

Notes

- WebDB bases the default join conditions shown in this page on the tables or views you selected in the Tables or Views page or the Master-Detail Form Wizard. You can accept these or choose different join conditions.
- Rows in one table may be joined to rows in another if common values exist in corresponding master and detail table or view columns.
- When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

Master-Detail Forms: Master Row Formatting

Description Use this page to choose the master table or view columns you want to display in the form. WebDB adds a field to the master-detail form for each column you choose to display.

For example, if you choose the DEPTNO column from the SCOTT.DEPT table WebDB adds an entry field to the form that enables end users to query or update the DEPT table based on values in the DEPTNO column. End users can also query or update multiple detail table or view rows based on the value in the DEPTNO column.

You can also specify a selectable List of Values, a display format, and code for validating each entry field on the form.

This page contains:

Column	Displays the names of all columns in the master table or view you selected in the Tables or Views page of the Master-Detail Form Wizard.
Display Name	Type text that identifies the column's entry field on the form. For example, you can add a display name next to the entry field for the DEPTNO column called <code>Department Number</code> .
Order	Type a number beginning with 1 to set the order in which master columns display on the form. For example, type 1 to display the column in the first position on the form, 2 to display the column in the second position, and so on. You can use decimals to order columns: 1.1, 1.15, 1.5, 2, etc. Tip For tables with many columns, it's easier to insert a decimal number than to reorder whole numbers.
Disp	Check to display the entry field for the master column on the form. Unchecking the check box prevents end users of the form from querying or updating values in the column.

Upd	Check to enable the end user to update the master column. Unchecking the check box prevents end users of the form from updating values in the column.
Begin on New Row	Check to position the column's entry field on a new row on the form.
Width	Type a width for the entry field in characters.
Max	Type the maximum number of characters the end user can type in the entry field. Note If you specify a Max larger than the Width, existing text scrolls past the left edge of the entry field as the user types.
Lines	Type the number of horizontal rows you want to display in the column's entry field. This option enables you to specify the height of an entry field. This option is useful for entry fields in which you expect end users of the form to enter a large amount of text.
Default	Type a default value for the column's entry field. End users have the option of accepting this value or specifying their own.
List of Values	Choose the List of Values type you want associate with the column's entry field on the form. End users of the form can choose values from the List of Values. For example, you could add a List of Values containing department numbers: 10, 20, or 30. Click  to the right of the text box to search for Lists of Values.

Forms

Form Validation

Type the name of a JavaScript validation application that verifies whether the end user enters a valid value in a field on a page.

Click  to the right of the text box to search for JavaScripts.

Note Form validation applications run when the end user submits the information on the page to WebDB; for example, after clicking an Insert button on the form.

Order Master Rows by

Choose the table or view column whose values will be used to sort master rows in the master-detail form. Choosing this option is equivalent to specifying a SQL ORDER BY clause.

Choose **Ascending** to sort query results according to the alphabetic (A-Z) or numeric (starting with the lowest number) order of column values, depending on the datatype for the column. Choose **Descending** to sort in the reverse order.

then by

Choose additional columns whose values will be used to sort master rows. For example, if you choose **Order by** Department ID, **then by** Employee Name, WebDB sorts master rows numerically using department IDs. Rows containing the same Department ID are then sorted alphabetically using employee names.

Note

- When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

Master-Detail Forms: Detail Row Formatting

Description Use this page to choose the detail table or view columns you want to display in the form. WebDB adds a field to the master-detail form for each column you choose. The entry fields appear at the bottom of the form in tabular format, as shown below.

Emp#	Name	Job Manager	Hiredate	Salary	Comm	Deptno	Delete
7876	ADAMS			1100			☐
7902	FORD			3000			☐
7566	JONES			2975			☐
7788	SCOTT			3000			☐
7369	SMITH			800			☐
							☐
							☐

For example, if you choose the DEPTNO column from the DEPT table WebDB adds a field to the form that enables end users to query or update the DEPT table based on values in the DEPTNO column. End users can also query or update multiple detail table or view rows based on values in the entry fields in the master row.

You can also specify a selectable List of Values, a display format, and code for validating each entry field on the form.

This page contains:

Column

Display Name	Type text that identifies the column's entry field on the form. For example, you can add a display name next to the entry field for the DEPTNO column called Department Number .
Order	Type a number beginning with 1 to set the order in which detail columns display on the form. For example, type a 1 to display the column in the first position on the form, a 2 to display the column in the second position, and so forth. You can use decimals to order columns: 1.1, 1.15, 1.5, 2, etc. Tip For tables with large numbers of columns, it's easier to insert a decimal number than to reorder whole numbers.
Disp	Check to display the entry field for the detail column on the form. Unchecking the Disp check box prevents end users of the form from querying or updating values in the column
Upd	Check allow the end user to update the detail column. Unchecking the Upd check box prevents end users of the form from updating values in the column.
Width	Type a width for the entry field in characters.
Max	Type the maximum number of characters the end user can type in the entry field. Note If you specify a Max larger than the Width, existing text scrolls past the left edge of the entry field as the user types.
Lines	Type the number of additional lines you want to display in the columns entry field. This option enables you to specify the height of an entry field.

	This option is useful for entry fields in which you expect end users of the form to enter a large amount of text.
Default	Type a default value for the column's entry field. End users have the option of accepting this value or specifying their own.
List of Values	Choose the List of Values type you want to associate with the column's entry field on the form. End users of the form can choose values from the List of Values. For example, you could add a List of Values containing department numbers: 10, 20, or 30. Click  to the right of the text box to search for Lists of Values.
Form Validation	Type the name of a JavaScript validation application that verifies whether the end user enters a valid value in a field on a page. Click  to the right of the text box to search for JavaScript routines. Note Form validation applications run when the end user submits the information on the page to WebDB; for example, after clicking an Insert button on the form.
Format Mask	Choose an Oracle display format for columns containing date and number data types. For example, if you choose DD/MM/RR, the date January 2nd, 1999 displays as 02/01/99. If you choose 999G999G999G999D99999999, the display format for numbers is 12 digits to the left of the decimal point and 8 to the right. Note Refer to the Oracle documentation for additional information about date formatting options.
Order Detail Rows by	Choose the table or view column whose values will be used to sort detail rows in the master-detail form. Choosing this option is equivalent to specifying a SQL ORDER BY clause.

Forms

Choose **Ascending** to sort query results according to the alphabetic (A-Z) or numeric (starting with the lowest number) order of column values, depending on the datatype for the column. Choose **Descending** to sort the query results in the reverse order.

then by

Choose additional columns whose values will be used to sort detail rows. For example, if you choose **Order by** Department ID, **then by** Employee Name, WebDB sorts detail rows numerically using department IDs. Rows containing the same Department ID are then sorted alphabetically using employee names.

Note

- When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

Master-Detail Forms: Display Options

Description Use this page to choose options that control the appearance of the finished master-detail form.

This page contains:

Run Options

Log Activity	Check to log in the WebDB activity log the names of end users who request the master-detail form as well as other performance information.
Show Timing	Check to display at the bottom of the master-detail form the time from when the request to generate the form was received by the server to when the HTML for the form was generated.
Show read only NULLs as	Type text that you want to display for values on the form that are read-only (cannot be updated) and contain null values; for example, (null).
Blank Detail Lines on Insert	Type the number of blank detail rows you want to display on the master-detail form. End users of the form can type values in these fields, then click the Insert button to insert a new row in the detail table.
Maximum Detail Rows	Type the number of detail rows you want to display on the master-detail form. End users will be able to view additional rows by clicking a button labeled Next on the master-detail form.

Delete Options

Cascade Delete of Master to Details	Choose Yes to delete all detail rows when the end user requests the master row be deleted. Note If you are using cascading deletes for your master table in the database, you do not need to select this option.
-------------------------------------	---

If you are not using cascading deletes and are using referential integrity in the database, attempting to delete a master row when details exist will result in an error unless you choose **Yes**.

Delete Detail Row Header

Type text that will display next to each detail row on the form. An end user can click this text to delete the row from the table or view on which the master-detail form is based.

Primary Key Options

Master Table Primary Key

Optionally choose a primary key column in the master table. The primary key will be used to perform operations on the master table such as INSERT or DELETE.

If you don't choose a table or view column, ROWID will be used as the primary key.

Detail Table Primary Key

Optionally choose a primary key column in the master table. The primary key will be used to perform operations on the detail table such as INSERT or DELETE.

If you don't choose a table or view column, ROWID will be used as the primary key.

Look and Feel Options

Type Face

Choose a font for displaying text on the master-detail form.

Font Size

Choose the size of the text font on the master-detail form.

Specify **Font Size** as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.

Font Color

Choose the color of text displayed on the master-detail form.

Master Box Background Color

Choose a background color for the box around master columns on the master-detail form.

Master Box Background Image	Choose a background image to display in the box around master columns on the master-detail form.
Master Box Border	Choose a style for the border around master columns on the master-detail form.
Detail Box Background Color	Choose a background color for the box around detail columns on the master-detail form.
Detail Box Background Image	Choose a background image to display in the box around detail columns on the master-detail form.
Detail Box Border	Choose a style for the border around detail columns on the master-detail form.

Master-Detail Forms: Button Options

Description Use this page to choose the buttons you want to display on the master-detail form. You can choose buttons that enable end users to perform the following actions:

- Save - Saves the end user's selections on the master-detail form.
- Delete - Deletes a table or view row based on values the end user specifies in the master-detail form's entry fields.

This page contains:

Name	Type the label you want to display next to the button; for example, Insert New Table Row .
Location	Choose the vertical location of the button on form. Note Choose Don't Show if you don't want the button added to the form.
Alignment	Choose whether to display the button on the left or right margin of the form.

Master-Detail Forms: Master Row Finder

Description Use this page to choose text that you want to display at the top or bottom of the Master Row Finder page. You can also add help text for the Master Row Finder page. The end user clicks a ? button on the page to view the text.

The Master Row Finder page enables end users to query the master table for master rows to display in the master-detail form. End users can also use this page to insert new rows into the master table.

This page contains:

Add Button Name	Type text you want to accompany a button on the Master Row Finder. The button enables end users to add a new row to the master table.
Query Button Name	Type text you want to accompany a button on the Master Row Finder that enables end users to query the master table. The query results are displayed in the master-detail form.
Title	Type text you want to display in the banner at the top of the Master Row Finder page.
Header Text	Type any introductory text that you want to display at the top of the Master Row Finder page, just below the title.
Footer Text	Type any text that you want to display at the bottom of the Master Row Finder page.
Form Help Text	Type any text that you want to display in a help page for the Master Row Finder page. If you type text in this text box, WebDB automatically adds a help button to the Master Row Finder page. End users can click this button to link to a page displaying the help text.
Show Order By	Check to enable end users of the Master Row Finder page to choose table or view columns whose values that will be used to sort rows in the query results, and the sort order.
Maximum Rows/Page	Check to enable end users of the Master Row Finder page to specify the maximum number of table or view rows to display on the query results page.

Master-Detail Forms: Text Options

Description Use this page to choose text that you want to display at the top or bottom of the Master Row Finder results page or the Master-Detail form. The results page displays the results of a query of the master table that the end user enters using the Master Row Finder.

You can also add help text for the results page or master-detail form. The end user clicks a ? button on the page to view the text.

This page contains:

Template	Choose a template to set the look and feel of results page and master-detail form elements such as background colors and images, and the image that appears in the upper left corner of the page.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the Master Row Finder Results page or the master-detail form page.
Header Text	Type any introductory text that you want to display at the top of the results page and the master-detail form page. If you type this text in the Master Row Finder Results text box, it appears at the top of the Master Row Finder Results page or the master-detail form page, just below the title.
Footer Text	Type any text that you want to display at the bottom of the of the results page or the master-detail form page.
Help Text	Type any text that you want to display in a help page for the of the results page or the master-detail form page. If you type text in this text box, WebDB automatically adds a help button to the of the results page and the master-detail form page.

End users can click this button to link to a page displaying the help text.

Notes

- You can choose a template in addition to any options you choose in the Display Options page. Templates control the look and feel of the pages on which the master-detail form and the results appear. Display options control the look and feel of the master-detail form and results.
- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** entry fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

- The WebDB Forms text areas can contain up to 32K.

Master-Detail Forms: Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different times during the execution of the HTML code that creates the Master Row Finder page and the master-detail form page.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Type or paste a PL/SQL procedure that will execute before the pages containing the Master Row Finder page or the master-detail form display.

Enter the PL/SQL code to execute after displaying the header

Type or paste a PL/SQL procedure that will execute after the Master Row Finder page or the master-detail form headers display.

Enter the PL/SQL code to execute before displaying the footer

Type or paste a PL/SQL procedure that will execute before the Master Row Finder page or the master-detail form displays

Enter the PL/SQL code to run after displaying the page

Type or paste a PL/SQL procedure that will after the Master Row Finder page or the master-detail form display.

Enter the PL/SQL code to execute before processing the form

Type or paste a PL/SQL procedure that will execute before the form is processed.

Enter the PL/SQL code to run after processing the form

Type or paste a PL/SQL procedure that will run after the form is processed.

Note

- The WebDB Forms text areas can contain up to 32K.

Query by Example Forms

Query by Example Forms: QBE Name and Schema

Description Use this page to choose a name for the Query by Example form and the name of the database schema in which the finished Query by Example form will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished Query by Example form. Only schemas that you are allowed to build in are listed in the drop-down list.
QBE Form Name	Type the name you want to use to identify the database package containing the finished Query by Example form; for example, MY_QBE_FORM.

Notes

- The **Schema** becomes part of the URL that end users specify to display the component.
- Follow these guidelines when typing a **QBE Form Name**:
 - You must specify a form name (null is not allowed).
 - The form name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a form MY_QBE_FORM, but not MY QBE FORM.
 - You cannot name a form with a PL/SQL reserved word; for example, COLUMNS, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

Query by Example Forms: Table or View

Description Use this page to choose the database table or view on which you want to base the Query by Example form.

This page contains:

Table/View

Choose the table or view whose data will be used to create the Query by Example form. The drop-down list contains all tables and views on which you have SELECT, INSERT, UPDATE, or DELETE privileges.

The table name in the list is prefixed by the schema that owns it. For example, a table named EMP in the SCOTT schema appears in the drop-down list as SCOTT.EMP.

Notes

- You automatically have SELECT, INSERT, UPDATE, or DELETE privileges on a table or view if it is owned by a schema in which you have Build In privileges.
- When end users display views, updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

Query by Example Forms: Table/View Columns

Description Use this page to choose the table or view columns you want to display in the Query by Example form. You also specify in this page the order in which columns display on the form.

This page contains:

Table/View Columns

Choose one or more table or view columns whose data you want to display in the Query by Example form.

Note Windows users can select more than one option by clicking it while pressing the Ctrl key.

Query by Example Forms: QBE Results Formatting

Description Use this page to choose how results returned by the Query by Example (QBE) form display on the results page.

This page contains:

Column	Displays the names of the table or view columns you selected in the previous page.
Column Heading Text	Type the heading name you want to use to identify the column in the QBE results page. For example, instead of displaying the table column name EMPNO in the results page, you could specify the more descriptive <code>Employee ID Number</code> as column heading text.
Sum	Check to sum values within the column and display the result in the QBE results page. Note This option is valid only for columns containing numeric data.
Align	Choose whether to align data to the left, center, or right margin of a QBE results page column. By default, numeric data align to the left, and alphabetical data align to the right.
Format Mask	Type an Oracle display format for columns containing numeric and date datatypes. For example, you could type <code>DD/MM/YYYY</code> to display dates according to this pattern, or <code>999,999,999.99</code> to place commas and decimals according to the pattern. Note Refer to the Oracle documentation for additional information about date and numeric formatting options.
Fixed Size	Type how many spaces wide you want the column to appear on the QBE results page. Note This option applies only to forms formatted for output as ASCII-style text.

Query by Example Forms: Display Options

Description Use this page to choose options that control the appearance of the Query by Example (QBE) form.

This page contains:

Run Options

Maximum Rows	Type the maximum number of rows you want to display in the QBE results page.
Show Null Values As	Type the text string you want to display for all null values in the QBE results page; for example, (null).
Draw Lines Between Rows	Check to display lines between QBE results page rows.
Show Query Conditions	Check to display at the bottom of the QBE results page all user-specified parameters passed to the query that created the QBE results page, and the time when the QBE results page was created.
Paginate	Check to display on the QBE results page a button labeled Next. Clicking the button allows the end user to see additional rows. The maximum number of rows that the end user can see is set by the Maximum Rows option.
Log Activity	Check to log in the WebDB activity log the names of users who request the QBE results page as well as other performance information.
Show Timing	Check to display at the bottom of the QBE results page the time from when the request to generate the QBE results page was received by the server to when the HTML for the QBE results page was generated.

Forms

Default Format

Choose a display format for the QBE results page:

- HTML

Formats the QBE results page using HTML tables and displays output on a new page in the web browser. Components that contain large amounts of data may take longer to display in this format.

- Excel

Downloads the QBE results page for display in Microsoft Excel.

- ASCII

Formats the QBE results page using the HTML PRE tag to display heading and values in the QBE results page as ASCII text. This option is useful for displaying large amounts of data.

Parameter Options

Choose parameter options that will appear on the parameter entry form for the QBE form. End users can choose these options when running the form.

Border

Choose a border style for the QBE form.

Break Options

Choose whether to break the QBE results on values in one, two, or three table or view columns.

Look and Feel Options

Type Face

Choose a font for displaying QBE results page text.

Font Size

Choose the size of the font for displaying QBE results page text.

Specify **Font Size** as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.

Font Color	Choose the color of the font for displaying QBE results page text.
Heading Background Color	Choose a background color for column headings in the QBE results page.
Table Background Color	Choose a background color for values that appear in the QBE results page body.

Row Order Options

Order by	Choose the table or view column whose values will be used to sort rows in the QBE results page. Choosing this option is equivalent to specifying a SQL ORDER BY clause. Choose Ascending to sort query results according to the alphabetic (A-Z) or numeric (starting with the lowest number) order of column values, depending on the datatype for the column. Choose Descending to sort in the reverse order.
then by	Choose additional columns whose values will be used to sort QBE results page rows. For example, if you choose Order by Department ID, then by Employee Name, WebDB sorts QBE results page rows numerically using department IDs. Rows containing the same Department ID are then sorted alphabetically using employee names.

Query by Example Forms: Button Options

Description Use this page to choose the buttons you want to display on the Query by Example (QBE) form. You can optionally choose up to five buttons that enable end users to perform the following actions:

- Query - Queries the table or view on which the QBE form is based.
- Save - Saves the end user's selections.
- Batch - Runs the query in batch mode and saves the results.
- Reset - Resets all QBE form entry fields to their default values.
- Insert - Inserts a new row into the table or view on which the QBE form is based .

This page contains:

Name	Type the label you want to display next to the button; for example, <code>Insert New Table Row</code> . Tip Keep the Name short to avoid displaying large buttons.
Location	Choose the vertical location of the button on form. Note Choose Don't Show if you don't want the button added to the form.
Alignment	Choose whether to display the button on the left or right margin of the form.

Query by Example Forms: Text Options

Description Use this page to choose text that you want to display at the top or bottom of the Query by Example (QBE) form or QBE Results page. You can also add help text for the Example (QBE) form or QBE Results page. The end user clicks a ? button on the page to view the text.

This page contains:

Template	Choose a template to set the look and feel of QBE form and Results page elements such as background colors and images, and the image that appears in the upper left corner of the page.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the QBE form or QBE Results page.
Header Text	Type any introductory text that you want to display at the top of the QBE form or QBE Results page, just below the title.
Footer Text	Type any introductory text that you want to display at the bottom QBE form or QBE Results page.
Help Text	Type any introductory text that you want to display in a help page for the QBE form or QBE Results page. WebDB automatically adds a help button to the QBE form or QBE Results page. End users can click this button to link to a page displaying the help text.

Forms

Notes

- You can choose a template in addition to any options you choose in the QBE Forms: Display Options page. Templates control the look and feel of the pages on which the QBE form or QBE Results appear, whereas display options control the look and feel of the QBE form or QBE Results.
- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** entry fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

Query by Example Forms: Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different points during the execution of the HTML code that creates the Query by Example or parameter entry form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Type or paste a PL/SQL procedure that will execute before the page containing the QBE form or QBE Results page displays.

Enter the PL/SQL code to execute before displaying the form

Type or paste a PL/SQL procedure that will execute before the QBE form or QBE Results page displays.

Enter the PL/SQL code to execute before displaying the footer

Type or paste a PL/SQL procedure that will execute before the QBE form or QBE Results page footer displays.

Enter the PL/SQL code to run after displaying the page

Type or paste a PL/SQL procedure that will run after the QBE form or QBE Results page displays.

Frame Drivers

Frame Driver Building

Description Use the Frame Driver Building page to create a new frame driver, or find an existing or recently edited frame driver. After you find a frame driver, you can edit it using the Edit Frame Driver dialog box.

This page contains:

Create a New Frame Driver

Create	Click to create a new frame driver. You must know how to create a SQL query to create a frame driver using this wizard.
--------	---

Find an Existing Frame Driver

Find in Schema	Choose the schema that owns the frame driver you want to find. The Find in Schema drop-down list displays all schemas you have privileges to browse.
----------------	---

Find	Click to find all frame drivers owned by the schema you specified.
------	--

Select a Recently Edited Frame Driver

Name	Displays the names of the five most recently created or edited frame drivers. Click a Name to edit the frame driver.
Schema	Displays the schema that owns the frame driver.
Type	Displays the component type, i.e. Frame Driver.
Changed	Displays in days, hours, minutes, and seconds how long ago the frame driver was created or last edited.
By	Displays the developer who created or last edited the frame driver.

Frame Drivers: Frame Driver Name and Schema

Description Use this page to choose a name for the frame driver and the name of the database schema in which the finished frame driver will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished frame driver. Only schemas that you are allowed to build in are listed in the drop-down list.
Frame Driver Name	Type the name you want to use to identify the database package containing the finished frame driver; for example, MY_FRAME_DRIVER.

Notes

- The **Schema** becomes part of the URL that end users can specify to display the component.
- Follow these guidelines when typing a **Frame Driver Name**:
 - You must specify a frame driver name (null is not allowed).
 - The name must be unique within the schema.

Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a frame driver MY_FRAME_DRIVER, but not MY FRAME DRIVER.

You cannot name a frame driver with a PL/SQL reserved word; for example, COLUMN, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

Frame Drivers: SQL Query for Frame Driver

Description Use this page to type or paste a SQL query, whose results will populate this frame. The SQL query must select two columns from a table or view. The first column is used to display selectable values in the driving frame. For example, the following query

```
select ename, empno from scott.emp
```

selects values from the employee name column from the SCOTT.EMP table to display in the driving frame.

The second column is used to display content in the target frame. Values in the EMPNO column of SCOTT.EMP display as text in the target frame.

In the second column of your SQL query, you can specify plain text, HTML code, PL/SQL, or a link to another URL to generate the contents of the target frame. After you create the SQL query, you must identify the target content type by choosing one in the **Target Link Type** drop-down list.

This page contains:

Specify the query that will populate the driving frame

Type or paste a SQL SELECT statement.

Target Link Type

Specify the method you used in your SQL query to create the contents of the target frame.

- PL/SQL - Choose this if you specified PL/SQL code in the second column of your SQL query.
- HTML /TEXT - Choose this if you specified either plain text or HTML code in the second column of your SQL query.
- URL - Choose this if you specified a link to a URL in the second column of your SQL query.

Display LOV As

Choose the List of Values format you want to provide to the end user for selecting display values in the driving frame. In the example at the top of this page, you could choose a radio button group to display all employee names in SCOTT.EMP as radio buttons in the driving frame.

Show NULL	Click to display any null values contained in the column you selected for the driving frame.
NULL text	Type the text string you want to display for all null values in the column you selected for the driving frame; for example, (null) .
NULL value	Type any value you want to use for all null values in the column you selected for the driving frame; for example, 0.

Notes

Use alphanumeric strings preceded by colons as bind variables (for example, :var1, :var2, :var3...). All bind variables must **begin** with an alpha character (a, b, c, etc.). For each bind variable you specify, WebDB displays a parameter entry field in the chart's parameter entry form. The entry field prompts end users to choose conditions for displaying data in the chart.

Do not add single or double quotes around bind variables. For example, do not specify ':var1' or ":var1" as a bind variable.

- Do not insert a colon in the bind variable itself. For example, the report fails if your bind variable is:

```
select * from WEBDB.test_time
where time_stanp = :timevar
```

- If you specify a bind variable in this page, you can associate a List of Values with it in the Frame Drivers: Parameter Entry Fields page.
- Preface table names with their owning schema (for example, SCOTT.EMP for a table owned by SCOTT) if your statement includes tables owned by a schema other than the one that will own the finished frame driver.
- You can include relative HTML links by coding them into the SELECT list.
- Do not include an ORDER BY clause if you plan to specify one in the Frame Driver: Display Options page.
- Use aliases for long columns names (greater than 32 characters).

Frame Drivers: Parameter Entry Form Display Options

Description Use this page to optionally specify a prompt and List of Values for each table or view column that you associated with a bind variable in the SQL Query for Frame Driver page.

This page contains:

Bind Variable	Displays the bind variables that you specified in the SQL Query for Frame Driver page. For each bind variable you specified, WebDB displays a parameter entry field in the frame driver's parameter entry form. The entry form enables end users to choose conditions for displaying data in the driving frame of the frame driver.
Prompt	Type the prompt text you want to display next to the entry field. The prompt text tells end users what to enter in the field; for example, Choose values you want to display in the driving frame.
LOV	Choose a List of Values for the parameter entry field. End users of the parameter entry form can choose values from this list to limit data displayed in the frame driver.
Display LOV As	Choose the List of Values format you want to provide to the end user for selecting parameters.

Frame Drivers: Initial Page in Target Frame

Description Use this page to choose the initial content of the target frame. This content appears in the target frame before the end user selects any options in the driving frame. The content can be anything. For example, you can add content that describes the options in the driving frame, displays a web page from another source, or a logo.

This page contains:

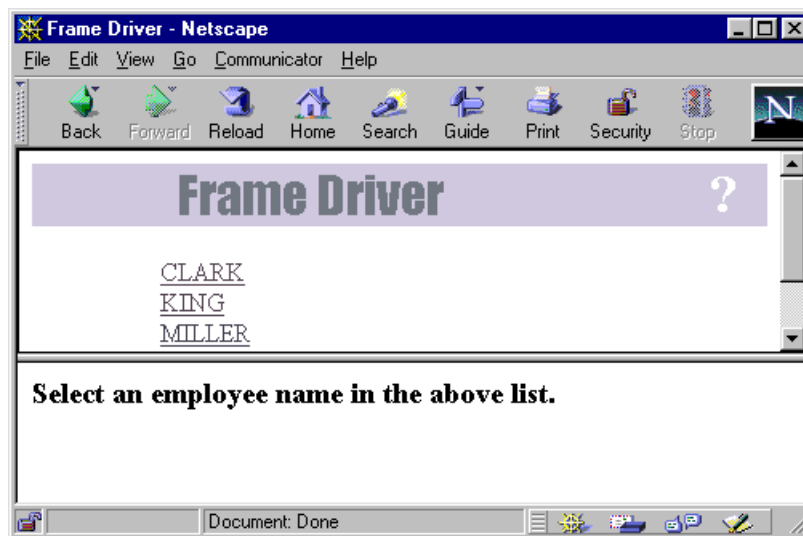
Enter text, HTML, PL/SQL, or a URL that will generate the initial page in the target frame

Specify a method for displaying the initial content of the target frame. This content appears in the target frame before the end user of the frame driver selects any options in the driving frame. The content can be any of the following:

- PL/SQL code
- HTML or TEXT
- a link to another URL

If you leave this entry field blank, the initial target frame that displays is empty.

In the example below, the content is HTML-formatted text.



Frame Drivers

Type

Specify the method you chose to display the initial content in the target frame;

- using PL/SQL code.
- using HTML code or plain text.
- by specifying a link to another URL.

Note Choose HTML/Text as the **Type** if you left the **Enter text, HTML, PL/SQL, or a URL that will generate the initial page in the target frame** entry field blank.

Frame Drivers: Display Options

Description Use this page to choose options that control the look and feel of the finished frame driver.

This page contains:

Divide Frames by	Choose COLS to divide the web page into vertical (left and right) frames. Choose ROWS to divide the web page in the horizontal (upper and lower) frames.
Border	Type a value that sets the width of the border separating the driving and target frames. A value of 1 produces the thinnest border. Higher number produce thicker borders.
Driving Frame Name	Displays the name of the driving frame. You can use this name or type a new one. Note You need to specify a Driving Frame Name only if you are creating a frame driver that calls another frame driver.
Target Frame Name	Displays the name of the target frame. You can use this name or type a new one. Note You need to specify a Target Frame Name only if you are creating a frame driver that calls another frame driver.
Driving Frame Width and Height	Choose the relative size of the driving frame versus the target frame. For example, if you specified COLS in the Divide Frames by entry field, the web page divides into left and right frames. If you choose 20% in the Driving Frame Width and Height entry field, the web page displays a left driving frame that is one quarter the size of the right target frame. Note To choose an absolute rather than relative frame size, specify point size instead of %; for example, 300 pt.

Frame Drivers

Target Frame Width and Height

Choose the relative size of the driving frame versus the target frame. For example, if you specified COLS in the **Divide Frames by** entry field, the web page divides into left and right frames. If you choose 20% in the **Target Frame Width and Height** entry field, the web page displays a left driving frame that is four times the size of the right target frame.

Note To choose an absolute rather than relative frame size, specify point size instead of %; for example, 300 pt .

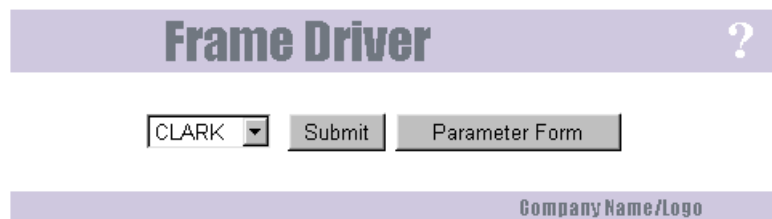
Button Name

Type the name you want to display on the driving frame Submit button.

Note The Submit button appears only if you specified a combo box or radio group **Display LOV As** in the SQL Query for Frame Driver page of this wizard

Button Location

Choose where you want the Submit button to appear in the driving frame. In the figure below, the Button Location is **Right of the LOV**.



Show Parameter Button

Check to display in the driving frame a button that the end user can click to display the frame driver's parameter entry form.

Parameter Button Name

Type the name you want to display on the show parameter button. In the figure above, the **Parm Button Name** is Parameter Form.

Notes

- Driving and target width and height percentages should add to 100%.
- You can specify an asterisk (*) as one frame size if you specify the other frame size in points. For example, you could specify driving frame size of 300 pt and a target frame size of * in a horizontally divided frame driver. This would display a driving frame 300 points high and a target frame whose height would be the difference between the default size of the end user's browser window and 300.

Frame Drivers: Text Options

Description Use this page to choose text that you want to display at the top or bottom of the driving frame or the frame driver's parameter entry form. You can also add help text for the frame driver or its parameter entry form. The end user clicks a ? button on the page to view the text.

This page contains:

Template	Choose a template to set the look and feel of frame driver and parameter entry form elements such as background colors and images, and the image that appears in the upper left corner of the page.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the frame driver or parameter entry form.
Header Text	Type any introductory text that you want to display at the top of the driving frame or parameter entry form, just below the title.
Footer Text	Type any text that you want to display at the bottom of the driving frame or the frame driver parameter entry form.
Help Text	Type any text that you want to display in a help page for the frame driver or parameter entry form. If you type text in this text box, WebDB automatically adds a help button to the frame driver or parameter entry form. End users can click this button to link to a page displaying the help text.

Notes

- You can choose a template in addition to any options you choose in the Frame Drivers: Display Options page. Templates control the look and feel of the page on which the frame driver appears, whereas display options control the look and feel of the frame driver itself.
- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** entry fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

Frame Drivers: Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different points during the execution of the HTML code that creates the driving frame or the frame driver parameter entry form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Type or paste a PL/SQL procedure that will execute before the page containing the driving frame or the frame driver parameter entry form.

Enter the PL/SQL code to execute after displaying the header

Type or paste a PL/SQL procedure that will execute after the driving frame or the frame driver parameter entry form header displays.

Enter the PL/SQL code to execute before displaying the footer

Type or paste a PL/SQL procedure that will execute before the driving frame or the frame driver parameter entry form footer displays.

Enter the PL/SQL code to run after displaying the page

Type or paste a PL/SQL procedure that will run after the page containing the driving frame or the frame driver parameter entry form displays.

Hierarchies

Hierarchy Building

Description Use this page to create a new hierarchy, or find an existing or recently edited hierarchy. After you find a hierarchy, you can edit it using the Edit Hierarchy dialog box.

This page contains:

Create a New Hierarchy

Create

Click to create a new hierarchy.

Find an Existing Hierarchy

Find in Schema

Choose the schema that owns the hierarchy you want to find. The **Find in Schema** drop-down list displays all schemas you have privileges to browse.

Find

Click to search for hierarchies in the schema you specified in the **Find in Schema** drop-down list.

Select a Recently Edited Hierarchy

Name

Displays the names of the five most recently created or edited hierarchies. Click a **Name** to edit the hierarchy.

Schema

Displays the schema that owns the hierarchy.

Type

Displays the component type; i.e. **Hierarchy**.

Changed

Displays in days, hours, minutes, and seconds how long ago the hierarchy was created or last edited.

By

Displays the developer who created or last edited the hierarchy.

Hierarchies: Hierarchy Name and Schema

Description Use this page to choose a name for the hierarchy and the name of the database schema in which the finished hierarchy will be created.

This page contains:

Schema	Choose a schema that will own the database package containing the finished hierarchy. Only schemas that you are allowed to build in are listed in the drop-down list.
Hierarchy Name	Type the name you want to use to identify the database package containing the finished hierarchy; for example, MY_HIERARCHY.

Notes

- The **Schema** becomes part of the URL that end users specify to display the component.
- Follow these guidelines when typing a **Hierarchy Name**:
 - You must specify a hierarchy name (null is not allowed).
 - The name must be unique within the schema.
 - Blank spaces are not allowed. Type an underscore character (_) to add a space in a name. For example, you can name a hierarchy MY_HIERARCHY, but not MY HIERARCHY.
 - You cannot name a hierarchy with a PL/SQL reserved word; for example, COLUMN, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

Hierarchies: Tables or Views

Description Use this page to choose the database table or view on which you want to base the hierarchy. You can choose only one table.

The table or view must have a recursive relationship. For example, the SCOTT.EMP table contains a MGR column foreign key column that is related to the primary key in the same table, the EMPNO column. The MGR and EMPNO columns share a recursive relationship.

This page contains:

Tables/Views

Choose the table or view whose data will be used to create the hierarchy.

The table name in the list is prefixed by the schema that owns it. For example, a table named EMP in the SCOTT schema appears in the list as SCOTT.EMP.

Notes

- The **Table/Views** drop down list contains all tables and views on which you have SELECT, INSERT, UPDATE, or DELETE privileges. Not all of these tables have a recursive relationship. If you are unsure, browse the table to view relationships between its columns.
- You automatically have SELECT, INSERT, UPDATE, or DELETE privileges on a table or view if it is owned by a schema in which you have Build In privileges.

Hierarchies: Table/View Columns

Description Use this page to choose a table or view column for each of the following entry fields on this page:

- **Primary Key Column** - the column you choose must contain values that uniquely identify each row in the table; for example, the EMPNO column of the SCOTT.EMP table.
- **Parent Key Column** - the column you choose must contain values that refer to the primary key; for example, the MGR column of the SCOTT.DEPT table. This column contains values that refer to the EMPNO column of SCOTT.EMP.
- **Start with Column** - this column contains a value that will be used to determine the topmost level in the hierarchy; for example, the value President in the JOB column of the SCOTT.EMP table.
- **Display Column** - this column contains the actual values that will display in the hierarchy; for example, the ENAME column of the SCOTT.EMP table.

A hierarchy based on the above example table columns would display a hierarchy whose topmost level contains the name of the president. The next level would contain the names of employees who report to the president. The next level would contain employees who directly report to these managers, and so on.

This page contains:

Primary Key Column	Choose the table or view's primary key column. The primary key column contains values that uniquely identify each row in the table.
Parent Key Column	Choose the table or view's parent key column. The parent key contains values that refer to the primary key in the same table. Note All parent key values must match an existing primary key value or else be null.
Start with Column	Choose the table or view column whose values will be used to determine the starting point for the hierarchy. After you choose a Start with Column , you can choose a Default Start with Value in this column to

determine which value in the **Start with Column** displays in the topmost level of the hierarchy. For example, you could choose the JOB column of SCOTT.EMP as the **Start with Column** and President as the **Default Start with Value**. This creates a hierarchy that displays on the topmost level data from table rows containing President .

This option is equivalent to specifying a SQL WHERE clause in the statement that selects table or view data to display in the topmost level of the hierarchy.

Default Start with Value

Choose a value in the **Start with Column** that determines which table row data displays in the topmost level of the hierarchy.

For example, you could choose the JOB column of SCOTT.EMP as the **Start with Column** and Manager as the **Default Start with Value**. This creates a hierarchy that displays Manager on the topmost level.

Note **Default Start with Value** is case-sensitive.

Start with LOV

Choose a List of Values that allows end users of the hierarchy to choose a **Default Start with Value** on the hierarchy's parameter entry form.

Display Column Expression

Type a column name whose values you want to display in each box in the hierarchy. In addition to the column name, you can specify text or an expression (for example ENAME | | ' - ' | | JOB).

If you do not specify a column name, values in the column you specified in **Start with Column** display in the hierarchy boxes.

Link

Choose a link form the text you specified in the **Display Column Expression** to another WebDB component or URL.

Hierarchies: Column Conditions

Description Use this page to optionally specify conditions that limit the data displayed in the hierarchy. To specify a condition, choose a **Column Name**, a **Condition**, and a **Value**. For example, to display in the hierarchy data about all employees in Department 10 of the SCOTT.EMP table, choose EMP.DEPTNO from **Column Name**, = from **Condition**, and 10 from **Value**.

You can also use this page to create fields on a parameter entry form and associate a List of Values with each field on the form.

This page contains:

Conditions

Column Name	Choose columns whose values will be used to limit the data displayed in the hierarchy. For example, if you want to display in the hierarchy all employees whose manager's employee number is 7698, choose EMP.MGR as the Column Name. Note After you choose all columns you want to include, make sure % is specified for any unused Column Name rows on this page.
Condition	Choose a condition that selects which column values will be used to display hierarchy data. For example, if you want to display in the hierarchy all employees whose manager's employee number is 7698, choose = as the Condition.
Value	Type a column value that will be used to limit the data displayed in the hierarchy. For example, if you want to display in the hierarchy all employees whose manager's employee number is 7698, type 7698 as the Value.
More Conditions	Click to display more fields on this page that allow you to specify additional conditions for limiting data displayed in the hierarchy.

Note

- To specify multiple values after an IN or NOT IN condition, type a colon (:) between each value. For example, to display in the hierarchy 3 departments from the SCOTT.EMP table, you could choose DEPTNO . EMP as the **Column Name**, IN as the **Condition**, and 10 : 20 : 30 as the **Value**

Hierarchies: Display Options

Description Use this page to choose options that control the appearance of the finished hierarchy.

This page contains:

Run Options

Maximum Levels	Choose the number of vertical levels you want to display in the hierarchy. You can choose a maximum of 3 levels.
Max Child Level Rows	Type the maximum number of table or view rows whose data rows you want to display on the child level of the hierarchy. The child level appears in the hierarchy just below the topmost, or parent, level. Data from each table row or view row you specify displays in a separate box on this level of the hierarchy.
Max Grandchild Level Rows	Type the maximum number of table or view rows whose data you want to display on the grandchild level of the hierarchy. The grandchild level appears in the hierarchy below the parent and child levels. Data from each table row or view row you specify displays in a separate box on this level of the hierarchy.
Show Query Criteria	Check to display on the hierarchy the SQL query used to build it.
Log Activity	Check to log in the WebDB activity log the names of end users who request the hierarchy as well as other performance information.
Show Timing	Check to display at the bottom of the hierarchy the time from when the request to generate the hierarchy was received by the server to when the HTML for the hierarchy was generated.

Look and Feel Options

Parent Type Face	Choose a font for displaying text on the uppermost, or parent, level of the hierarchy.
Parent Font Size	Choose the size of the text displayed on the uppermost, or parent, level of the hierarchy. Specify Font Size as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.
Parent Font Color	Choose the color of text displayed on the uppermost, or parent, level of the hierarchy.
Parent Box BG Color	Choose a color for the boxes that appear in the uppermost, or parent, level of the hierarchy.
Child Type Face	Choose a font for displaying text in the child level, just below the parent level of the hierarchy.
Child Font Size	Choose the size of text displayed in the child level, just below the parent level of the hierarchy.
Child Font Color	Choose the color of text displayed in the child level, just below the parent level of the hierarchy.
Child Box BG Color	Choose a color for the boxes that appear in the child level, just below the parent level of the hierarchy.
Grandchild Type Face	Choose a font for displaying text in the lowest, or grandchild, level of the hierarchy.
Grandchild Font Size	Choose the size of text displayed in the lowest, or grandchild, level of the hierarchy.
Grandchild Font Color	Choose the size of text displayed the lowest, or grandchild, level of the hierarchy.

Hierarchies

Type

Choose a display format for the hierarchy.

- HTML Table - displays the hierarchy in two rows of boxes.
- Break Down - displays the hierarchy as a bulleted list. Indented bullets indicate lower levels on the hierarchy.

Hierarchies: Parameter Entry Form Display Options

Description Use this page to optionally display for each table or view column an entry field on the hierarchy's parameter entry form. The entry field enables end users to choose their own condition for displaying data from the column on the report.

For example, if you choose the JOB column from the SCOTT.EMP table as a **Column Name** on this page, WebDB adds a entry field for the column to the hierarchy's parameter entry form. End users can type a job title to display only data about employees who have that job on the hierarchy.

You can optionally add a List of Values to the entry field. In the previous example, instead of requiring end users to type a job title, you could add a List of Values that enables them to choose Clerk, Analyst, or Manager.

Other options on this page enable you to choose which buttons and options are displayed to the end user of the parameter entry form. For example, you can choose whether to display a Batch button that allows the end user to run the hierarchy in batch mode, or choose a style for displaying the hierarchy.

This page contains:

Value Required

Check to require the end user to specify a value for the column's entry field on the hierarchy's parameter entry form. If you do not check this box, the end user is not required to specify a value.

Column Name

Choose a table or view column. An entry field will be added to the hierarchy's parameter entry form that allows end users to specify values that will used to limit the column's data displayed in the hierarchy.

If you do not choose a table or view column, an entry field for it does not appear on the parameter entry form.

Prompt

Type the prompt text you want to display next to the entry field. The prompt text tells end users what to enter in the field; for example

Hierarchies

	Display all employees who have this manager:
LOV	Type the name of the List of Values you want to use for the column's entry field. End users of the parameter entry form can choose values from this list to limit data displayed in the hierarchy. For example, you could add a List of Values containing departments: 10, 20, 30, etc. Click  to the right of the text box to search for Lists of Values.
Display LOV As	Choose the List of Values format you want to provide the end user for selecting parameters for the column's entry field.
More Parameters	Click to display more fields on this page that allow you to add additional entry fields on the parameter entry form.
Hierarchy Style	Check to enable end users of the parameter entry form to choose a display format for the hierarchy.
Start With Value	Check to enable end users of the parameter entry form to choose a value in the Start With column that will determine the starting point for the hierarchy.
Maximum Level	Check to enable end users of the parameter entry form to choose the number of vertical levels in the hierarchy.
Max Child Level Rows	Check to enable end users of the parameter entry form to choose the maximum number of table or view rows whose data will display on the child level of the hierarchy. The child level appears in the hierarchy just below the topmost, or parent, level. Data from each table row or view row displays in a separate box on this level of the hierarchy.

Max Grandchild Level Rows	Check to enable end users of the parameter entry form to choose the maximum number of table or view rows whose display on the grandchild level of the hierarchy. The grandchild level appears in the hierarchy below the parent and child levels. Data from each table row or view row displays in a separate box on this level of the hierarchy.
Run	Check to display a Run button on the hierarchy's parameter entry form. End users can click the Run button to display the hierarchy with the options they have specified in the parameter entry form.
Save	Check to display a Save button on the hierarchy's parameter entry form. End users can click the Save button to save their option selections.
Reset	Check to display a Reset button on the hierarchy's parameter entry form. End users can click the Reset button to reset all entry fields to their default values.
Name	Type the label you want to display on the button; for example, <code>Display Sales Org Chart</code>. Tip Keep the Name short to avoid displaying large buttons.
Location	Choose the vertical location of the button on the parameter entry form. Note Choose Don't Show if you don't want the button added to the parameter entry form.
Alignment	Choose whether to display the button on the left, center, or right margin of the parameter entry form.

Hierarchies: Parameter options

Description Parameter options allow you to choose which options appear on the parameter entry form for the hierarchy. End users can choose these options when running the hierarchy.

Hierarchy Style	Enables end users to choose a display format for the hierarchy.
Start with Value	Enables end users to choose a value in the Start With column that will determine the starting point for the hierarchy.
Maximum Levels	Enables end users to choose the number of vertical levels in the hierarchy.
Max Child Level Rows	Enables the end user to choose the maximum number of table or view rows whose data will display on the child level of the hierarchy. The child level appears in the hierarchy just below the topmost, or parent, level. Data from each table row or view row displays in a separate box on this level of the hierarchy.
Max Grandchild Level Rows	Allows the end user to choose the maximum number of table or view rows whose display on the grandchild level of the hierarchy. The grandchild level appears in the hierarchy below the parent and child levels. Data from each table row or view row displays in a separate box on this level of the hierarchy.

Hierarchies: Text Options

Description Use this page to choose text that you want to display at the top or bottom of the hierarchy or parameter entry form. You can also add help text for the hierarchy or parameter entry form. The end user clicks a ? button to view the text.

This page contains:

Template	Choose a template to set the look and feel of hierarchy elements such as background colors and images, and the image that appears in the upper left corner of the page.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the hierarchy page or its parameter entry form.
Header Text	Type any introductory text that you want to display at the top of the hierarchy or parameter entry form, just above the title.
Footer Text	Type any text that you want to display at the bottom of the hierarchy or parameter entry form.
Help Text	Type any text that you want to display in a help page for the hierarchy or parameter entry form. If you type text in this text box, WebDB automatically adds a help button to the hierarchy or parameter entry form. End users can click this button to link to a page displaying the help text.

Hierarchies

Notes

- You can choose a template in addition to any options you choose in the Hierarchies: Display Options page. Templates control the look and feel of the page on which the hierarchy appears, whereas display options control the look and feel of the hierarchy itself.
- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** entry fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

Hierarchies Wizard: Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different points during the execution of the HTML code that creates the hierarchy or parameter entry form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Enter the PL/SQL code to execute after displaying the header

Enter the PL/SQL code to execute after displaying the footer

Enter the PL/SQL code to run after displaying the page.

Type or paste a PL/SQL procedure that will execute before the page containing the hierarchy or parameter entry form displays.

Type or paste a PL/SQL procedure that will execute after the hierarchy or parameter entry form header displays.

Type or paste a PL/SQL procedure that will execute after the hierarchy or parameter entry form footer displays.

Type or paste a PL/SQL procedure that will run after the page containing the hierarchy or parameter entry form displays.

Menus

Menu Building

Description Use the Menu Building page to create a new menu, or find an existing or recently edited menu. After you find a menu, you can edit it.

This page contains:

Create a New Menu

Create	Click to create a new menu.
--------	-----------------------------

Find an Existing Menu

Find in Schema	Choose the schema that owns the menu you want to find. The Schema drop-down list displays all schemas you have privileges to browse.
----------------	---

Find	Click to find all menus owned by the schema you specified.
------	--

Select a Recently Edited Menu

Name	Displays the name of the five most recently created or edited menus. Click a Name to edit the menu.
------	--

Schema	Displays the schema that owns the menu.
--------	---

Type	Displays the menu type, i.e., Menu .
------	---

Changed

Displays in days, hours, minutes, and seconds how long ago the menu was created or last edited.

By

Displays the user who created or last edited the menu.

Menus: Menu Name and Schema

Description Use this page to choose a name for the menu and the name of the database schema in which the finished menu will be created.

This page contains:

Schema	Choose the schema that will own the stored procedure containing the finished menu. Only schemas that you are allowed to build in are listed in the list.
Menu Name	Type the name you want to use to identify the stored procedure containing the finished menu; for example, MY_MENU.

Notes

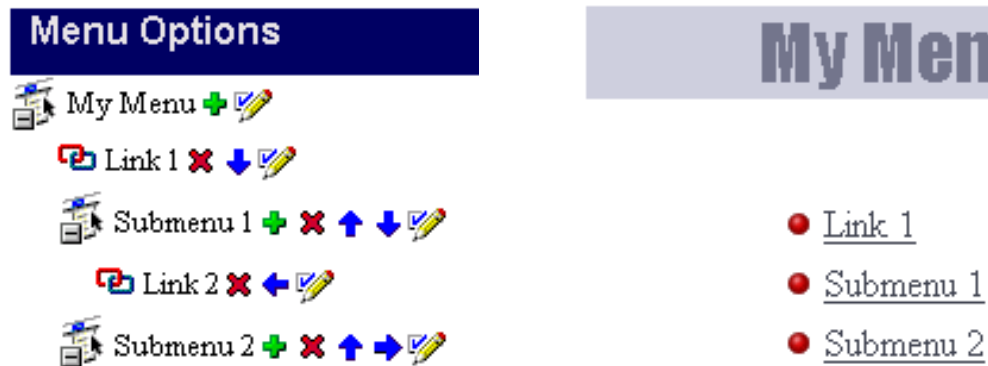
- The **Schema** becomes part of the URL that end users can specify to display the component.
- Follow these guidelines when typing a **Menu Name**:
 - You must specify a name (null is not allowed).
 - The menu name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a menu MY_MENU but not MY MENU.
 - You cannot name a menu with a PL/SQL reserved word; for example, COLUMNS, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

Menus: Menu Options - Menu

Description Use the right frame of this page to specify options that control the appearance and behavior of menus. A menu appears on the topmost level of a menu hierarchy, as shown in the left frame of this page. There is only one menu per hierarchy. A submenu is simply a menu called by another menu or submenu.

Menus and submenus appear in the hierarchy as  icons, whereas links appear as  icons. Links appear on menus and submenus. An end user can click a link to jump to a WebDB component or another URL.

In the hierarchy shown in the left figure below, My Menu is a menu, Submenu 1 and Submenu 2 are submenus, and Link 1 and Link 2 are links. This hierarchy creates a menu similar to that shown to the right of it.



The buttons that appear to the right of each menu, submenu, or link in the hierarchy indicate the actions you can perform on it.

Button

Allows you to:



Insert a submenu below the current menu or submenu, **OR** add a link that appears on the menu or submenu. Clicking this

Menus

button opens a dialog box in which you to choose whether to add the submenu or link.



Delete a submenu or link. **Note** You cannot delete a menu (the top level of the hierarchy).



Change the order in which the submenu or link appears on a menu or submenu.



Move the submenu or link under a new parent in the menu hierarchy.



Edit the menu, submenu, or link.

When you click  next an item in the hierarchy, the right frame displays a different set of options depending on whether you are editing a menu, submenu or link. The options shown below are for editing a menu. You can update these options by typing a new value

in a corresponding entry field. Click  or **Finish** when you have completed all of your updates.

This right frame of this page contains:

Role Security

Choose the role that will be required to display the menu. For example, if you choose DEPT1, only members of the DEPT1 role will be able to access the menu. To allow all end users to view the menu, choose PUBLIC.

Note A DBA creates roles and assigns members to it in the Role and User Managers. If you don't find a role you want to use in the list, contact your DBA.

Name

Type a name that will be displayed at the top of the menu; for example, **My menu**.

Template

Choose a template to set the look and feel of menu elements such as background colors and images, and the image that appears in the upper left corner of the menu.

Hint To set a common look and feel for all menus and submenus in the hierarchy, choose a template for the menu, then choose **Default to Parent Template** for all submenus under the menu.

Preview Template

Click to view the appearance of the template currently selected in the **Template** list.

Sublevels

Choose how many sublevels of the menu hierarchy you want to display on the menu. For example, if you specify 2, the menu in the example hierarchy shown at the top of this help topic might look like this:



- [Link 1](#)
- [Submenu 1](#)
 - [Link 2](#)
 - [Submenu 2](#)

Welcome Text

Type any introductory text that you want to display at the top of the menu below the Name, as shown in the example below.



This is welcome text.

- Link 1
- Submenu 1
- Submenu 2

Find Menu Options

This is footer text.

Elapsed Time: .9 second(s)

Menu Footer	Type any text that you want to display at the bottom of the menu, as shown in the above example.
Show Timing	Click to display at the bottom of the menu the time from when the server received a request to generate the menu to when the HTML for the menu was generated. See the example above for an example.
Log Activity	Click to log in the WebDB activity log the names of end users who access the menu as well as other performance information.
Show Find Option Button	Click to enable end users to search for text on submenus linked to this menu.

Specifying this option adds a button to the current menu labeled **Find Menu Options**. An end user of the menu can type search criteria and click the button to search for text in any menus that are linked from the menu containing the button. Clicking a search result navigates to the menu containing the text. See the example above for an example.

Note

- You can specify HTML in the **Welcome Text** and **Footer Text** fields. For example, to change the color of text to blue, you could specify

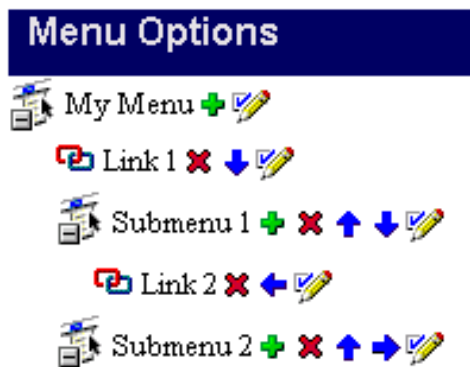
```
<font color="blue">This is blue text.</font>
```

Menus: Menu Options - Submenu

Description Use the right frame of this page to specify options that control the appearance and behavior of submenus. A submenu is simply a menu called by another menu or submenu. A submenu appears as a hypertext link on its parent menu. Links also appear on menus and submenus. An end user can click a link to jump to a WebDB component or another URL.

The left frame of this page displays the current menu hierarchy. Menus and submenus appear as  icons, whereas links appear as  icons. The top level of the hierarchy is called the menu.

In the hierarchy shown in the left figure below, My Menu is a menu, Submenu 1 and Submenu2 are submenus, and Link 1 and Link2 are links. The hierarchy creates a menu similar to that shown to the right of it.



The buttons that appear to the right of each menu, submenu, or link in the hierarchy indicate the actions you can perform on it.

Button	Allows you to:
	Insert a submenu below the current menu or submenu, OR add a link that appears on the menu or submenu. Clicking this icon opens a dialog box in which you to choose whether to add the submenu or link.
	Delete a submenu or link. Note You cannot delete a menu (the top level of the hierarchy).
	Change the order in which the submenu or link appears on a menu or submenu.
	Move the submenu or link under a new parent in the hierarchy.
	Edit the current menu, submenu, or link.

When you click  next an item in the hierarchy, the right frame displays a different set of options depending on whether you are editing a menu, submenu or link. The options shown below are for editing submenus. You can update these options by typing new a

new value in a corresponding entry field. Click  or **Finish** when you have completed all of your updates.

This right frame of this page contains:

Font Face	Choose a font for the hypertext link to the submenu that appears on its parent menu. As shown in the figure below, you can choose a different font for each hypertext link. Note If the font you want to use isn't listed, you can add it to the list using the Font Manager.
-----------	---



	This is welcome text.
	● <u>Link 1</u>
	● <u>Submenu 1</u>
	● <u>Submenu 2</u>
Font Size	Choose the font size for displaying the hypertext link to the submenu that appears on its parent menu. Specify Font Size as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font. As shown in the figure above, you can choose a different font for each hypertext link.
Role Security	Choose the role that will be required to display the submenu. For example, if you choose DEPT1, only members of the DEPT1 role will be able to access the submenu. To allow all end users to view the submenu, choose PUBLIC. Note A DBA creates roles and assigns members to it in the Role and User Managers. If you don't find a role you want to use in the list, contact your DBA.

Name	Type a name for the submenu; for example, Employee Forms . This name will appear in at the top of the submenu. It will also appear as a hypertext link on its parent menu.
Bullet Icon	Choose an image file (.gif format) to use as the bullet next to the hypertext link to the submenu that appears on its parent menu. Click  the right of the text box to search for available bullet icons. Note The icon must be located in the /images/ virtual directory. You can add an icon to this directory using the Image Manager.
Description	Type descriptive text that will display just below the hypertert link to the submenu on its parent menu. For example, if you specify Menu Item 1 as the Name of a submenu and This is descriptive text for Menu Item 1 as the Description , it appears on its parent menu like this:



Menu Item 1

- This is descriptive text for menu item 1.

Template	Choose a template to set the look and feel of submenu elements such as background colors and images, and the image that appears in the upper left corner of the submenu.
----------	--

Menus

Hint To set a common look and feel for all menus and submenus in the hierarchy, choose a template for the menu, then choose **Default to Parent Template** for all submenus under the menu.

Preview Template

Click to view the appearance of the template currently selected in the **Template** list.

Sublevels

Choose how many sublevels of the menu hierarchy you want to display on the menu. For example, if you specify 2, the menu in the example hierarchy shown at the top of this help topic might look like this:



- [Link 1](#)
- [Submenu 1](#)
 - [Link 2](#)
 - [Submenu 2](#)

Help Link

If the **Template** you are using for the submenu contains a help button, you can type a link to a URL containing help text. When an end user click the help button on the menu or submenu, the help text will display.

Welcome Text

Type any introductory text that you want to display at the top of the submenu just below the Name, as shown in the following example.

Submenu 1

This is welcome text

- [Link 1](#)
- [Submenu 2](#)

Back: [My Menu](#)

This is footer text

Elapsed Time: .16 second(s)

Menu Footer

Type any text that you want to display at the bottom of the submenu, as shown in the above example.

Note

- You can specify HTML in the **Welcome Text** and **Footer Text** fields. For example, to change the color of the text to blue, you could specify

```
<font color="blue">This is blue text</font>
```

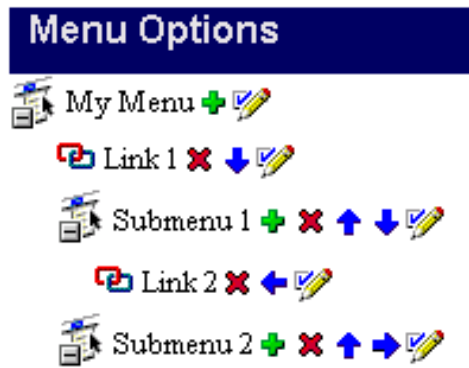
Menus: Menu Options - Links

Description Use this page to add, edit, or delete a link in the menu hierarchy. Links appear on menus and submenus, as shown in the example below. An end user can click a link to jump to a WebDB component or another URL. The URL could be to any Web page, including a WebDB Web site.

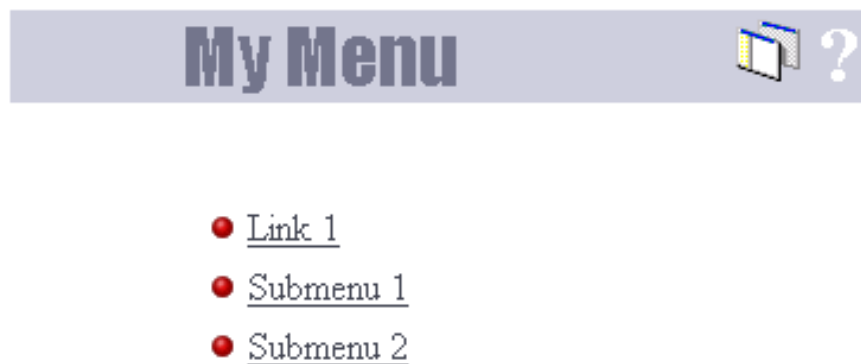
The top level of the hierarchy is called the menu. A submenu is simply a menu called by another parent menu or submenu. As shown below, a menu or submenu can also display a hypertext link to another submenu.



The left frame of the page you are currently looking at displays the current menu hierarchy. Links appear as  icons, whereas menus and submenus appear as  icons.



In the above hierarchy My Menu is a menu, Submenu1 and Submenu2 are submenus, and Link 1 and Link 2 are links. This hierarchy creates a menu similar to the following, and two submenus:



The buttons that appear to the right of each menu, submenu, or link in the hierarchy indicate the actions you can perform on it.

Menus

Button

Allows you to:



Insert a submenu below the current menu or submenu, **OR** add a link that appears on the menu or submenu. Clicking this icon opens a dialog box in which you to choose whether to add the submenu or link.



Delete a submenu or link. **Note** You cannot delete a menu (the top level of the hierarchy).



Change the order in which the submenu or link appears on a menu or submenu.



Move the submenu or link under a new parent in the hierarchy.



Edit the current menu, submenu, or link.

When you click  next an item in the hierarchy, the right frame displays a different set of options depending on whether you are editing a menu, submenu or link. The options shown below are for editing links. You can update these options by typing new a new value in a corresponding entry field. Click **Finish** when you have completed all of your updates.

This right frame of this page contains:

Font Face

Choose a font for displaying the link text; for example, *Arial*. The link text appears on its parent menu or submenu.

As shown in the figure below, you can choose a different font for each hypertext link.

Note If the font you want to use isn't listed, you can add it to the list using the Font Manager.

My Menu



- Link 1
- **Link 2**

Find Menu Options

Font Size

Choose the font size for displaying link text.

Specify **Font Size** as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.

As shown in the figure above, you can choose a different font size for each hypertext link.

Role Security

Choose the role that will be required to view the link. For example, if you choose DEPT1, only members of the DEPT1 role will be able see the link displayed on its parent menu or submenu. To allow all end users to view the link, choose PUBLIC.

Note A DBA creates roles and assigns members to it in the Role and User Managers. If you don't find a role you want to use in the list, contact your DBA

Menus

Name	Type a name for the link ; for example, Link to Employee Chart . The Name appears on the link's parent menu as selectable hypertext.
Bullet Icon	Choose an image file (.gif format) to use as the bullet next to the link on its parent menu or submenu. Click  the right of the text box to search for available bullet icons. Note The icon must be located in the /images/ virtual directory. You can add an icon to this directory using the Image Manager.
Link	- If you want to link to another URL, type its location in the text box; for example, <code>http://www.oracle.com</code> - If you want to link to a WebDB component, you can type the name of the package containing the component; for example, <code>SCHEMA . COMPONENT . SHOW</code> where SCHEMA is the name of the schema that owns the component, COMPONENT is the component name, and SHOW is the procedure used to display the component (you can also specify SHOW_PARDS to display the parameter entry form for the component). OR you can click  next to the text box to choose from a list of links to components that have been defined in the Link Manager.

Description

Type descriptive text that will display just below a link on its parent menu. For example, if you specify **Link to Employee Report** as the **Name** of a submenu and **Click above to view a report of employee names** as the **Description**, it appears on its parent menu like this:



- Link to Employee Report
Click above to view a report of employee names.

Reports

Report Building

Description Use the Report Building page to create a new report, or find an existing or recently edited report. After you find a report, you can edit it using the Edit Report dialog box.

This page contains:

Create a New Report

Report from Query Wizard

Click this radio button to create a new report using the Report Wizard. The wizard guides you through all pages for creating a report, including creating the SQL query that selects the data displayed in the report.

Report from SQL Query

Click this radio button to create a report by writing your own SQL query that selects the data displayed in the report.

Create

Click to create a new report using either the SQL query build wizard or by writing your own SQL query.

Find an Existing Report

Find in Schema

Choose the schema that owns the report you want to find. The **Find in Schema** drop-down list displays all schemas you have privileges to browse.

Find

Click to find all reports owned by the schema you specified.

Select a Recently Edited Report

Name	Displays the name of the five most recently created or edited reports. Click a Name to edit the report.
Schema	Displays the schema that owns the report.
Type	Displays the report type, either: • A report created using the SQL query build wizard • A report created using a hand-coded SQL query
Changed	Displays in days, hours, minutes, and seconds how long ago the report was created or last edited.
By	Displays the name of the developer who created or last edited the report.

Reports: Name and Schema

Description Use this page to choose a name for the report and the name of the database schema in which the finished report will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished report. Only schemas that you are allowed to build in display in the list.
Report Name	Type the name you want to use to identify the database package containing the finished report; for example, MY_REPORT.

Notes

- The **Schema** becomes part of the URL that end users can specify to display the component.
 - Follow these guidelines when typing a **Report Name**:
 - You must specify a report name (null is not allowed).
 - The name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a report MY_REPORT, but not MY REPORT.
 - You cannot name a report with a PL/SQL reserved word; for example, COLUMN, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

Reports: Table and Views

Description Use this page to choose the database tables or views on which you want to base the report.

This page contains:

Tables/Views

Choose the tables or views whose data will be used to create the report. The drop-down list contains all tables and views on which you have SELECT, INSERT, UPDATE, or DELETE privileges.

The table name in the list is prefixed by the schema that owns it. For example, a table named EMP in the SCOTT schema appears in the list as SCOTT.EMP.

The tables SCOTT.EMP and SCOTT.DEPT are selected by default in this list.

Note You can choose more than one table or view. This is equivalent to specifying a join condition in a SQL WHERE clause.

Note

- You automatically have SELECT, INSERT, UPDATE, or DELETE privileges on a table or view if it is owned by a schema in which you have Build In privileges.

Reports: Join Conditions

Description Use this page to specify join conditions between one or more tables or views that you selected in the Tables and Views page of the Create Report Wizard. The columns shown in the two sets of list boxes in this page are contained in the tables or views you selected. For example, a column named JOB from the EMP table appears in the list boxes as EMP . JOB .

This page contains:

Column	Choose the table or view columns that you want to include in the JOIN condition of the SQL query used to build the report. To specify a join condition, choose a column from one of the left Column drop-down lists and a column in the Column drop-down list directly to the right of it.
--------	--

Notes

- WebDB bases the default join conditions shown in this page on the tables or views you selected in previous pages. You can accept these or choose different join conditions.
- Rows in one table may be joined to rows in another if common values exist in corresponding table or view columns.

Reports: Table/View Columns

Description Use this page to choose the table or view columns you want to display in the report. You also specify the order in which columns display.

This page contains:

Columns

Select table or view columns whose data you want to display in the report. Then click > to move a selected column to the **Selected Columns** list box.

Note The columns shown in the two sets of list boxes in this page are contained in the tables or views you selected. For example, a column named JOB from the SCOTT.EMP table appears in the combo boxes as emp . j ob.

Selected Columns

Displays all columns that will appear in the report.

The vertical order of columns in this list box determines their display order from left to right in the report. To change the order, select a column, then click ^ and v to move it up and down in the list box.

To remove a column from this list, select it and click the < button.

Notes

- Click >> and << to move all columns between list boxes.
- Click ^ and v to change the order in which columns display on the report.

Reports: Column Conditions

Description Use this page to optionally specify conditions that limit the data displayed in the report. To specify a condition, choose a **Column Name**, a **Condition**, and a **Value**. For example, to display in the report data about a sales department, choose Department from **Column Name**, like from **Condition**, and Sales from **Value**.

This page contains:

Conditions

Column Name

Choose a column whose values will be used to limit the data displayed in the report.

For example, if you want to display in the report values greater than 3000 from the EMPNO column of the SCOTT.EMP table, choose EMPNO as the **Column Name**.

Note After you choose all columns you want to include, make sure % is specified for any unused Column Name rows on this page.

Condition

Choose a condition that selects which column values will be used to display report data.

For example, if you want to display in the report values greater than 3000 from the EMPNO column of the SCOTT.EMP table, choose greater than as the **Condition**.

Value

Type a column value that will be used to limit the data displayed in the report.

For example, if you want to display in the report values greater than 3000 from the EMPNO column of the SCTT.EMP table, type 3000 in the **Value** text box.

More Conditions

Click to display more fields on this page that allow you to add specify additional conditions for limiting data displayed in the report.

Notes

- To specify multiple values after an IN or NOT IN condition, type a colon (:) between each value. For example, to display in the report 3 departments from the SCOTT.EMP table, you could choose EMP.DEPTNO as the **Column Name**, IN as the **Condition**, and 10 : 20 : 30 as the **Value**.

Reports: Column Formatting

Description Use this page to choose options that control report column formatting on a column-by-column basis.

This page contains:

Column	Displays the names of the table or view columns you selected in the previous page.
Column Heading Text	Type the heading name you want to use to identify the column in the report. For example, instead of displaying the table column name EMPNO in the report, you could specify the more descriptive Employee ID Number as column heading text.
Sum	Check to sum values within the column and display the result in the report. Note This check box is valid only for columns containing numeric data.
Align	Choose whether to align data to the left, center, or right margin of a report column. By default, numeric data align to the left, and alphabetical data align to the right.
Format Mask	Type an Oracle display format for columns containing numeric and date datatypes. For example, you could type DD/MM/YYYY to display dates according to this pattern, or 999,999,999.99 to place commas and decimals according to the pattern. Note Refer to the Oracle documentation for additional information about date and numeric formatting options.

Fixed Size	Type the number of spaces for the column width. Note This option applies only to reports formatted for output as ASCII text.
Link	Choose whether to specify a link from values in the column to another WebDB component or URL. Values in the column will appear as hypertext in the report. Note You can specify a link only if one has been created for this column and stored in the database. If a link does not exist, you can create the report without the link, create a link, then edit the report to include the link you created.

Note

- There is a limit of 255 columns that can be displayed in a report for SELECT statements.

Reports: Display Options

Description Use this page to choose options that control the appearance of the finished report.

This page contains:

Run Options

Maximum Rows	Type the maximum number of rows you want to display in the report.
Show Null Values As	Type the text string you want to display for all null values in the report; for example, (null).
Draw Lines Between Rows	Check to display lines between report rows.
Show Query Conditions	Check to display at the bottom of the report all user-specified parameters passed to the query that created the report, and the time when the report was created.
Paginate	Check to display on the report a button labeled Next. Clicking the button allows the end user to see more report rows. The maximum number of report rows that the end user can see is set by the Maximum Rows option.
Log Activity	Check to log in the WebDB activity log the names of users who request the report as well as other performance information.

Show Timing

Check to display at the bottom of the report the time from when the server received the request to generate the report to when the HTML for the report was generated.

Default Format

Choose a display format for the report:

- **HTML Format**

Formats the report using HTML tables and displays output on a new page in the web browser. Components that contain large amounts of data may take longer to display in this format.

- **Microsoft Excel**

Downloads the report for display in Microsoft Excel.

- **ASCII text**

Formats the report using the HTML PRE tag to display heading and values in the report as ASCII text. This option is useful for displaying large amounts of data.

Border

Choose **Thin Border** or **Thick Border** to add a border around the report. Choose **No Border** if you don't want a border.

Break Options

Break Style

Choose a style for breaking the report on the columns you choose in the **First Break Column**, **Second Break Column**, or **Third Break Column** drop-down lists.

Reports

The **Left Break** style begins breaking the report from its left margin. The **First Break Column** displays as the far left column, the **Second Break Column** is second to the left margin and so on.

First Break Column

Breaks the report using the column you specify.

Second Break Column

Breaks the report using the column you specify, after first breaking it on the column you specified in the **First Break Column** field.

Third Break Column

Breaks the report using the column you specify, after first breaking it on the column you specified in the **First Break Column** and **Second Break Column** fields.

Look and Feel Options

Type Face

Choose a font for displaying report text.

Font Size

Choose the size of the font for displaying report text.

Specify **Font Size** as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.

Font Color

Choose the color for displaying report text.

Heading Background Color

Choose a background color for column headings in the report.

Table Row Color(s)

Choose a color for table rows in the report. You can choose multiple colors.

Row Order Options

Order by

Choose the table or view column whose values will be used to sort rows in the report. Choosing this option is equivalent to specifying a SQL ORDER BY clause.

Choose **Ascending** to sort query results according to the alphabetic (A-Z) or numeric (starting with the lowest number) order of column values, depending on the datatype for the column. Choose **Descending** to sort in the reverse order.

then by

Choose additional columns whose values will be used to sort report rows. For example, if you choose Order by Department ID, then by Employee Name, WebDB sorts report rows numerically using department IDs. Rows containing the same Department ID are then sorted alphabetically using employee names.

Reports: Parameter Entry Form Display Options

Description Use this page to optionally display for each table or view column an entry field on the report's parameter entry form. The entry field enables end users to choose their own condition for displaying data from the column on the report. For example, if you choose the DEPTNO column from the SCOTT.EMP table as a **Column Name** on this page, WebDB adds a entry field for the column to the report's parameter entry form. End users can type a department number in the field to display only data about employees from that department on the report.

You can optionally add a List of Values to the entry field. In the previous example, instead of requiring end users to type a numeric value, you could add a List of Values that enables them to choose 10, 20, or 30.

Other options on this page enable you to choose which buttons and options are displayed to the end user of the parameter entry form. For example, you can choose whether to display a Batch button that allows the end user to run the report in batch mode, or options for displaying the report output.

This page contains:

Value Required

Check to require the end user to specify a value for the column's entry field on the report's parameter entry form. If you do not check this box, the end user is not required to specify a value.

Column Name

Choose a table or view column. An entry field will be added to the report's parameter entry form that allows end users to specify values that limit the column's data displayed in the report.

If you do not choose a table or view column, an entry field for it does not appear on the parameter entry form.

Prompt	Type the prompt text you want to display next to the entry field. The prompt text tells end users what to enter in the field; for example Display all employees in this department:
LOV	Type the name of the List of Values you want to use for the column's entry field. End users of the parameter entry form can choose values from this list to limit data displayed in the report. For example, you could add a List of Values containing ranges of department numbers: 10, 20, 30, etc. Click  to the right of the text box to search for Lists of Values.
Display LOV As	Choose the List of Values format you want to provide the end user for selecting parameters for the column's entry field.
More Parameters	Click to display more fields on this page that allow you to add additional entry fields on the parameter entry form.
Output Format	Check to enable end users of the parameter entry form to choose a display format for the report: HTML Format Formats the report using HTML tables and displays output on a new page in the web browser. Microsoft Excel Downloads the report for display in Microsoft Excel.

Reports

- **ASCII text**

Formats the report using the HTML PRE tag to display heading and values in the report as ASCII text.

Maximum Rows/Page

Check to enable end users of the parameter entry form to specify the maximum number of table or view rows to display in the report.

Break Columns

Check to enable end users of the parameter entry form to specify which table or view columns to break the report on.

Font Size

Check to enable end users of the parameter entry form to choose the size of the font for displaying report text.

Order By

Check to enable end users of the parameter entry form to choose table or view columns whose values will be used to sort rows in the report, and the sort order.

Run

Check to display a Run button on the report's parameter entry form. End users can click the Run button to display the report with the options they have specified in the parameter entry form.

Save

Check to display a Save button on the report's parameter entry form. End users can click the Save button to save their option selections.

Batch

Check to display a Batch button on the report's parameter entry form. End users can click the Batch button to run the report in batch mode and save the results in the database.

Reset	Check to display a Reset button on the report's parameter entry form. End users can click the Reset button to reset all entry fields to their default values.
Name	Type the label you want to display on the button; for example, <code>Display Sales Report</code> . Tip Keep the Name short to avoid displaying large buttons.
Location	Choose the vertical location of the button on the parameter entry form. Note Choose Don't Show if you don't want the button added to the parameter entry form.
Alignment	Choose whether to display the button on the left, center, or right margin of the parameter entry form.

Reports: Text Options

Description Use this page to specify text that you want to display at the top or bottom of the report page or parameter entry form. You can also add help text about the report or parameter entry form. The end user clicks a button with a question mark (?) to view the help text.

This page contains:

Template	Choose a template to set the look and feel of page elements such as background colors and images, and the image that appears in the upper left corner of the page. The template sets the appearance of both the report and parameter entry form.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the report page.
Header Text	Type any introductory text that you want to display at the top of the report or parameter entry form. If you type this text in the Report text box, it appears on the report page below the title of the report or parameter entry form.
Footer Text	Type any text that you want to display at the bottom of the report or parameter entry form.
Help Text	Type any text that you want to display in a help page for the report or parameter entry form.

If you type text in this text box, WebDB automatically adds a help button to the report or parameter entry form. End users can click this button to link to a page displaying the help text.

Notes

- You can choose a template in addition to any options you choose in the Reports: Display Options page. Templates control the look and feel of the page on which the report appears, whereas display options control the look and feel of the report itself.
- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

Reports: Advanced PL/SQL Code

Description Use this page to specify PL/SQL code that runs at different points during the execution of the HTML code that creates the report or parameter entry form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Type or paste a PL/SQL procedure that will execute before the page containing the report or parameter entry form displays.

Enter the PL/SQL code to execute before displaying the header

Type or paste a PL/SQL procedure that will execute before the report or parameter entry form header displays.

Enter the PL/SQL code to execute before displaying the footer

Type or paste a PL/SQL procedure that will execute before the report or parameter entry form footer displays.

Enter the PL/SQL code to run after displaying the page

Type or paste a PL/SQL procedure that will run after the report or parameter entry form displays.

Advanced PL/SQL code examples

Specifying the following in the Reports **Add Advanced PL/SQL Code** wizard page produces the report shown below. This example demonstrates the locations in the finished report where the executed PL/SQL code displays.

	Report	Parameter Entry Form
... before displaying the page.	<code>htp.print('<H1>This text displays before displaying the page </H1>');</code>	
... after displaying the header.	<code>htp.print('<H1>This text displays after displaying the header </H1>');</code>	
... before displaying the footer.	<code>htp.print('<H1>This text displays before displaying the footer </H1>');</code>	
... after displaying the page.	<code>htp.print('<H1>This text displays after the page displays</H1>');</code>	

Example reports formatted with column breaks

The following report, based on the SCOTT.EMP table, was formatted with EMP . DEPTNO specified for the **First Break Column** option on the Display Options page.

Deptno	Empno	Ename	Job	Mgr	Hiredate	Sal	Comm
10	7782	CLARK	MANAGER	7839	09-JUN-81	2450	(null)
	7839	KING	PRESIDENT	(null)	17-NOV-81	6000	(null)
	1111	A	CLERK	(null)	01-JAN-99	1000	(null)
	5555	E	ANALYST	(null)	10-MAR-98	1234	(null)
	7934	MILLER	CLERK	7782	23-JAN-82	1300	(null)
	1234	AA	(null)	(null)	10-SEP-98	2000	(null)
sum						13984	0
20	7369	SMITH	CLERK	7902	17-DEC-80	800	(null)
	7876	ADAMS	CLERK	7788	12-JAN-83	1100	(null)
	7902	FORD	ANALYST	7566	03-DEC-81	3000	(null)
	7788	SCOTT	ANALYST	7566	09-DEC-82	3000	(null)
	7566	JONES	MANAGER	7839	02-APR-81	2975	(null)
	4444	D	ANALYST	(null)	02-JAN-98	1000	(null)
	3333	C	SALESMAN	(null)	01-FEB-99	1500	(null)
sum						13375	0
30	7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300
	7698	BLAKE	MANAGER	7839	01-MAY-81	2850	(null)

The same report, formatted with EMP . DEPTNO specified for the **First Break Column** option and EMP . JOB specified for the **Second Break Column** option.

Deptno	Job	Empno	Ename	Mgr	Hiredate	Sal	Comm
10	MANAGER	7782	CLARK	7839	09-JUN-81	2450	(null)
	sum					2450	0
	PRESIDENT	7839	KING	(null)	17-NOV-81	6000	(null)
	sum					6000	0
	CLERK	1111	A	(null)	01-JAN-99	1000	(null)
	sum					1000	0
	ANALYST	5555	E	(null)	10-MAR-98	1234	(null)
	sum					1234	0
	CLERK	7934	MILLER	7782	23-JAN-82	1300	(null)
	sum					1300	0
	(null)	1234	AA	(null)	10-SEP-98	2000	(null)
	sum					2000	0
sum						13984	0
20	CLERK	7369	SMITH	7902	17-DEC-80	800	(null)
		7876	ADAMS	7788	12-JAN-83	1100	(null)
	sum					1900	0
	ANALYST	7902	FORD	7566	03-DEC-81	3000	(null)
		7788	SCOTT	7566	08-DEC-82	3000	(null)

SQL-based Reports

SQL-based Reports: Report Name and Schema

Description Use this page to choose a name for the report and the name of the database schema in which the finished report will be created.

This page contains:

Schema	Choose the schema that will own the database package containing the finished report. Only schemas that you are allowed to build in are listed in the drop-down list.
Report Name	Type the name you want to use to identify the database package containing the finished report; for example, MY_REPORT.

Notes

- The **Schema** becomes part of the URL that end users specify to display the component.
- Follow these guidelines when typing a **Report Name**:
 - You must specify a report name (null is not allowed).
 - The name must be unique within the schema.
 - Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a report MY_REPORT, but not MY REPORT.
 - You cannot name a report with a PL/SQL reserved word; for example, COLUMN, PACKAGE, VARCHAR. Refer to the Oracle documentation for more information about reserved words.

SQL-based Reports: SQL Statement

Description Use this page to type or paste the SQL statement that selects the table or view data to display in the report.

This page contains:

Enter the SQL SELECT
Statement

Type or paste a SQL SELECT statement.

Notes

Use alphanumeric strings preceded by colons as bind variables (for example, `:var1`, `:var2`, `:var3...`). All bind variables must **begin** with an alpha character (a, b, c, etc.). For each bind variable you specify, WebDB displays a parameter entry field in the report's parameter entry form. The entry field prompts end users to choose conditions for displaying data in the report.

- Do not add single or double quotes around bind variables. For example, do not specify `' :var1 '` or `" :var1 "` as a bind variable.
- Do not insert a colon in the bind variable itself. For example, the report fails if your bind variable is:

```
select * from WEBDB.test_time
where time_stanp = :timevar
```

- If you specify a bind variable in this page, you can associate a List of Values with it in the Reports: Parameter Entry Fields page.
- Preface table names with their owning schema (for example, `SCOTT.EMP` for a table owned by `SCOTT`) if your statement includes tables owned by a schema other than the one that will own the finished report.
- You can include relative HTML links by coding them into the `SELECT` list.

SQL-based Reports

- Do not include an ORDER BY clause if you plan to specify one in the Reports: Display Options page.
- Use aliases for long columns names (greater than 32 characters).

SQL-based Reports: Column Formatting

Description Use this page to choose options that control report column formatting on a column-by-column basis.

This page contains:

Column	Displays the table or view columns you specified in the SQL Query page.
Column Heading Text	Type the heading name you want to use to identify the column in the report. For example, instead of displaying the table column named EMPNO in the report, you could specify the more descriptive <code>Employee ID Number</code> as column heading text.
Sum	Check to sum values within the column and display the results in the report. Note This option is valid only for columns containing numeric data.
Align	Choose whether to align data to the left, center, or right margin of a report column. By default, numeric data align to the left, and alphabetical data align to the right.
Format Mask	Type an Oracle display format for columns containing numeric and date datatypes. For example, you could type <code>DD/MM/YYYY</code> to display dates according to this pattern, or <code>999,999,999.99</code> to place commas and decimals according to the pattern. Note Refer to the Oracle documentation for additional information about date and numeric formatting options.

SQL-based Reports

Fixed Size

Type how many spaces wide you want the column to appear in the report.

Note This option applies only to reports formatted for output as ASCII-style text.

Note

- There is a limit of 255 columns that can be displayed in a report for SELECT statements.

SQL-based Reports: Display Options

Description Use this page to choose options that control the appearance of the finished report.

This page contains:

Run Options

Maximum Rows	Type the maximum number of rows you want to display in the report.
Show Null Value As	Type the text string you want to display for all null values in the report; for example, (null).
Draw Lines Between Rows	Check to display lines between report rows.
Show Query Conditions	Check to display at the bottom of the report all user-specified parameters passed to the query that created the report, and the time when the report was created.
Paginate	Check to display on the report a button labeled Next. Clicking the button allows the end user to see additional report rows. The maximum number of rows that the end user sees on a page is set by the Maximum Rows option.
Log Activity	Check to log in the WebDB activity log the names of end users who request the report as well as other performance information.
Show Timing	Check to display at the bottom of the report the time from when the request to generate the report was received by the server to when the

HTML for the report was generated.

Default Format

Choose a display format for the report:

- **HTML Format**

Formats the report using HTML tables and displays output on a new page in the web browser. Components that contain large amounts of data may take longer to display in this format.

- **Microsoft Excel**

Downloads the report for display in Microsoft Excel.

- **ASCII text**

Formats the report using the HTML PRE tag to display heading and values in the report as ASCII text. This option is useful for displaying large amounts of data.

Border

Choose **Thin Border** or **Thick Border** to add a border around the report. Choose **No Border** if you don't want a border.

Break Options

Break Style

Choose a style for breaking the report on the columns you choose in the **First Break Column**, **Second Break Column**, or **Third Break Column** drop-down lists.

Left Break style begins breaking the report from its left margin. The **First Break Column** displays as the left-most column, the **Second Break Column** is second to the left margin and so on.

First Break Column

Breaks the report using the column you specify.

Second Break Column Breaks the report using the column you specify after first breaking it on the column you specified in the **First Break Column** field.

Third Break Column Breaks the report using the column you specify after first breaking it on the column you specified in the **First Break Column** and **Second Break Column** fields.

Look and Feel Options

Type Face Choose a font for displaying report text.

Font Size Choose the size of the font for displaying report text.

Specify **Font Size** as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.

Font Color Choose the color of font for displaying report text.

Heading Background Color Choose a background color for column headings in the report.

Table Row Color(s) Choose a for report rows. You can choose multiple colors.

Row Order Options

Order by Choose the table or view column whose values will be used to sort rows in the report. Choosing this option is equivalent to specifying a SQL ORDER BY clause.

SQL-based Reports

Choose **Ascending** to sort query results according to the alphabetic (A-Z) or numeric (starting with the lowest number) order of column values, depending on the datatype for the column. Choose **Descending** to sort in the reverse order.

then by

Choose additional columns whose values will be used to sort report rows. For example, if you choose **Order by** Department ID, **then by** Employee, WebDB sorts report rows numerically using department IDs. Rows containing the same Department ID are then sorted alphabetically using employee names.

SQL-based Reports: Parameter Entry Form Display Options

Description Use this page to optionally display entry fields in the report’s parameter entry form. WebDB displays a parameter entry field for each table or view column for which you specified a bind variable on the SQL-based Reports: SQL Statement page. The entry field enables end users to choose conditions for displaying data in the report.

For example, if you specified a bind variable for the DEPTNO column from the SCOTT.EMP table, WebDB adds an entry field for the column to the report’s parameter entry form. End users can type a department number in the field to display only data about employees from that department on the report.

You can optionally add a List of Values to the entry field. In the previous example, instead of requiring end users to type a numeric value, you could add a List of Values that enables them to choose 10, 20, or 30.

Other options on this page enable you to choose which buttons and options are displayed to the end user of the parameter entry form. For example, you can choose whether to display a Batch button that allows the end user to run the report in batch mode, or options for displaying the report output.

This page contains:

Bind Variable	Displays each bind variable you specified on the SQL-based Reports: SQL Statement page.
Prompt	Type the prompt text you want to display next to the entry field. The prompt text tells end users what to enter in the field; for example Display all employees in this department:

SQL-based Reports

LOV

Type the name of the List of Values you want to use for the column's entry field. End users of

the parameter entry form can choose values from this list to limit data displayed in the report.

For example, you could add a List of Values containing ranges of department numbers: 10, 20, 30, etc.

Click  to the right of the text box to search for Lists of Values.

Display LOV As

Choose the List of Values format you want to provide the end user for selecting parameters for the column's entry field.

Output Format

Check to enable end users of the parameter entry form to choose a display format for the report:

- **HTML Format**

Formats the report using HTML tables and displays output on a new page in the web browser.

- **Microsoft Excel**

Downloads the report for display in Microsoft Excel.

- **ASCII text**

Formats the report using the HTML PRE tag to display heading and values in the report as ASCII text.

Maximum Rows/Page	Check to enable end users of the parameter entry form to specify the maximum number of bars to display in the report.
Break Columns	Check to enable end users of the parameter entry form to specify which table or view columns to break the report on.
Font Size	Check to enable end users of the parameter entry form to choose the size of the font for displaying report text.
Order By	Check to enable end users of the parameter entry form to choose table or view columns whose values that will be used to sort rows in the report, and the sort order.
Run	Check to display a Run button on the report's parameter entry form. End users can click the Run button to display the report with the options they have specified in the parameter entry form.
Save	Check to display a Save button on the report's parameter entry form. End users can click the Save button to save their option selections.
Batch	Check to display a Batch button on the report's parameter entry form. End users can click the Batch button to run the report in batch mode and save the results in the database.
Reset	Check to display a Reset button on the report's parameter entry form. End users can click the Reset button to reset all entry fields to their default values.
Name	Type the label you want to display on the button; for example, Run Sales Report.

SQL-based Reports

Tip Keep the **Name** short to avoid displaying large buttons.

Location

Choose the vertical location of the button on the parameter entry form.

Note Choose **Don't Show** if you don't want the button added to the parameter entry form.

Alignment

Choose whether to display the button on the left, center, or right margin of the parameter entry form.

SQL-based Reports: Text Options

Description Use this page to choose text that you want to display at the top or bottom of the report page or parameter entry form. You can also add help text. End users can click a ? button on the report or parameter entry form to view the help text.

This page contains:

Template	Choose a template to set the look and feel of report and parameter entry form elements such as background colors and images, and the image that appears in the upper left corner of the page.
Preview Template	Click to view the appearance of the template currently selected in the Template drop-down list.
Title	Type text you want to display in the banner at the top of the report or parameter entry form.
Header Text	Type any introductory text that you want to display below the report or parameter entry form title.
Footer Text	Type any text that you want to display at the bottom of the report or parameter entry form.
Help Text	Type any text that you want to display in a help page for the report or parameter entry form. If you type text in this text box, WebDB automatically adds a help button to the report or parameter entry form. End users can click this button to link to a page displaying the help text.

Notes

- You can choose a template in addition to any options you choose in the Reports: Display Options page. Templates control the look and feel of the page on which the report appears, whereas display options control the look and feel of the report itself.
- You can specify HTML in the **Title**, **Header Text**, **Footer Text**, and **Help Text** fields. For example, to change the color for the text in any of these fields, you could specify

```
<font color="blue">this is text</font>
```

SQL-based Reports: Advanced PL/SQL Code

Description Use this page to choose PL/SQL code that runs at different points during the execution of the HTML code that creates the report or parameter entry form.

This page contains:

Enter the PL/SQL code to execute before displaying the page

Type or paste a PL/SQL procedure that will execute before the page containing the report or parameter entry form displays.

Enter the PL/SQL code to execute before displaying the header

Type or paste a PL/SQL procedure that will execute before the report or parameter entry form header displays.

Enter the PL/SQL code to execute before displaying the footer

Type or paste a PL/SQL procedure that will execute before the report or parameter entry form footer displays.

Enter the PL/SQL code to run after displaying the report

Type or paste a PL/SQL procedure that will run after the report or parameter entry form displays.

Manage Components

Manage Component

Description Use the Manage Component page to perform actions on existing WebDB components; for example, executing, editing, copying, dropping, or viewing information about the component. The actions you can perform on the component depend on your privileges.

The name of the component on which you can perform these actions appears in the upper left corner of the page.

This page contains:

(Component Type and Name)	Displays the component's type and name; for example: <code>Form (table) my_form</code> for a form based on a table called my_form.
Version(s) Status	Displays the all versions of the component and the current status of each version. Click a status to edit the component version. Note If there are no hyperlinks, you do not have privileges to edit the component.
Last Changed	Displays the name of the user who last created or edited the component, and the date and time when the component was created or last edited.
Run Link	Displays the URL for the procedure or procedures that, when executed, display the

component. You can copy and paste this URL into another web page to create a link to the component.

Note A procedure that executes the component without parameters has the suffix `.show`. A procedure that executes the component with parameters has the suffix `.show_parms`.

Edit	Click to edit the most recent version of the component. For example, you can reselect any table columns on which the component is based, change any fields or text that appear in the component, or choose a new look and feel.
Run	Click to run the current PRODUCTION version of the component. Note If a valid package for the component doesn't exist, the component will not run.
Parameters	Click to display the parameter entry form for the component. The parameter entry form enables you to specify values that will be used to display the component. Note If the current component is a form, Browse appears instead of Parameters on this page.
Privileges	Click to grant another WebDB user privileges to run the component.
Monitor	Click to view a chart of all requests for the component and the users who made the request.
Manage	Click to display additional options for managing the component such as copying, exporting or dropping the component from the database.

Manage Component Privileges

Description Use the Manage Component Privileges page to grant or revoke privileges that enable other WebDB users or roles to execute a component you own. To revoke a privilege, uncheck the check box next to the **Existing Grants: Grantee** name, and click **Revoke**.

This page contains:

Grant Additional Privileges

Schema	Displays the schema that owns the component to which you are granting execute privileges.
Component	Displays the name of the component to which you are granting execute privileges.
User/Role	Type the name of the user or role who you want to allow to execute the component. Click  to the right of the text box to search for users and roles.
Grant Execute Privilege	Click to grant execute privileges on the component to the user or role you specify in the User/Role text box. The user or role is added to the list of existing grants.

Existing Grants

Grantee	Displays the WebDB users and roles who currently have privileges to run the component displayed in the Component field. Check next to the Grantee name and click Revoke to revoke the user or role's privilege to execute the privilege.
Privilege	Displays current privileges for the WebDB user or role.

Manage Component Privileges

Revoke

Click to revoke privileges from one or more selected Grantees. A privilege is revoked if you check the check box next to the Grantee's name and click **Revoke**.

Return to component
management

Click to return to the Manage Component page.

Manage Component

Description Use the Manage Component page to perform actions on existing components.

This page contains:

About	Click to view component attributes and other information about the component, including all versions of the component, version owners, and locking information. Component attributes are the options specified by the user during the creation or most recent edit of the component. For example, the name of the table on which the component is based, column and condition parameters, and look and feel options.
Export	Click to export the component to a remote database. A page displays the component source code. To export the component, cut and paste the text into a file and save it using the Save feature in your web browser.
Copy	Click to create a copy of the component.
Rename	Click to change the name of the component.
Generate	Click to generate the source code for the component.
Drop	Click to drop one or more component versions from the database. A page displays asking you to confirm which versions to drop.
Enter/Exit Debug Mode	Click toggle debug mode on and off. If you click Enter Debug Mode, the SQL code used to create the component displays on the same page

when you run the component. If you click **Exit Debug Mode**, the component displays without the code.

Package Spec

Click to view the specification for the package that, when executed, displays the component.

The package spec contains the list of functions, procedures, variables, constants, cursors, and exceptions contained within the package.

Package Body

Click to view the body of the package that, when executed, displays the component.

The package body contains the PL/SQL code implementing the specification.

Show Call Interface

Click to view the component call interface.

Show Locks on this Component

Click to view a report of developers who currently have locks on the component. Users with the DBA role can make selections on this report to unlock the component.

WebDB automatically locks a components whenever a developer edits a component. The lock prevents other developers from accessing the component while it is being edited.

Return to Component page

Click to return to the Manage Component page.

Copy Component

Description Use the Copy Component page to create a copy of an existing WebDB component. You can copy the component into a different schema, or copy it into its current schema using a different name.

This page contains:

Current Schema	Displays the name of the schema that owns the component you are copying.
Current Component Name	Displays the name of the component you are copying.
New Schema	Type the name of the schema where you want to locate the copy of the component. This schema will own the copy.
New Component Name	Type the name you want to give to the copy of the component. Note If you are copying the component to the same schema where it is currently located, the Current Component Name must differ from the New Component Name .
Copy	Click to create a copy of the component.
Return to Manage Component Page	Click to return to the Manage Component page.

Note

- To insert a space in a **New Component Name**, type an underscore (_) character.

Rename Component

Description Use the Rename Component page to give a new name to an existing WebDB component.

This page contains:

Current Component Name	Displays the current name of the component.
New Component Name	Type the new name you want to give to the component.
Rename	Click to rename the component with the name you specified in the New Component Name field.
Return to Component Page	Click to return to the Manage Component page.

Notes

- To insert a space in a **New Component Name**, type an underscore (_) character.
- After you rename a component, all requests for the component must use the new name.

Drop Component

Description Use the Drop Component page to drop a component version from the Oracle database. The page displays all versions of the component that you can to drop.

This page contains:

Select the component versions to drop check boxes	Check next to each component version you want to drop from the database.
Yes	Click to drop the component versions you selected from the database.
No	Click to leave the component versions you selected in the database and return to the Component Manager page.
Return to Component Type Manager	Click to return to the Component Manager page.

Note

- Only a component owner (the user whose schema owns the component) and users with the DBA role can drop a component.

Build Shared Components

Manage Colors

Description Use this page to maintain a list of color names and associated values, called color definitions. You can add a new color definition or edit an existing one. The colors you define on this page are used in fonts, page backgrounds, and other elements of WebDB sites and components.

This page contains:

Add Color Name

Color Name

Type a unique name to identify the color. Make sure that the color you add does not already exist. This field is case-sensitive. For example, if you already added Black, you cannot add the color names BLACK or black.

Color Value

Type a color value. You can specify any color value supported by your Web browser, or its hexadecimal equivalent; for example, Blue or #C0D9D9 for Light Blue.

Note Hexadecimal values must be prefixed with the # character and is limited to 6 digits.

Add Color

Click to add the color definition to the **Edit Color Definition: Name** list.

Tip If you specify an invalid **Color Value**, the color definition is added to the list, with a color **Value** of black.

Edit Color Definition

Name

Displays all current color definitions. Click a name to edit the color definition.

Tip To delete a color, click the color name to display the Edit Color page.

Drop Component

Value	Displays all current color definition values.
Test Color	Displays the color.

Edit Color Definition

Description Use this page to change a color name, color value, or to delete a color definition.

This page contains:

Color Name

Displays the name of the color definition you selected. To change it, type a new name and click **Apply Changes**. Make sure that this color name does not already exist. For example, if you already added Black, you cannot add the color names BLACK or black.

Tip To delete a color definition, leave the **Color Name** text box blank and click **Apply Changes**.

Color Value

Displays the color value of the color definition you selected. To change it, type a new value and click **Apply Changes**.

You can specify any color value supported by your web browser, or its hexadecimal equivalent; for example, Blue or #C0D9D9 for Light Blue.

Note Hexadecimal values must be prefixed with the # character and is limited to 6 digits.

Cancel

Click to not apply any recent changes.

Apply Changes

Click to update the color definition based on the color name and value you specified, or delete the color definition.

Manage Images

Description Use this page to maintain a list of images that are used in WebDB components and structured U/I templates. To add an image to the list, place it in a directory mapped to the WebDB virtual directory `/images/`, then assign it a name and an image type. The image name, type, and the file name containing the image comprise the image definition.

Use this page to create a new image definition, or find and edit an existing one. You have the option of searching for images by name, file name, image type or any combination of these criteria.

This page contains:

Create a New Image Name

Image Name

Type a unique name to identify the image; for example, `Department Logo`. This field is case sensitive. Make sure that the image name does not already exist. For example, if you already added `Logo`, you cannot add the images with the names `LOGO` or `logo`.

Note The image name can be the same as the file name.

File Name

Type the name and extension of the file containing the image; for example, `logo.gif`. The image must be located in a directory mapped to the WebDB virtual directory `/images/`.

Image Type

Choose a type for the image. Other users can search for the image using this type

Add Image

Click to add the image definition to the WebDB image list.

Find an Existing Image Definition

Image Name Contains

Type the name or part of the name of the image you want to find.

File Name Contains	Type the file name or part of the file name of the image you want to find.
File Type	Choose the type of image you want to find.
Find Images	Click to search for image definitions based on the criteria you specified. Note: For long lists, up to 30 images per page are returned at a time.
Select a Recently Changed Image	
Name	Displays the names of the five most recently created or edited image definitions. Click a Name to edit the image definition. Tip To delete an image definition, click the Name to display the Edit Image page
File	Displays the name of the file containing the image.
Type	Displays the image type (specified by the image definition creator using the Create a New Image Name: Image Type list on this page).
Last Changed	Displays the date the image definition was created or last updated.
By	Displays the name of the user who created or last changed the image definition.
Image	Displays a thumbnail view of the image.

Edit Image Definition

Description Use this page to update or delete an image definition. The image name, type, and the file name containing the image comprise the image definition.

This page contains:

Image Name	Displays the name of the image definition. To change it, type a new image name and click Apply Changes . Make sure that the image name you add does not already exist. For example, if you already added Logo, you cannot add the images with the names LOGO or logo. This field is case-sensitive. Tip To delete an image definition, leave the Image Name text box blank, then click Apply Changes .
File Name	Displays the file name containing the image. To change it, type a new file name, then click Apply Changes .
Image Type	Displays the image type (specified by the image definition creator using the Create a New Image Name: Image Type list on the Manage Images page).
Apply Changes	Click to update the image definition.

Manage Fonts

Description Use this page to maintain a list of font names and HTML fonts, called a font definition, for use in building WebDB components. Use this page to create a new font definition or edit an existing one.

This page contains:

Create a New Font Definition

Font Definition Name

Type a unique name of the font you want to add to the list: for example, `Times New Roman`. This field is case-sensitive. Make sure this font name does not already exist.

In order to be displayed by the end user's web browser, the **Font Definition Name** must match the name of a font supported by the web browser. If you specify a font that is not supported, the web browser will display its default font.

You can specify alternative fonts by separating them with commas in the **Font Definition Name** text box; for example, `Times New Roman, Times`. In this example, if the user's web browser doesn't support Times New Roman, it will display text using the Times font. If the Times font is not supported, the web browser will display its default font.

Add

Click to add the font definition to the **Edit Font Definition: Font Definition** list.

Edit Font Definition

Font Definition

Displays all currently defined font names. Click a name to edit the font. This field is case-sensitive. Make sure this font name does not already exist.

Note To delete a font from this list, click its name to display the Edit Font Definition page.

Test Font

Displays the font's appearance.

Edit Font Definition

Description Use this page to change a font name.

This page contains:

Font Name	Displays the name of the font definition. To change it, type a new font name and click Apply Changes . This field is case-sensitive. Make sure this font name does not already exist. To delete a font definition, leave the Font Name text box blank and click Apply Changes .
Apply Changes	Click to update the Font Name , or delete the font definition.

Manage U/I Templates

Description Use this page to manage user interface templates that control the appearance of the page on which a component appears. These look and feel elements include background colors and images, and the image that appears in the upper left corner of the page. You can create two types of U/I templates:

- Structured U/I templates display the same image and text in every component that uses the template. For example, if a structured style template contains a company logo and introductory text, the same company logo and text display in a chart, a report, or any other component that uses it.
- Unstructured U/I templates are created using HTML code that creates a web page. After you specify this code, you add substitution tags to the HTML code. These tags embed WebDB components into the web page. For example, you can add a #BODY# tag that adds a WebDB component such as a chart or report to the original web page background.

Use this page to create a new U/I template, or find and edit an existing one. You can search for a U/I template by name, by its type, or using both of these search criteria.

This page contains:

Create a U/I Template

Owning Schema	Choose the name of the schema that will own the finished U/I template. Only schemas in which you are allowed to build are listed. If you want to build in another schema, contact your DBA to obtain Build In privileges for that schema.
Structured	Click to create a U/I template that displays the same image and text in every component to which it applies.
Unstructured	Click to create a U/I template based on HTML source code and embedded substitution tags.
Create Template	Click to create the U/I template.

Drop Component

Edit U/I Template

Owning Schema	Displays the schema that owns the U/I template.
Name	Displays the names of all U/I templates. Click a name to edit a template.
Type	Displays the type of U/I template, either structured or unstructured.
Action	Click Export to display the SQL source code for the U/I template. You can use this code to export the U/I template to another database.

Create/Define Structured U/I Template

Description Use this page to create/define, edit, test, or delete a structured U/I template.

The Create Structured U/I Template page is divided into two frames. The top frame contains options for specifying the images, text, links, and look and feel of the U/I template you are currently creating. The bottom frame enables you to view the appearance of the template you are creating. After specifying an option in the top frame, you can click **Test** to update the appearance of the template with the new option in the bottom frame.

This page contains:

Return to Manage Template page	Click to display the Manage Template page.
Test	Click to view how the U/I template appears based on the options you have specified in the top frame of this page. The updated U/I template appears in the bottom frame.
Save	Click to save the U/I template based on the options you specified in the top frame of this page.
Delete	Click to delete the template displayed in the Template Owner and Name: Name field. Note The Delete button displays on this page only if you are editing an existing U/I template. It does not appear if you are creating a new U/I template.
Reset	Click to clear or reset all entry fields in the top frame of this page to their default values.
Template Owner and Name	
Owner	Choose the schema that will own the finished template. Note To make the U/I template available to others, place it in a schema in which other

Drop Component

WebDB developers have been granted privileges to browse. For example, the DBA at your site might give to all WebDB users privileges to browse in a schema named `TEMPLATES`.

Name

Choose a name to identify the template.

Since other component developers will be using your template, you should give the template a descriptive name that identifies its function; for example, `COMPANY_LOGO_BLUE` for a template that includes the company logo on a blue banner.

Application Image Attributes:

Name

Type the name and extension of the file containing the image you want to display in the upper left corner of the template.

Click  to the right of the text box to search for images. You can search for all images located in the WebDB virtual directory, `/images/`.

If you don't choose an image name, none will display in this location on the template.

Link

Type a link to another URL. When an end user clicks the image you specify in the **Application Image Attributes: Image** combo box, the URL destination you specify in the **Application Image Attributes: Link** text box displays.

ALT Tag

Type the text you want to associate with the image's ALT tag in the HTML code that creates the page. The ALT tag displays a small pop-up (or "floatover") window when an end user places the mouse cursor over the image. ALT tags are useful for displaying text that describes the image.

Home Link:

URL

Type a link to another URL. When an end user clicks the image you specify in the **Home Link: Image** combo box, the URL destination you

	specify in the Home Link: URL text box displays.
Image	Type the name and extension of the file containing the image you want to display in the upper right corner of the template. If you don't choose an image name, none will display in this location on the template.
Help Image:	
Image	Type the name and extension of the file containing the image you want to display as a help icon on the template. If you don't choose an image name, none will display in this location on the template.
Template Title:	
Font Size	Type a font size for displaying title text. The title appears to the right of the application image in a banner at the top of the page. Specify Font Size as a relative size (+1, +2, and so forth). The relative font size is the number specified plus the size of the last font specified in the HTML code for the page; for example, 14 pt and a relative size of +2 displays the title as a 16 pt font.
Font Color	Choose the color for the title text. Click  to the right of the text box to search for colors.
Font Face	Choose a font style for displaying title text.
Heading Background:	
Color	Choose the color for the heading background. The heading background appears behind the application image, title, home image, and help image. Click  to the right of the text box to search for colors.

Drop Component

Image

Type the name and extension of the file containing the image you want to display as the template heading background.

Click  to the right of the text box to search for images. You can search for all images located in the directory mapped to the WebDB virtual directory, /images/.

Note If you specify both a **Heading Background: Color** and **Image**, the heading background will be the color of the image.

Background Color and Image:

Color

Choose a color for the background. The background appears behind the component body.

Click  to the right of the text box to search for colors.

Image

Type the name and extension of the file containing the image you want to display in the background.

Click  to the right of the text box to search for images. You can search for all images located in the directory mapped to the WebDB virtual directory, /images/.

Note If you specify both a **Background: Color** and **Image**, the background will be the color of the image.

Create Unstructured UI Template

Description Use this page to create, edit, test, or delete a U/I template based on HTML code that you write or copy from another source.

The Create Unstructured U/I Template page is divided into two frames. The top frame contains options for specifying the images, text, links, and look and feel of the U/I template you are currently creating. The bottom frame enables you to view the appearance of the template you are creating. After specifying an option in the top frame, you can click **Test** to update the appearance of the template with the new option in the bottom frame.

This page contains:

Return to Manage Template Page	Click to return to the Manage U/I Templates page. Note Any changes you have made on this page will be lost unless you click the Save button before clicking Return to Manage Template Page .
Test	Click to view how the U/I template appears based on the options you have specified in the top frame of this page. The updated U/I template appears in the bottom frame.
Save	Click to save the U/I template based on the options you have specified in the top frame of this page.
Delete	Click to delete the template displayed in the Owner field. Note The Delete button displays on this page only if you are editing an existing U/I template. It does not appear if you are creating a new U/I template.
Reset	Click to clear or reset all fields in the top frame of this page to their default values.
Enter HTML code that creates a web page:	Type or paste HTML source code that creates a Web page. Then, embed substitution tags in the

Drop Component

HTML code to add a title, a heading, a WebDB component, help links, or other elements. The elements are dynamically generated when the HTML code to create the web page runs.

For example, you can add a `#BODY#` tag that adds a component such as a chart or report to the original web page background. When a component developer selects the unstructured template while building the component, the Build wizard uses the substitution variables in the template to create the component. The Chart Build wizard, for example, substitutes a finished chart for the `#BODY#` tag in the template, the home page link that the developer selected in the wizard for the `#HOME#` tag, and so forth.

Manage Links

Description Use this page to search for existing links or create new ones. A link enables an end user to jump from a WebDB component to another component, component parameter entry form, or URL.

This page contains:

Create a New Link

Create Link

Click to create a link using a wizard.

Find an Existing Link

Link Name Contains

Type the name or consecutive characters in the name of the link you want to find.

Find Link

Click to search for links based on the criteria you specified in **Link Name Contains**.

Select a Recently Edited Link

Link Name

Displays the names of the five most recently created or edited links. Click a **Name** to edit the link.

Last Changed

Displays the date the link was created or last updated.

Test Link

Click to test the link. Click **Test Link** to display:

- The link, including the arguments passed to it.
- An HTML anchor. An anchor <A> tag indicates the beginning and end of a hypertext link.
- An example SQL query using the anchor.

Drop Component

- A list of link column values. Click a column value to test the link. The component defined as the link destination specified in the **Identify component to be linked to: Component** entry field of the **Edit Link** page should display.

Click **Export** to display code that you can copy to export the link to another database.

Manage

Click **Delete** to delete the link from the database.

Links: Link Name and Schema

Description Use this page to choose a name for the link and the name of the Component Schema that will own it.

This page contains:

Schema	Choose the schema that will own the finished link. Only schemas in which you have Build In privileges are listed. If you want to build in a schema not in the list, contact your DBA to obtain privileges.
Link Name	Type a name for the link. WebDB developers choose from these names when asked by a build wizard to choose a link to include in the component.

Notes

Follow these guidelines when typing a **Link Name**:

- You must specify a link name (null is not allowed). Type a descriptive name that identifies the link's function; for example, `LINK_TO_MYREPORT` for a link from the EMP table to a report
- The name must be unique within the schema.
- Blank characters are not allowed. Type an underscore character to add a space in a name. For example, you can name a link `LINK_TO_MYREPORT`, but not `LINK TO MYREPORT`.
- You cannot name a link with a PL/SQL reserved word; for example, `COLUMN`, `PACKAGE`, `VARCHAR`. Refer to the Oracle documentation for more information about reserved words.

Links: Link Target Type and Name

Description Use this page to specify the name of the link's target and the type of target.

This page contains:

The link is pointing to a:

WebDB Component

Click if you want the target of the link to be a WebDB component such as a chart or report.

After clicking this option, you must also specify the schema and name of the component in the text box labeled **The name of the component or page you are linking to**; for example, SCOTT.MY_CHART.

WebDB Component
Parameter Form

Click if you want the target of the link to be the parameter entry form for a WebDB component.

After clicking this option, you must also specify the schema and name of the component in the text box labeled **The name of the component or page you are linking to**; for example, SCOTT.MY_CHART.

HTML Link

Click if you want the target of the link to a web page.

After clicking this option, you must also specify the URL of the web page in the text box labeled **The name of the component or page you are linking to**; for example,
http://webdb.mycompany.com

The name of the component
or page you are linking to

If you clicked **WebDB Component** or **WebDB Component Parameter Form**, type the name of the component: prefixed with the schema that owns it; for example, SCOTT.MY_CHART. You can click  to search for all components located in schemas in which you have privileges to browse.

If you clicked **HTML Link**, type the URL of the web page that will be linked to; for example:
http://webdb.mycompany.com

Note

- If you enter an invalid component name or URL in the text box labeled **The name of the component or page you are linking to**, you will get an error when you try to use or test the link.

Links: Link Target Inputs

Description Use this page to specify parameters that will be passed by the link to the target component. For example, you can create a link from a chart displaying average salaries for each JOB in the SCOTT.EMP table to a report based on SCOTT.EMP table that lists employees and their salaries.

Before you build a link to a target component, you must ensure the target component accepts parameters. You can pass as parameters literal values or values in the table or view columns on which the source component is based. In the above example, you could specify that values in the JOB column of the SCOTT.EMP table be passed to the report. When an end user of the chart clicks on a job in the chart, the report displays data for employees corresponding to that job type.

You can specify conditions on the parameter that will be used to limit the data displayed in the component. For example, you can create a link to a report based on the SCOTT.EMP table and pass to it the parameter JOB=CLERK. When an end user clicks the link, the report will always display data for clerks.

You also use this page to choose display options that control the appearance of the target component; for example, the maximum number of rows that can appear on a page of the report. The options correspond to options on the Display Options page of the wizard used to build the target component. Options for a report can include show_header (enables you to show an HTML for the report when displaying it) or max_rows (enables you to specify the maximum number of rows displayed on each page of the report).

This page contains:

Parameters

Parameter

Displays the component's parameters, if the WebDB Developer who created the target component specified columns that can accept parameters (either with a bind variable or using an option in the build wizard).

Parameters map to the columns of the table or view on which the target component is based. For example, if you chose to link to a component based on the SCOTT.EMP table, you might see ENAME, EMPNO, and DEPTNO included in the parameters on this page.

Required?	Displays whether the end user is required to specify a parameter on the parameter entry form. This option was set by the component developer in the component build wizard.
Condition	Choose a condition that selects which parameter values will be used to display report data. For example, if you want to link to a report based on SCOTT.EMP that displays all employees in a particular department, specify = as a Condition, Co lumn as the Value Type, and DEPTNO as the Value.
Value Type	Choose Literal if you want to specify a literal value that will be passed to the component . A literal is an explicit numeric, character, string, or Boolean value not represented by an identifier. You can specify any alphanumeric string; for example, 10 or Manager . If you want to pass a literal value of greater than 3000, specify > as the Condition, 3000 as the Value, and Literal as the Value Type. Choose Column if you want to pass as parameters the values in a column in the table or view (shown in the Parameter list) on which the component is based. Note: Make sure that you set the correct Value Type to correspond to the Value you specify. For example, if you type ENAME as the Value, and you want it to pass as a literal, make sure that you set the Value Type to Literal. Otherwise, the value will pass as parameters in a Column.
Value	Type a column value that will be used to limit the data displayed in the report. For example, if you want the link to be a report showing all employees in the SCOTT.EMP table whose ID numbers are greater than 3000, specify the following for the DEPTNO parameter: > as a Condition, 3000 as the Value, and Literal as the Value Type.
Options	
Required?	Displays whether the end user is required to specify an option on the parameter entry form. This option was

Drop Component

	set by the component developer in the component build wizard.
Option	Displays options that control the appearance of the component that is being linked to. These options correspond to options on the Display Options page of the build wizard that created the component. For example, if the Maximum Rows option was selected when the report being linked to was built, the option <code>max_rows</code> appears in the Option list.
Value Type	Choose Literal if you want to specify a literal value for the option. For example, if the option is Show Header, you could specify a Value Type of <code>Literal</code> and type <code>YES</code> or <code>NO</code> as the Value. Or, if <code>font_size</code> displays as an Option, you could specify a Value Type of <code>Literal</code> and type <code>+1</code>, <code>+2</code>, etc. as the Value. Choose Column if you want to use a value in a table or view as the Value Type.
Value	Displays the current value for the option. For example, if <code>Maximum Rows=20</code> was selected when a report being link to was built, 20 displays in the Value text box. When an end user click the link to navigate to the component, this value will be used to display the component. If the link is to a parameter entry form, the value will appear as a default. You can accept the current Value, or type a new one.

Notes

- This page displays only if you selected a component on **The name of the component or page you are linking to** entry on the Link Name page.
- Do not insert a colon in the bind variable itself. For example, the report fails if your bind variable is:

```
select * from WEBDB.test_time
where time_stanp = :timevar
```


Links: Finish Link

Use the Finish page to create a new link or update an existing link based on your selections in the previous pages of this wizard. If you are satisfied with the choices you made in the wizard, click **OK** to store the link as a shared component. Other WebDB developers can access your link when building components such as charts, forms, and reports.

If you aren't satisfied with your choices, use the  button to navigate back to any wizard page where you want to make changes.

After you make changes, click  to return to the Finish page. Always use these buttons to navigate to and from previous wizard pages. Don't use your web browser's Back and Forward buttons.

After you click **OK**, the Manage Link page displays the name of the link you just created or edited. You can test, edit, export, or delete the link using options on the Manage Link page.

Manage List of Values

Description Use this page to manage Lists of Values (LOVs) that enable end users to choose values for entry fields on WebDB forms and parameter entry forms. You can create two types of Lists of Values:

- Dynamic - based on a SQL statement
- Static - based on hardcoded values in a static list

Use this page to create a new List of Values or find and edit an existing one. You can search for a List of Values by name, by type (dynamic or static), or both.

This page contains:

Create a List of Values

Dynamic	Click to create a new List of Values based on a SQL query.
Static	Click to create a List of Values based on a hard-coded list.
Create LOV	Click to create a new dynamic or static List of Values.

Find an Existing List of Values

LOV Name Contains	Type the name or part of the name of the LOV you want to find.
LOV Type	Choose a List of Values type (either dynamic or static) that will be used to search for List of Values. Choose All Types to search for both static and dynamic Lists of Values.
Find LOV	Click to search for Lists of Values.

Select a Recently Changed List of Values

Owning Schema	Displays the schema that owns the List of Values.
---------------	---

Name	Displays the five most recently created or edited Lists of Values. Click the name to edit the List of Values.
Type	Displays the List of Values type, either STATIC (based on hardcoded values in a static list) or DYNAMIC (based on a SQL statement).
Test As	Click a display format to test the List of Values. For example, if you click Combo, a page appears displaying the List of Values as a combo box. Clicking Radio displays the List of Values as a radio button group. Click Export to display the SQL source code for the List of Values. You use this source code to export the List of Values to another database.

Create Dynamic List of Values

Description Use this page to create a dynamic List of Values (LOV) by specifying:

- The name of the object you want to associate with the List of Values; for example, a table, view, or stored procedure.
- The name of a table or view column or stored procedure attribute. Any WebDB component based on this column or attribute can use the List of Values. The List of Values enables end users to select values in fields in the component or its parameter entry form. These user-selected values query or modify values in the column or attribute corresponding to the entry field.
- A format for displaying the List of Values in the component or its parameter entry form; for example, a pop-up list, or group of check boxes.
- A SQL SELECT statement query that identifies the values displayed to the end user in the List of Values and the actual values passed to the component.

This page contains:

Create this List of Values:

Owning Schema

Choose the name of the schema that will own the finished List of Values.

Notes Only schemas in which you are allowed to build are listed. If you want to build in another schema, contact your DBA to obtain Build In privileges for that schema.

To make the List of Values available to others WebDB Developers, place it in a schema which these developers have been granted browse privileges. For example, the DBA at your site might give to all WebDB developers privileges to browse in a schema named LOVS.

Name

Type the name you want to use to identify the List of Values; for example, **DEPTNO** for a List of Values for a Department Number entry field.

Associate with this Object:

Owner	Choose the schema that owns the table, view, or stored procedure that you want to associate with the List of Values.
Object Name	Type the table, view, or stored procedure you want to associate with the List of Values.
Column or Attribute	Type the name of the column or attribute you want to associate with the List of Values. WebDB will create the List of Values based on all unique values in this column or attribute. End users will be able to query or update the column or attribute using the List of Values. For example, you could create a List of Values and associate it with the JOB column of the SCOTT.EMP table. End users could choose from a List of Values containing five values (Clerk, Salesman, Analyst, Manager, and President) to perform queries based on the JOB column.

Default Display Format:

Default Format	Choose a default format for displaying the List of Values on a component or its parameter entry form. Component developers who add the List of Values to their component have the option of overriding this default and displaying a different format.
Show Null Value	Choose Yes to display null values in the List of Values. If you choose No, null values do not display in the List of Values. Choose % to display null values as %.
Enter SQL	Type or paste a valid SQL SELECT statement that selects values from two table or view columns: The first column specifies each value displayed to the end user in the List of Values.

Manage List of Values

- The second column specifies the actual values passed to the component.

For example, the query
`select ename, empno from
scott.emp`

creates a List of Values that displays to the users employee names from the ENAME column (SMITH, ALLEN, WARD, etc.). It passes the employee's associated ID number (7369, 7499, 7521, etc.) to the component when an end user chooses a name from the List of Values.

Note You can use the same column to generate display values and actual values. For example, you can specify:

```
select ename, ename from  
scott.emp
```

or

```
select ename from scott.emp
```

Add LOV

Click to create the List of Values.

Note

- You must qualify table names in your SQL SELECT query with the owning schema. For example:

<code>select ename from scott.emp</code>	is valid.
<code>select ename from emp</code>	is not valid.

Edit Dynamic List of Values

Description Use this page to update or delete an existing dynamic List of Values.

This page contains:

Edit this List of Values:

Owning Schema

Choose the name of the schema that will own the finished List of Values.

Only schemas in which you are allowed to build are listed. If you want to build in another schema, contact your DBA to obtain Build In privileges for that schema.

Name

Type the name you want to use to identify the List of Values; for example, DEPTNO for a List of Values for a Department Number entry field.

Tip To delete the List of Values, leave the **Name** text box blank and click **Apply Changes**.

Associate with this Object:

Owner

Choose the schema that owns the table, view, or procedure that you want to associate with the List of Values.

Object Name

Type the table, view, or procedure you want to associate with the List of Values.

Column or Attribute

Type the name of the column or attribute you want to associate with the List of Values.

WebDB will create the List of Values based on all unique values in this column or attribute.

End users will be able to query or update the column or attribute using the List of Values. For example, you could create a List of Values List of Values and associate it with the JOB column of the SCOTT.EMP table. End users could choose from a List of Values

containing five values (Clerk, Salesman, Analyst, Manager, and President) to perform queries based on the JOB column.

Default Display Format:

Default Format

Choose a default format for displaying the List of Values on a component or its parameter entry form. Component developers who add the List of Values to their component have the option of overriding this default and displaying a different format.

Show Null Value

Choose **Yes** to display null values in the List of Values.

If you choose **No**, null values do not display in the List of Values.

Choose % to display null values as %.

Enter SQL Query

Type or paste a valid SQL SELECT statement that selects values from two table or view columns:

- The first column specifies each value displayed to the end user in the List of Values.
- The second column specifies the actual values passed to the component.

For example, the query

```
select ename, empno from  
scott.emp
```

creates a List of Values that displays to the users employee names from the ENAME column (SMITH, ALLEN, WARD, etc.). It passes the employee's associated ID number (7369, 7499, 7521, etc.) to the component when an end user chooses a name from the List of Values.

Note You can use the same column to generate display values and actual values. For example, you can specify:

Manage List of Values

```
select ename, ename from  
scott.emp
```

or

```
select ename from scott.emp
```

Apply Changes

Click to update or delete the List of Values.

Create Static List of Values

Description Use this page to create a List of Values based on a static hardcoded list.

This page contains:

Create this List of Values:

Owning Schema

Choose the name of the schema that will own the finished List of Values.

Notes Only schemas in which you are allowed to build are listed. If you want to build in another schema, contact your DBA to obtain Build In privileges for that schema.

To make the List of Values available to others WebDB developers, place it in a schema which these developers have been granted browse privileges. For example, the DBA at your site might give to all WebDB developers privileges to browse in a schema named LOVS.

Name

Type the name you want to use to identify the List of Values; for example, DEPTNO for a List of Values for a Department Number entry field.

Associate with this Object:

Owner

Choose the schema that owns the table, view, or stored procedure that you want to associate with the List of Values.

Object Name

Choose the table, view, or stored procedure that you want to associate with the List of Values.

Column/Argument

Choose the table or view column, or procedure argument that you want to associate with the List of Values.

Default Display Formats:

Default Format

Choose a default format for displaying the List of Values in a component or its parameter entry form. Component developers who add the List of Values to their component have the option of overriding this default and displaying a different format.

Show Null Value

Choose **Yes** to display null values in the List of Values.

If you choose **No**, null values do not display in the List of Values.

Choose **%** to display null values as %.

Display Value

Type each value you want to display in the List of Values in a separate Display Value text box; for example, **Clerk**, **Manager**, and **President** in a Job Title List of Values

Return Value

Type the actual value that will be passed to the component when the end user choose a display value in the List of Values. For example, the List of Values could pass the value **1** to the component when the end user selects the display value Clerk, **2** for Manager, and **3** for President.

Display values can match the values passed to the component; for example, **Clerk** for the Clerk display value.

Display Order

Type numbers beginning with **1** to specify the order in which the display values should appear in the List of Values. For example, to display Clerk, Manager, and President in this order from the top of the list to bottom, specify **1** for Clerk, **2** for Manager, and **3** for President.

To display values in the reverse order, specify **3** for Clerk, **2** for Manager, and **1** for President

Add LOV

Click to create the List of Values.

Edit Static List of Values

Description Use this page to update or delete a List of Values based on a static hardcoded list.

This page contains:

Edit this List of Values:

Owning Schema

Choose the name of the schema that will own the finished List of Values.

Notes Only schemas in which you are allowed to build are listed. If you want to build in another schema, contact your DBA to obtain Build In privileges for that schema.

To make the List of Values available to others, place it in a schema which other WebDB developers have been granted privileges to browse in. For example, the DBA at your site might give to all WebDB developers privileges to browse in a schema named LOVS.

Name

Type the name you want to use to identify the List of Values; for example, DEPTNO for a List of Values for a Department Number entry field.

Tip To delete the List of Values, leave the **Name** text box blank and click **Apply Changes**.

Associate with this Object:

Owner

Choose the schema that owns the table, view, or stored procedure that you want to associate with the List of Values.

Object Name

Choose the table, view, or stored procedure that you want to associate with the List of Values.

Column/Argument

Choose the table or view column, or procedure argument that you want to associate with the List of Values.

Default Display Formats:

Default Format	Choose a default format for displaying the List of Values in a component or its parameter entry form. Component developers who add the List of Values to their component have the option of overriding this default and displaying a different format. For example, instead of displaying the LOV as a drop-down list, a component developer could choose to display a radio button group.
Show Null Value	Choose Yes to display null values in the List of Values. If you choose No, null values do not display in the List of Values. Choose % to display null values as %.
Display Value	Type each value you want to display in the List of Values in a separate Display Value text box; for example, Clerk , Manager , and President in a Job Title List of Values
Return Value	Type the actual value that will be passed to the component when the end user chooses a display value in the List of Values. For example, the List of Values could pass the value 1 to the component when the end user selects the display value Clerk, 2 for Manager, and 3 for President. Display values can match the values passed to the component; for example, Clerk for the Clerk display value.
Display Order	Type numbers beginning with 1 to specify the order in which the display values should appear in the List of Values. For example, to display Clerk, Manager, and President in this order from the top of the list to bottom, specify 1 for Clerk, '2' for Manager, and '3' for President. To display values in the reverse order, specify '3' for Clerk, '2' for Manager, and '1' for President
Apply Changes	Click to update or delete the List of Values.

Manage JavaScripts

Description Use this page to manage JavaScript applications that perform validation on individual entry fields within WebDB components. Field-level validation is performed when the end user causes the OnBlur condition to occur after entering a value in a field; for example, when tabbing to another entry field. Form-level validation occurs after the end user enters a value in a field and submits all values on the page; for example, when clicking an **OK** button.

Use this page to create a JavaScript application, or find and edit an existing one. You can search for JavaScripts by name, by the schema that owns them, or both.

See "Guidelines for writing JavaScripts" for information on other JavaScript functions supported by WebDB.

This page contains:

Create a New JavaScript

Create JavaScript	Click to create a new field- or form-level JavaScript application.
-------------------	--

Find an Existing JavaScript

Owning Schema	Type the name of the schema that owns the JavaScript you want to find.
---------------	--

JavaScript Name Like	Type the name or part of the name of the JavaScript you want to find; for example, IsNumber for a JavaScript that validates that values in a field are numeric.
----------------------	--

Find JavaScripts	Click to search JavaScripts based on the criteria you specified in Owning Schema or JavaScript Name Like .
------------------	--

Select a Recently Edited JavaScript

Schema	Displays the schema that owns the JavaScript application.
--------	---

Name	Displays the name of the five most recently created or edited JavaScripts. Click the name to edit the JavaScript.
------	---

Note: System JavaScripts cannot be edited. For example, you cannot edit the `inRange0_100`, `isNull`, and `isNumber` JavaScripts in the WebDB schema.

Last Changed

Displays the date the JavaScript was created or last updated.

By

Displays the name of the WebDB user who created or last edited the JavaScript.

Test Script

Click **Field** to test a field-level JavaScript validation application. Click **Form** to test a form-level JavaScript validation application.

Create JavaScript

Description Use this page to create a new field- or form-level JavaScript application.

This page contains:

Owning Schema	Choose the name of the schema that will own the finished JavaScript. Note Only schemas in which you are allowed to build are listed. If you want to build in another schema, contact your DBA to obtain Build In privileges for that schema. To make the JavaScript available to other WebDB developers, place it in a schema in which these developers have been granted browse privileges. For example, the DBA at your site might give all WebDB developers privileges to browse in a schema named JAVASCRIPTS.
JavaScript Name	Type the name you want to use to identify the JavaScript; for example, <code>NotNull</code> for a JavaScript that checks for null values in a field.
Language	Type the name of the language in which you plan to write the JavaScript; for example, <code>JavaScript1.1</code> or <code>JavaScript1.2</code> .
Enter Script	Type or paste the JavaScript code that performs field- or form-level validation.
Add JavaScript	Click to create the JavaScript in the schema specified in the Owning Schema entry field.

Note

- See "Guidelines for writing JavaScripts" for information on other JavaScript functions supported by WebDB.

Guidelines for writing JavaScripts

Follow these guidelines when writing a field- or form-validation JavaScript application:

- All validation applications should be written as functions, and return either TRUE or FALSE values.
- The application should alert the end user with a message if the element (entry field) being validated contains an invalid value.
- The application should bring focus (position the cursor) to the element (the entry field) where the end user entered the incorrect value flagged by the JavaScript application.

The following JavaScript validates that the end user enters a numeric value into a field:

- Identifies the name of the function and the entry field being validated. Checks whether the absolute value of the entry field is a number. `isNaN` ("Is Not a Number") is a JavaScript function.
- If the value in the entry field is not a number, a message alerts the end user: "Value must be a number."
- Brings focus to the entry field.

```
function isNumber(theElement)
{
  if ( isNaN( Math.abs(theElement.value) ) )
  {
    alert( "Value must be a number." );
    theElement.focus();
    return false;
  }
  return true;
}
```

Other Supported JavaScript Functions

Other system JavaScript functions provided with WebDB that validate entry fields in components and parameter entry forms include:

Manage JavaScripts

JavaScript function	Description
isDate	Validates that the entered date is in the correct format: DD-MON-YY.
isInteger	Validates that the entered value is an integer value. For example, -120, 20 is accepted; 20.50 is not.
isAlphaNumeric	Validates that the entered value is alphanumeric (alphabetical letters and numbers are accepted). For example, a-z, A-Z, 0-9.
isAlpha	Validates that the entered value is an alphabetical letter. For example, a-z or A-Z.
isEmailAddress	Validates that the entered e-mail address has the @ symbol included. This function does not validate if the e-mail address exists.

Note

- System JavaScripts cannot be edited. For example, you cannot edit the inRange0_100, isNull, and isNumber JavaScripts in the WebDB schema.

Edit JavaScript

Description Use this page to edit an a new field- or form-level JavaScript.

This page contains:

Owning Schema

Choose the name of the schema that will own the finished JavaScript.

Notes Only schemas in which you are allowed to build are listed. If you want to build in another schema, contact your DBA to obtain Build In privileges for that schema.

To make the JavaScript available to others, place it in a schema which other WebDB developers have been granted privileges to browse in. For example, the DBA at your site might give to all WebDB developers privileges to browse in a schema named LOVS.

JavaScript Name

Type the name you want to use to identify the JavaScript; for example, `NotNull` for a JavaScript that checks for null values in a field.

Note: System JavaScripts cannot be edited. For example, you cannot edit the `inRange0_100`, `isNull`, and `isNumber` JavaScripts in the WebDB schema.

Tip To delete a JavaScript, leave the JavaScript Name text box blank and click **Apply Changes**.

Language

Type the name of the language in which you plan to write the JavaScript; for example, `JavaScript1.1` or `JavaScript1.2`.

Enter Script

Type or paste the JavaScript code that performs field- or form-level validation.

Apply Changes

Click to update or delete the JavaScript.

Note

- See "Guidelines for writing JavaScripts" for information on other JavaScript functions supported by WebDB.

Test JavaScript

Description Use this page to test a JavaScript validation application.

This page contains:

Owning Schema	Displays the schema that owns the JavaScript application you are testing.
JavaScript Name	Displays the name of the JavaScript you are testing.
Validation Style	Displays whether the JavaScript you are testing is a field-level or form-level validation application.
Text Field	Type the value you are validating into the first entry field on the page. To test a field-level JavaScript application, you must cause the OnBlur condition to occur; for example, by pressing the Enter key or tabbing to the second text field. After you cause the condition to occur, the JavaScript field-level validation application should run. Form-level applications should run after you type an invalid value in the entry field and click Submit.
Submit	Click to test a form-level JavaScript application with the values you specified in the Text Field entry field.
JavaScript Under Test	Displays the source code for the JavaScript application you are currently testing.

Build Database Objects

Build Database Objects

Description Use this page to choose the type of database object you want to create.

This page contains:

Create a Database Object

Click the type of database object you want to

create. For example, click  to create an index. After you click an object type, a wizard guides you through the pages for creating the object.

Database object finish

Use this page to create a new database object based on your selections in the previous pages of the object build wizard.

If you are satisfied with the choices you made in the wizard, click **OK** to create the object.

If you aren't satisfied with your choices, click the  button to navigate back to any wizard page where you want to make changes. Always click the  and  buttons to navigate to and from previous wizard pages. Don't use your web browser's Back and Forward buttons.

After you make changes, click  to return to the finish page. Then, click **OK** to create the object. WebDB does not create the object until you click **OK** on this page.

After you click **OK** on this page, you still can edit the object by clicking  at the bottom of any WebDB page, clicking the type of object you want to edit, and entering its name and schema on the first page that appears in the object build wizard.

Functions

Create Function: Schema, Function Name, and Return Datatype

Description Use this page to choose a name for the function, the name of the schema that will own the finished function, and a return datatype.

This page contains:

Schema	Choose the schema that will own the finished function. Only schemas that you are allowed to build in are listed in the drop-down list.
Function Name	Type the name you want to use to identify the function. Note If you choose a function name that is already in use by an object owned by the schema you choose in the Schema field, WebDB will overwrite the object. Use the WebDB Browse feature to ensure to that another object with the same name does not exist in the schema.
Return Datatype	Choose a datatype for the value returned by the function.

Create Function: Arguments

Description Use this page to specify the names of one or more arguments that will be passed to the function, and their datatypes.

This page contains:

Argument Name	Type a name to identify each argument that will be passed to the function.
In/Out	Choose whether the argument is an input, an output, or both. • IN - Passes a value or constant from the calling environment to the function. • OUT - Returns a value from the function to the calling environment. • IN OUT - Passes a value from the calling environment to the function, and returns a possibly different value from the function to the calling environment.
Datatype	Choose the argument's datatype.
Default Value	Type a default value for the argument. This value is used to initialize the parameter if the end user does not specify one in the function call.
Add More	Click to make five blank rows available for passing additional arguments.

Create Function: Function Body

Description Use this page to edit the code generated by the Create Function wizard.

This page contains:

Function Body

Type or paste code that creates a function object.

Indexes

Create Index: Schema and Index Name

Description Use this page to choose a name for the index and the schema that will own it. You also identify in this page the name of the table on which the index will be based.

This page contains:

Schema	Choose the schema that will own the finished index. Only schemas that you are allowed to build in are listed in the drop-down list. Note After you choose the schema, click your browser's Reload button to refresh the contents of the Table drop-down list on this page.
Index Name	Type the name you want to use to identify the index. Note If you choose an index name that is already in use by an object owned by the schema you choose in the Schema field, WebDB will overwrite the object. Use the WebDB Browse feature to ensure that another object with the same name does not exist in the schema. Tip It's a good idea to create an index name that reflects the table and columns being indexed.
Table	Choose the table on which the index will be based. Note You can choose only tables owned by the schema you specified in the Schema field.
Uniqueness	Choose whether the index will be unique or nonunique:

- UNIQUE - ensures that values in the table columns specified in the next page of this wizard are unique.
- NONUNIQUE - values in the table columns specified in the next page of this wizard are not unique.

Create Index: Table Columns

Description Use this page to choose one or more columns in the table to be indexed. Specifying multiple columns creates a composite index. The order in which you specify columns on this page is the order in which the columns are indexed.

This page contains:

Column Name

Choose one or more columns in the table that you specified in Schema and Index Name page of this wizard.

Note Windows users can choose more than one column by clicking it while pressing the Ctrl key.

Add More

Click to display more **Column Name** entry fields on this page that you can use to choose additional table columns.

Create Index: Storage Parameters

Description Use this page to choose storage parameters for the index. By default, an index's tablespace and storage are the same as the tablespace on which it is based, but you can override these in this page.

This page contains:

Tablespace	Choose the name of the tablespace where the index will be stored.
Initial Extent	Type the size, in bytes, of an initial area of disk space that can be used by the index if it fills the tablespace.
Next Extent	Type the size, in bytes, of all subsequent extents that will be used if the disk space in the Initial Extent is used.
Max Extents	Type the maximum number of extents that can be allocated as the index grows.
Min Extents	Type the minimum number of extents that will be created.

Note

- Specify the initial and next extents in this page as an integer (specifies the number of bytes), followed by a K (number of bytes multiplied by 1024) or an M (number of bytes multiplied by 1048576); for example, an **Initial Extent** of 24K.

Packages

Create Package: Schema and Package Name

Description Use this page to choose a name for the package and the schema that will own it.

This page contains:

Schema

Choose the schema that will own the finished package.

Only schemas that you are allowed to build in are listed in the drop-down list.

Package Name

Type the name you want to use to identify the package.

Note If you choose a package name that is already in use by an object owned by the schema you choose in the **Schema** field, WebDB will overwrite the object. Use the WebDB Browse feature to ensure to that another object with the same name does not exist in the schema.

Create Spec and Body

Description Use this page to specify code that creates a specification and body for the package.

This page contains:

Package Specification

Type or paste PL/SQL code that creates a package specification.

A package specification contains the public list of functions, procedures, variables, constants, cursors, and exceptions contained within the package.

Package Body

Type or paste PL/SQL code that creates a package body.

A package body contains the PL/SQL code implementing the specification.

Procedures

Create Procedure: Schema and Procedure Name

Description Use this page to specify a name for the procedure and the schema that will own it.

This page contains:

Schema	Choose the schema that will own the finished procedure. Only schemas that you are allowed to build in are listed in the drop-down list.
Procedure Name	Type the name you want to use to identify the procedure. Note If you choose a procedure name that is already in use by an object owned by the schema you choose in the Schema field, WebDB will overwrite the object. Use the WebDB Browse feature to ensure to that another object with the same name does not exist in the schema.

Create Procedure: Arguments

Description Use this page to specify the names of one or more arguments that will be passed to the procedure, and their datatypes.

This page contains:

Argument Name	Type a name to identify each argument that will be passed to the procedure.
In/Out	Choose whether the argument is an input, an output, or both. • IN - Passes a value or constant from the calling environment to the procedure. End users of the procedure must specify a value for an input argument. • OUT - Returns a value from the procedure to the calling environment. • IN OUT - Passes a value from the calling environment to the procedure, and returns a possible different value from the procedure to the calling environment.
Datatype	Choose the argument's datatype.
Default	Choose a default value for the argument. This value is used to initialize the argument if the end user has not specified one in a procedure call.
Add More	Click to make five blank rows available for passing additional arguments.

Sequences

Create Sequence: Schema and Sequence Name

Description Use this page to choose a name for the sequence, the name of the schema that will own it, and sequence increment values.

This page contains:

Schema	Choose the schema that will own the finished sequence. Only schemas that you are allowed to build in are listed in the drop-down list.
Sequence Name	Type the name you want to use to identify the sequence. Note If you choose a schema name that is already in use by an object owned by the schema you choose in the Schema field, WebDB will overwrite the object. Use the WebDB Browse feature to ensure to that another object with the same name does not exist in the schema.
Increment By	Type a number that will be used to increment the sequence. A positive number increases the sequence by the value of the number. A negative number decreases the sequence by the value of the number.
Start With	Type a starting number for the sequence. The Start With value must be greater than the Min Value.
Min Value	Type the lowest number the sequence will generate. For example, if you specify an Increment By number of -1, a Start with Value of 100, and a Min Value of 1, the sequence would decrement 99 times before reaching the Min Value.

Synonyms

Create Synonym: Schema and Synonym Name

Description Use this page to choose a name for the synonym and the schema that will own it. You can also specify a database link if the object is stored in a remote database.

This page contains:

Schema	Choose the schema that will own the finished synonym. Only schemas that you are allowed to build in are listed in the drop-down list.
Synonym Name	Type the name you want to use to identify the synonym. Note If you choose a synonym name that is already in use by an object owned by the schema you choose in the Schema field, WebDB will overwrite the object. Use the WebDB Browse feature to ensure to that another object with the same name does not exist in the schema.

Create Synonym: Object Name

Description Use this page to choose the object on which the synonym is based.

This page contains:

Object

Choose the table or view with which the synonym will be associated.

Notes The pop-up list displays only objects owned by the schema you selected in the Schema and Synonym Name page.

You can create a synonym of a synonym by choosing the synonym's name in the **Object** drop-down list.

Note

- If the **Object** is contained in a remote database, you can type a link to the database; for example, EMP@REMOTE_DB.

Tables

Create Table: Schema and Table Name

Description Use this page to specify a name for the table and the schema that will own it.

This page contains:

Schema

Choose the schema that will own the finished table.

Only schemas that you are allowed to build in are listed in the drop-down list.

Table Name

Type the name you want to use to identify the table.

Note If you choose a table name that is already in use by an object owned by the schema you choose in the **Schema** field, WebDB will overwrite the object. Use the WebDB Browse feature to ensure to that another object with the same name does not exist in the schema.

Create Table: Columns

Description Use this page to choose the names of the columns that will be in the table, and the datatype of each column's data.

This page contains:

Column Name	Type a name to identify each column in the table.
Datatype	Choose a datatype for the values that each column will contain.
Length	Applicable to NUMBER, VARCHAR, and RAW datatypes. Type the total number of digits or characters in the characters in the column. The length can range from 1 to 38.
Precision	Type the number of digits to the right of the decimal point. The precision can range from -84 to 127. Note This field is applicable only to NUMBER datatypes.
Null?	Check to allow null values in the table column.
Primary Key?	Check if this column is a primary key.
Add More	Click to add five more blank rows to the page. This allows you to add five more columns to the table.

Create Table: Storage

Description Use this page to choose storage parameters for the table.

This page contains:

Tablespace	Choose the name of the tablespace where the table will be stored.
Initial Extent	Type the size, in bytes, of an initial area of disk space that can be used by the table if it fills the tablespace.
Next Extent	Type the size, in bytes, of all subsequent extents that will be used if the disk space in the Initial Extent is used.
Min Extents	Type the minimum number of extents that will be created initially; for example, 1.
Max Extents	Type the maximum number of extents that can be created after the disk space in the Initial Extent is used; for example 100.

Note

- Specify the initial and next extents in this page as an integer (specifies the number of bytes), followed by a K (number of bytes multiplied by 1024) or an M (number of bytes multiplied by 1048576); for example, an **Initial Extent** of 24K.

Triggers

Create Trigger: Schema and Trigger Name

Description Use this page to choose a name for the trigger and the schema that will own it.

This page contains:

Schema

Choose the schema that will own the finished trigger.

Only schemas that you are allowed to build in are listed in the drop-down list.

Trigger Name

Type the name you want to use to identify the trigger.

Note If you choose a trigger name that is already in use by an object owned by the schema you choose in the **Schema** field, WebDB will overwrite the object. Use the WebDB Browse feature to ensure to that another object with the same name does not exist in the schema.

Create Trigger: Trigger Table and Options

Description Use this page to identify the name of the table on which the trigger will be based and the SQL statements that cause the trigger to execute. You also choose a trigger type in this page.

This page contains:

Table

Choose the table that the trigger will reference. The trigger will execute when the SQL command or command you specify on this page executes against this table.

Note You can choose only tables owned by the schema you specified in the **Schema and Script Name** page of this wizard.

Type

Choose a trigger type:

- **BEFORE EACH ROW** - trigger executes once before each table row that is affected by a triggering statement such as INSERT, UPDATE, or DELETE.
- **BEFORE STATEMENT** - trigger executes once before a triggering statement (such as INSERT, UPDATE, or DELETE) without regard to the number of rows processed.
- **INSTEAD OF EACH ROW** - trigger executes instead of the triggering statement for each table row that is affected by the statement. **Note** This is supported in Oracle 8i.
- **INSTEAD OF STATEMENT** - trigger executes instead of a triggering statement such as INSERT, UPDATE, or DELETE. **Note** This is supported in Oracle 8i .
- **AFTER EACH ROW** - trigger executes once after each table row that is affected by a triggering statement.

Triggers

- AFTER STATEMENT - trigger executes once after a triggering statement.

On Insert

Check to cause the trigger to execute when the SQL INSERT command executes.

On Update

Check to cause the trigger to execute when the SQL UPDATE command executes.

On Delete

Check to cause the trigger to execute when the SQL DELETE command executes.

Create Trigger: Trigger Script

Description Use this page to edit the code generated by the Trigger Wizard.

This page contains:

Trigger Body

Type or paste SQL code that creates a trigger object.

Views

Create View: Schema and View Name

Description Use this page to choose a name for the view and the schema that will own it.

This page contains:

Schema

Choose the schema that will own the finished view.

Only schemas that you are allowed to build in are listed in the drop-down list.

View Name

Type the name you want to use to identify the view.



If you choose a view name that is already in use by an object owned by the schema you choose in the **Schema** field, WebDB will overwrite the object. Use the WebDB Browse feature to ensure that another object with the same name does not exist in the schema.

Create View: View Body

Description Use this page to edit the code generated by the View Wizard.

This page contains:

View Body

Type or paste SQL code that creates a view object.

Note

- Updateable columns are displayed in black, non-updateable columns are displayed in blue, and required (mandatory) columns are displayed in red.

Create Sites

Manage Sites page

Description Use this page to create a new WebDB web site, or find an existing web site. After you find a web site, you can click its name to navigate to it.

This page contains:

Create a New Site

Create	Click to create a new WebDB web site. The site build wizard guides you through all pages for creating a web site.
--------	---

Find an Existing Site

Find	Type the name of the web site you want to find, then click the Find button.
------	--

Select a Recently Edited Site

Name	Displays the name of the five most recently created or edited web sites. Click a Name to open the site.
Schema	Displays the schema that owns the web site.
Type	Identifies this as a web site.
Changed	Displays in days, hours, minutes, and seconds how long ago the web site was created or last edited.
By	Displays the developer who created the web site.

Site Name

Description Use this page to name your WebDB site.

This page contains:

Site Name

Type the name you want to use to identify the site. After the site is created, you use this name to search for it on the Sites: Site Building page.

Notes

- You must specify a name (null is not allowed).
- The name must be unique.

Select Owing Schema

Description Use this page to choose a name for the schema that will own the site. The Owing Schema cannot currently exist in the database and is generally not used as an actual user account.

The Owing Schema and two other user accounts will be created when you create the WebDB site:

- The Owing Schema that owns the site.

The user account that allows public access to the site. WebDB uses the name and password of this account to automatically configure a Listener Database Access Descriptor (DAD). This DAD allows public users to access the WebDB site without having to log on.

The name of this user account is the **Owing Schema** name suffixed with `_PUBLIC`. For example, if you specify `MYSITE` as the **Owing Schema**, a user account will be created with the name `MYSITE_PUBLIC`.

- A user account used by a site administrator to log on to the site with administration privileges.

The name of this user account is suffixed with `_ADMIN`. For example, if you specify `MYSITE` as the **Owing Schema**, a user account will be created with the name `MYSITE_ADMIN`.

The default passwords for Owing Schema, public, and administrator user accounts are the same as their corresponding schema names; for example, `MYSITE_PUBLIC/MYSITE_PUBLIC`. After the site is created, you must log in to the administration account and change the passwords for the Owing Schema and administration account.

This page contains:

Owing Schema

Type the name of the schema that will own this site.

The **Owning Schema** cannot be a schema that currently exists in the database. The Site Creation wizard creates a new schema and two user accounts in the database based on the name that you specify in this text box.

Note

- The Site Build wizard automatically grants the schema all the necessary privileges that it needs.

Language

Description Use this page to select the language for all text that appears in your site. The drop-down list of languages is based on the character sets that are installed in the database where you are creating this site.



After you create the site by completing all of the steps in this wizard, you cannot change this setting.

This page contains:

Language

Choose the language that you want to use to display all text that appears in your WebDB site.

Select Tablespaces

Description Use this page to choose the required tablespaces for the WebDB site.



After you create the site by completing all of the steps in this wizard, you cannot change these settings.

This page contains:

User Tablespace

Choose a tablespace that will be used to store the objects that render the HTML for the web site. Click  to the right of the text box to search for tablespaces.

The tablespace will contain objects that will not change dramatically in size over the life of the site.

Temporary Tablespace

Choose a tablespace for creation of temporary storage for operations such as sorting table rows. Click  to the right of the text box to search for temporary tablespaces.

Document Tablespace

Choose a tablespace that will be used to store any items uploaded to a WebDB Web site. These item types can include files, images, folders, and stored procedures.

Note The Document Tablespace will gradually fill as end users add items to WebDB Web sites. You should choose a tablespace large enough to accommodate these additions.

You can choose the same Default, Temporary, or Document tablespace. For example, you can choose USER_DATA as the Default Tablespace and the Document Tablespace.

Select Demos

Description Use this page to choose which demonstration folders to install on the WebDB site you are creating. The demonstration folder can help new end users become familiar with features of a WebDB site.

This page contains:

Demonstration folders to
install

Check the check box next to the demo you want
to install.

Create Site

Description Use this page to review the choices you made in this wizard. You can

change choices you have made by using the  and  arrows. When you are satisfied, click **Finish** to start the web site creation process. A status page indicates the progress of the installation.

The WebDB site will be created with the name you specified on the Site Name page and the schema that you specified in the Owning Schema page of the wizard. The site will be totally self-contained in the schema. If you decided to drop the WebDB site at a later date, you will drop the entire contents of schema that owns the site.

Once installation is complete, you can click **Done** to return to WebDB. You can navigate to the web site by clicking its name in the **Select a Recently Created Site: Name** list on the Sites: Site Building page.

Changing Default Passwords



After the site is created, you must log in to the administration account and change the passwords for the Owning Schema and administration account.

To change these passwords:

1. Click  at the bottom of any WebDB page.
The Site Building page displays.
2. Click the name of the site you just created in the **Select a Recently Created Site: Name** list.
3. Click **Site Administration**. Type the user name and default password for the site administration user account; for example, MYSITE_ADMIN/MYSITE_ADMIN.
4. Click  or the **Administration** link from the site's navigation bar.
5. From the Administration page, click **User** under Access Managers to display the User Manager.

Create Site

6. In the **Find User: Name** field, type the user name you specified when you logged in to the site; for example, MYSITE_ADMIN, then click **Edit**.

The Details tab of the User Manager displays.

7. Type the new password in the **Password** field.
8. In the **Confirm Password** field, type the password again to verify that you have entered it correctly.
9. Click **Apply**.
10. Repeat steps 6-9 to change the password for the Owning Schema.

Configuring the Site Data Access Descriptors

The WebDB site is automatically configured with default Data Access Descriptor (DAD) information for public and authorized user access of the WebDB site.



If you want to change a DAD name, you must first change it in the web site's Site Manager, then in the Listener Settings page.

To change a DAD name:

1. Log in to the administration user account by following steps 1-4 above.
2. Click **Site**.
3. Change a DAD name at the bottom of the Site Manager: Main tab page.
4. Click **Apply**.

Note The following step is performed in WebDB itself, not in the web site.

5. On the WebDB home page, click , then **Listener Settings**.
6. Scroll down to the **Data Access Descriptor Names** you created for this site. There should be two DAD names for the web site:

A DAD for public access to the web site. This DAD name is the same name as the owning schema; for example, **MYSITE**. The DAD is automatically configured with the user name and password you see on this page. This means that public users can access the WebDB site without having to log on.

A DAD for authorized access to the web site. This DAD name is the same name as the owning schema plus the character **S**; for example **MYSITES**. This DAD is not configured with a user name and password. This means that users cannot access this portion of the web site without first logging on.

7. Change the **Data Access Descriptor Name** to the new name you specified on the Site Manager: Main tab page.
8. (Optional) If there are other DAD settings you want to make, type them in the entry fields on this page and click **Apply**.

Click  if you have questions about any entry field on this page.

Administration

Role Manager

Description Use this page to create a new role or find an existing or recently created role. Once you find a role, you can view all members of the role, add new members to the role, or remove current members from the role. A member can be a user or another role.

This page contains:

Create a New Role

Role

Type a name for the new role.

Notes Special characters such as % or # are not allowed in the role name.

Blank characters are not allowed in the role name. Type an underscore character (_) to add a space in a name. For example, you can create a role named MY_ROLE, but not MY ROLE.

Create

Click to create the role with the name you specified in the **Role** text box.

After you click **Create**, the Manage Roles page displays. You can add users and other roles to the role using options on this page.

Find an Existing Role

Role

Type the name of the role you want to find. Once you find the role, you can view all role members, or add and remove members from the role.

Click  to the right of the text box to search for roles.

Find	Click to find the role you specified in the Role text box.
Select a Recently Created Role	
Role	Displays the names of the five most recently created roles. Click a role to view its members, or add and remove members from the role.
Date Created	Displays the day and time when the role was created.

Notes

- You can make a role a member of another role using the Role Manager.

Manage Roles

Description Use this page to view all members of the selected role, add new members, or remove current member from the role. A member can be any user or another role.

The selected role appears in the text at the top of the page.

This page contains:

User/ Role	Type the name of a WebDB user or role you want to add to the selected role. The name of the role you selected displays in the text at the top of the page. If you don't know the name of the user or role you want to add, click  to the right of the text box to search for roles.
Add	Click to make the WebDB user or role specified in the User/Role text box a member of the role. After you click Add , the user or role you assigned to the role appears in the Member list box.
Members	Displays all users or roles who are current members of the selected role.
Remove	Removes a user or role selected in the Members list box from the role. Note Windows users can choose more than one member in the list by clicking it while pressing the Ctrl key.
Remove All	Click to remove all users or roles in Members list box from the role.
Apply	Click to add or remove members from roles based on the changes you made on this page.

WebDB Grant Manager

Description Use this page to find an existing or recently created database object. Once you find an object, you can grant or revoke privileges that allow a user to perform SELECT, INSERT, UPDATE, DELETE or EXECUTE actions on it.

This page contains:

Find Database Objects

Schema	Type the name of the schema that owns the database object you want to find; for example, SCOTT. Click  to search for schemas you have privileges to browse. If you leave this entry field blank, WebDB searches all schemas you have privileges to browse.
Object Type	Choose the type of database object you want to find; for example, Tables. Choose All objects to search for all object types.
Name	Type the name of the object you want to find.
Find	Click to search the database for objects based on the search criteria you specified.

Recently Created Objects

Object	Displays the name of the five most recently created database objects. Click an object to grant or revoke SELECT, INSERT, UPDATE, DELETE or EXECUTE privileges on it.
Date Created	Displays the day and time when the object was created.

Grant Manager

Description Use this page to grant or revoke database object privileges to a user.

The text at the top of the page displays the name of the object on which you are granting or revoking privileges. The page displays all developers and roles that have been granted SELECT, INSERT, UPDATE, DELETE or EXECUTE privileges on the objects. You can grant or revoke object privileges by checking or unchecking the appropriate check box, then clicking **Apply**.

If a user or role does not appear on the page, type its name in the **User/Role** text box and click **Add** to add it to the page.

This page contains:

User/Role	Type the name of a user or role that you want to grant object privileges to. Click  to the right of the combo box to search for Users or roles.
Add	Click to add the user or role you specified in the User/Role combo box to the page.
Select	Check to enable the user or role to query a table or view.
Insert	Check to enable the user or role to insert a row into a table or view.
Update	Check to enable the user or role to update a row in a table or view.
Delete	Check to enable the user or role to delete a row in a table or view.
Execute	Check to enable the user or role to run an object such as a procedure or function
Apply	Click to grant or revoke the user's or role's privileges based on your selections.

Notes

- The Grant Manager displays check boxes for updating privileges depending on the type of object on which you are granting privileges. For tables and views, for example, the **Select**, **Insert**, **Update**, and **Delete** check boxes display. For procedures, the **Execute** check box displays.
- To revoke a privilege, uncheck a **Select**, **Insert**, **Update**, **Delete**, or **Execute** check box, then click **Apply**.

Report WebDB Developer Privileges

Description Use this page to view current browse and build privileges for all WebDB developers (users with the WEBDB_DEVELOPER role).

This page contains:

WebDB Developer	Displays the name of the WebDB developer. Click a name to update to display the User Manager, where you can update the developer's build and browse privileges.
Build In	Displays all schemas privileges in which the developer currently has privileges to build and browse. Click Build In to sort the contents of this page by the schema names displayed in this column,
Browse In	Displays all schemas privileges in which the developer currently has privileges to browse. Click Browse In to sort the contents of this page by the schema names displayed in this column.

Change Your Password

Description Use this page to change the password you use to log into WebDB.

This page contains:

New Password	Type the new password you want to use to log into WebDB. Asterisks (*) appear for each character you type.
Confirm Password	Type the same password you typed in the New Password text box to confirm you want to use it. Asterisks (*) appear for each character you type.
Apply	Click to change your old password to the one you typed in the New Password text box.

User Manager

Description Use this page to create a new WebDB user and assign a tablespace where any objects created by the user will be contained. You must specify whether the user will be a WebDB developer or component schema:

- **WebDB Developers** can build, by default, components and objects in their own schemas. In addition, they can build components and objects in any Component Schemas in which they have been granted Build In privileges.
- **Component Schemas** own the components or objects built by WebDB Developers. After you create a component schema, it automatically appears on the Build Privileges tab of the User Manager as a schema available for developers to build objects and components in. You use the tab to grant to any developer privileges to build objects and components in the component schema.

You can also use this page to find existing or recently created WebDB users in order to view additional information about the user such as the privileges and roles assigned to them.

This page contains:

Create a New User

User Name

Type a user name for the new WebDB user. If you type the name of an existing user, an error message displays.

Notes Special characters such as % or # are not allowed in the user name.

Blank characters are not allowed in the user name. Type an underscore character (_) to add a space in a name. For example, you can create a user named SCOTT_P, but not SCOTT P.

Password

Type the password you want to assign to the new user you specified in the **User Name** text box.

Confirm Password	Asterisks (*) appear for each character you type. Type the same password you typed in the New Password text box to confirm you want to use it.
Default Tablespace	Asterisks (*) appear for each character you type. Choose the default tablespace that will be used to store any database objects created by the user. Note The drop-down list contains all tablespaces that have been created in the Oracle database. Refer to the Oracle documentation for more information about creating tablespaces in your database.
Temporary Tablespace	Choose the tablespace for creation of temporary storage for operations such as sorting table rows.
Oracle Profile	Choose a database profile for the user (the amount of system and database resources that are available to the user). If you don't select a profile, the DEFAULT profile is assigned to the user. Note Use Oracle SQL commands to create new profiles. Refer to the Oracle documentation for more information.
WebDB Developer	Check to enable this user to build components in his own schema as well as any component schema in which the user is granted Build In privileges. Note To grant the user privileges to build in a component schema, you must navigate to the Build Privileges tab of the User Manager.
Component Building Schema	Check to make this user a schema in which WebDB developers build components and objects. After you create a component schema, it automatically appears on the Build Privileges tab of the User Manager as a schema available

User Manager

	for developers to build objects and components in.
Create	Click to create a new user with the options you specified on this page.
Find an Existing User	
User Name	Type the name of the user you want to find.
Find	Click to find the user you specified in the Find an Existing User: User Name field.
Select Recently Created Users	
Name	Displays the names of the five most recently created database users. Click a Name to edit information about the user such as the user's privileges and roles.
Date Created	Displays the day and time when the user was created.

User Manager: User

Description Use this page to review and update information about the WebDB user you selected:

- Whether the user is a WebDB Developer or Component Building schema.
- Roles that the user is a member of.
- Object privileges granted to the user.
- WebDB Build In or Browse In privileges granted to the user.
- The default and temporary tablespaces where the objects and components created by the user are stored.
- The user's resource profile.

You can also use this page to change a user's password.

This page contains:

User Name	Displays name of the WebDB user you selected.
Password	Type the new password you want to assign to the WebDB user. Asterisks (*) appear for each character you type.
Confirm Password	Type the same password you typed in the Password text box to confirm you want to use it. Asterisks (*) appear for each character you type. Click Apply to change the password.
Default Tablespace	Displays the default tablespace that will be used to store any database objects created by the user. You can assign a different default tablespace by choosing one in the Default Tablespace drop-down list, then clicking Apply .
Temporary Tablespace	Displays the tablespace that will be used for creation of temporary storage for operations such as sorting table rows.

User Manager

	You can assign a different temporary tablespace by choosing one in the Temporary Tablespace drop-down list, then clicking Apply.
Oracle Profile	Displays the resource profile assigned to the user. You can assign a different profile to the user by choosing one in the Oracle Profile drop-down list, then clicking Apply. Note Use Oracle SQL commands to create new profiles. Refer to the Oracle documentation for more information.
WebDB Developer	Check to enable this user to build components in his own schema as well as any component schema in which the user has been granted Build In privileges. Note To grant the user privileges to build in a component schema, you must navigate to the Build Privileges tab of the User Manager.
Component Building Schema	Check to make this user a schema in which WebDB developers build components and objects. After you create a component schema, it automatically appears on the Build Privileges tab of the User Manager as a schema available for developers to build objects and components in.
Roles	Displays all roles that the user is a member of. Click this tab to assign a user to a new role or remove user from a role.
Grants	Displays all database objects on which the user has been granted SELECT, INSERT, UPDATE, DELETE, or EXECUTE privileges. Click this tab to grant or revoke the user's privileges to perform these actions on an object.

Build Privileges	Displays all schemas the user has privileges to Build In. Click this tab to enable the user to Build In an additional schema or to revoke Build In privileges.
Browse Privileges	Displays all schemas the user has privileges to Browse In. Click this tab to enable the user to Browse In an additional schema or to revoke Browse In privileges.
Apply	Click to update the user's information or password with the changes you have made on this page.

User Manager: Grants

Description Use this page to grant or revoke privileges that allow users to grant SELECT, INSERT, UPDATE, DELETE or EXECUTE privileges on database objects.

You can grant privileges for one or more objects shown on the page by checking or unchecking the appropriate check box next to it, then clicking **Apply**. If an object does not appear on the page, type its name in the **Object** combo box and click **Add** to add it to the page.

Object names display on this page by object type (**Sequences, Table/Views, Procedures/Functions/Packages**).

This page contains:

Object	Type the name of a database object. You will be granting privileges on this object to the selected user or role. Click  to search for objects.
Add	Click to add the object you specified in the Object combo box to the page.
Select	Check to enable the user or role to query a table or view.
Insert	Check to enable the user or role to insert a row into a table or view.
Update	Check to enable the user or role to update a row in a table or view.
Delete	Check to enable the user or role to delete a row in a table or view.
Execute	Check to enable the user or role to run an object such as a procedure or function
Apply	Click to grant or revoke the developer's or role's privileges based on your selections.

Note

- The User Manager: Grants page displays check boxes for updating privileges depending on the type of object on which you are granting privileges. For tables and views, for example, the **Select**, **Insert**, **Update**, and **Delete** check boxes display. For procedures, the **Execute** check box displays.
- To revoke a privilege, uncheck a **Select**, **Insert**, **Update**, **Delete**, or **Execute** check box, then click **Apply**.

User Manager: Roles

Description Use this page to assign or remove a member from a role. A member can be a user or another role.

The text at the top of the page displays the name of the user to whom you are assigning or removing a role.

This page contains:

Role	Type the name of a role you want to assign to the user. The name of the user you selected is shown in the upper left corner of the page. Click  to the right of the text box to search for roles.
Add	Click to add a role to the list of roles assigned to the user. After you click Add , the role you are assigning to the member appears in the Is Member Of list box. Note You must click Apply to add the user to the role.
Is Member Of	Displays all roles currently assigned to the member.
Remove	Click to remove the selected role from the list of roles assigned to the WebDB user. Note Windows users can choose more than one role in the list by clicking it while pressing the Ctrl key.
Remove All	Click to remove all roles from the member.
Apply	Click to assign or remove a role from the user.

Notes

- You cannot make a role a member of another role using the Role tab of the User Manager. You can do this in the Role Manager , however.

User Manager: Build Privileges

Description Use this page to grant or revoke privileges to build in Component Schemas. The Build In privilege enables a WebDB Developer to build a component in a specified Component Schema.

The **Current Privileges** list box displays all Component Schemas in the database plus the schema for the WebDB Developer. The name of the WebDB Developer's schema displays in the upper left corner of the page after the text, **Build Privileges**. Selected (highlighted) schemas in the list box are those in which the WebDB Developer currently has Build In privileges.

This page contains:

Current Privileges	Displays all Component Schemas in the database plus the selected WebDB Developer's own schema. Select (highlight) a schema to grant to the WebDB Developer privileges to Build In it. Deselect a schema to revoke the WebDB Developer's privileges to Build In it.
Apply	Click to update the WebDB Developer's Build In privileges based on the changes you made on this page.

Note

- Component Schemas are identified using the **Component Building Schema** check box on the User Manager page .
- Granting a user privileges to Build In a schema automatically grants them Browse In privileges in the same schema

User Manager: Browse Privileges

Description Use this page to view all schemas that the selected user has privileges to browse. The text at the top of the page displays the user you selected.

You can also grant privileges to browse other schemas, or revoke existing browse privileges.



The browse privilege enables the user to search for objects in the selected schema. To use an object to build a component, the user must explicitly be granted object privileges on the Grant Manager page.

This page contains:

Schema

Type the name of the schema in which you want to enable the selected WebDB user to browse. The user you selected is shown in the text at the of the page.

If you don't know the name of the schema, click  to the right of the text box to search for schema names.

Note Use this entry field to grant privileges to browse in schemas that are not listed in the **Current Privileges: Schema** list. If the schema appears in the list, you only need to update the privileges for the WebDB user by checking or unchecking the **Browse In** check box.

Add

Click to grant Browse In privileges to the user.

After you click **Add**, the schema displays in the **Current Privileges: Schema** list.

Current Privileges:

Schema

Displays all schemas in which the selected user is currently allowed to browse.

User Manager

Browse In

Check to enable the user to browse the schema shown in the **Current Privileges: Schema** list.

Uncheck to revoke the privilege.

Apply

Click to update the user's browse privileges based on the changes you made on this page.

Listener Settings

Oracle WebDB PL/SQL Gateway Settings

Description Use this page to update Database Access Descriptors (DADs) for WebDB. A DAD is a set of values that specify how the PL/SQL gateway connects to the Oracle database server to fulfill an HTTP request. WebDB DADs are conceptually similar to Oracle Application Server (OAS) DADs, but contain different settings.

There are separate DADs associated with each instance of WebDB you installed in the database and for each web site you create using WebDB .

For example, if you install WebDB and use it to create a web site, the following DADs are automatically configured with default values that you specify at installation time:

- A DAD for the component building features of WebDB. The name for this DAD is the same as the **WebDB User Name** that was specified in the WebDB install wizard when WebDB was installed on Windows NT or Solaris.

A DAD for public access of the web site. This is the portion of the web site accessible to all end users. The DAD name for this account is the same name as the Owning Schema name that was specified in the site creation wizard when the site was built; for example, **MYSITE**. The DAD is automatically configured with the user name and password you see on this page. This means that public users can access the WebDB site without having to log on.

A DAD for authorized access the web site. This is the portion of the web site that a user must type a password to access. The DAD name for this account is the same name as the owning schema plus the character S; for example **MYSITES**. This DAD is not configured with a user name and password. This means that users cannot access site administrative features without first logging on.

As you generate more web sites at your location, additional DADs for the public and administrative user accounts for these sites are generated.



You can update values in any field on this page by typing a new one over the old value and clicking **Apply**.

Listener Settings

This page contains:

Change Listener Settings	Click to display a page where you can change WebDB listener settings.
--------------------------	---

Global Settings

Default Database Access Descriptor (DAD)	Specify a path that points to the default DAD. If the end user types a WebDB URL without specifying the DAD name, the home page for the default DAD will be displayed.
--	--

Default = WebDB

You can change the DAD name by typing a new one in this field.

Apply	Click to update the WebDB PL/SQL Gateway settings with the changes you made on this page.
-------	---

Reset	Click to clear or reset to their default values all fields on this page.
-------	--

Data Access Descriptor Settings

Database Access Descriptor Name	Displays the name for this DAD. The name is set at installation time or during creation of new WebDB web sites. You can change the name by typing a new one in this field.
---------------------------------	--

Oracle User Name	Displays the Oracle database account user name. The user name is typically set at WebDB installation or during creation of new WebDB web sites. You can change it by typing a new name in this entry field.
------------------	---

Oracle Password	Displays the Oracle database account password. The password is typically set at WebDB installation, but you change it by typing a new password in this entry field.
-----------------	---

Notes The **Oracle User Name** and **Password** are the default user name and password for logging in to WebDB or a web site created with WebDB. If you leave the **Oracle User Name**

	and Oracle Password entry fields blank, you will be prompted to create a user name and password when first logging into WebDB. There should always be an Oracle User Name and Password for <code>_PUBLIC</code> web site DADs. This enables public users to access the WebDB site without having to log on.
Oracle Connect String	Type a net alias if you are using a remote database. Leave this field blank if the database is local.
Maximum Number of Worker Threads	Type the number of worker threads that will be used to service applications. Tip You'll need to adjust this number depending on your server, its capacity, and the number of connected users. As a rule of thumb, set this number at between 5 and 10 at a medium sized installation (approximately 200 users).
Keep Database Connection Open Between Requests?	Choose whether, after processing one URL request, the database connection should be kept open to process future requests. In most configurations, specify Yes for maximum performance.
Default (Home) Page	Type the PL/SQL procedure that will be invoked when none is specified as part of the URL itself. For example, if you specify a default home page of <code>webdb.home</code> and an end user types this URL in a browser: <code>http://webdb.myserver.com:2000/webdb/</code> WebDB will automatically update the URL to: <code>http://webdb.myserver.com:2000/webdb/webdb.home</code>
Document Table	For DADs related to the web site building features of WebDB only.

Listener Settings

	Type the name of the database table into which files uploaded to a web site created with WebDB will be stored. The default value in this entry field is based on the name of the schema in which you created the WebDB site.
Document Access Path	For DADs related to the web site building features of WebDB only. Type a path in the URL for the current installation that is used to indicate a document is being referenced. In the following URL, for example: <pre>http://webdb.myserver.com:2000/my_site/docs/corner/presentation.htm</pre> docs is the document access path. The default value in this entry field is based on the name of the schema in which you created the web site using WebDB.
Document Access Procedure	For DADs related to the web site building features of WebDB only. Type the procedure that will be used to upload and download documents. The default value in this entry field is based on the name of the schema in which you created the web site using WebDB.
Apply	Click to update the WebDB PL/SQL Gateway settings with any changes you made on this page.
Reset	Click to clear or reset to their default values all entry fields on this page.

Oracle WebDB Listener Settings

Description Use this page to update settings for the WebDB Listener such as Multipurpose Internet Mail Extension (MIME) types, directory mappings, and other miscellaneous settings.

The Listener uses MIME types to describe the content sent to the browser in response to a URL request. Directory mappings associate the physical locations of files with their virtual locations.



You can change values in any field on this page by typing a new one over the old value and clicking **Apply**.

This page contains:

Change PL/SQL Gateway Settings	Click to display a page where you can change PL/SQL gateway settings.
Change Directory Mappings	Click to navigate to entry fields that enable you to change directory mappings.
Change MIME Types	Click to navigate to entry fields that enable you to change MIME types.

Miscellaneous Settings

Home Page	Display the URL for the WebDB home page that will display after users log in.
Default Mime Type	Displays the MIME type that is used by the Listener when a MIME type for a given URL request can't be determined.
Maximum Number of Threads for Serving Files	Displays the number of threads that serve static files off the file system. As a rule of thumb, increasing the number of threads improves performance, although this is contingent on operating system limitations.

Listener Settings

Maximum Number of Dispatcher Threads	Displays the number of threads used to dispatch HTTP requests. As a rule of thumb, increasing the number of threads improves performance, although this is contingent on operating system limitations.
Logging Level	Displays how much logging information will be captured by logging files located in the listener log directory. You can use this information for debugging purposes and for performance analysis. You may want to test the following Logging Level settings to determine how much logging information you want to capture: • NONE • STANDARD • TEST • ETEST • EXTENDED • ERROR • DEBUG • EDEBUG
Apply	Click to update the Listener settings with any changes you made on this page.
Reset	Click to clear or reset to their default values all entry fields on this page.

Directory Mappings

Physical Directory	Type the path to the directory containing the files you want to map. For example, on a UNIX server, WebDB image files could be located in the physical directory: <code>myserver.home/webdb/images</code>
Virtual Directory	Type the virtual directory name; for example, <code>/images/</code> .

Apply	Click to update the Listener settings with any changes you made on this page.
Reset	Click to clear or reset to their default values all entry fields on this page.
MIME Types	
Mime Type	Specify a MIME type. MIME types describe the type of file being transferred to the web browser. The default set of MIME types that are shown in these entry fields have been chosen by WebDB. You can add new or remove existing MIME types from this set.
File Extensions	Type all file extensions that you want to map to the MIME type.
Apply	Click to update the Listener settings with any changes you made on this page.
Reset	Click to clear or reset to their default values all entry fields on this page.

Stored Results

Stored Results Manager

Description Use this page to view information about currently queued batch jobs, or find successfully completed batch jobs stored in the database as stored results. Jobs run in batch mode when an end user clicks the Batch button in a component's parameter entry form. You can search for and view any stored results that:

- you own. You own stored results if you originally submitted the batch job.
- other WebDB users own. These users must designate the stored results public for you to view them.
- have not exceeded their expiration date.

If you are the owner, you can also change the expiration date of the stored results, change the public designation, or delete the results from the database.

This page contains:

Queued Requests

Action	Click Remove to remove a batch job from the queue.
Job	Displays the batch job number. After an end user clicks the batch button on a component's parameter entry form, a message displays this number. For example: Your Job (2181) has been submitted to the batch queue.
Logging User	Displays the name of the end user who submitted the batch request.
Total Time	Displays how long a currently executing batch job has been running.
Status	Displays the current status of the batch job:

	Queued - the job is currently in the queue with other batch jobs awaiting processing Executing - the job is undergoing batch processing. The Total Time field indicates how long the job has been running.
Requery	Click to update the Stored Results Manager page with the most current information about batch jobs that are executing or in the queue.
Find Stored Results	
User	Type the name of the end user who originally submitted the batch job. For example, if a user who logs into WebDB as SCOTT submitted the job, you can search for the job by typing SCOTT as the User .
Program	Type the name of the stored results program. The program name is the same name as the procedure used to display the component for which the batch job was submitted. WebDB uses the program name to identify stored results. For example, the results of a batch job submitted on a component named MY_CHART would be stored in the database under the program name my_chart . show.
Find	Click to find the stored results of a submitted batch job based on the options you specified in User or Program .
Recently Submitted Results	
Action	Displays the actions you can perform on the stored results: • VIEW - Click to view the stored results. • EDIT - Click to edit the stored results properties. These include changing the expiration date of the stored results, changing the public designation, or deleting the results from the database.

Stored Results

Program	Displays the name of the stored results program. The program name is the same name as the procedure used to display the component for which the batch job was submitted.
Executed	Displays the date and time when the batch job completed and the results were stored in the database
User	Displays the owner of the stored results. The stored results owner is the user who originally submitted the batch job.
Public	Displays whether the owner has designated the stored results PUBLIC or PRIVATE. If a stored result is PRIVATE, only the owner can display it.
Expires	Displays the date when the stored results will be dropped from the database, as designated by the owner
Size (bytes)	Displays the size, in bytes, of the stored results.

Batch Results

Description Use this page to view the number for the batch job you submitted. For example:

Your Job (2) has been submitted to the batch queue.

You can use this number to view the status of your job on the Stored Results Manager page.

This page contains:

[View Results](#)

[Click to navigate to the Stored Results Manager page.](#)

Edit Stored Results



You must be a stored results owner to use any of the options on this page. You are an owner if you originally submitted the batch job that created the created the stored results.

Description Use this page to:

- Change the stored results program name.
- Change the expiration date of the stored results.
- Designate the stored results public or private.
- Delete the results from the database.

You can change values in any of the entry fields on this page by typing or selecting new ones, then clicking **Apply**.

This page contains:

Program Name	Type a new name for the stored results program. The default program name is the same name as the procedure used to display the component for which the batch job was submitted. For example, the results of a batch job submitted on a component named <code>MY_CHART</code> would be stored in the database under the program name <code>my_chart . show</code> .
Document Expires In	Type an expiration date (in days from the current date) for the stored results.
Is Public	Choose whether the stored results are public (viewable by any WebDB user) or private (viewable only by you).
Apply	Click to change properties of the stored results based on the options you specified on this page.
Remove	Click to remove the stored results from the database.

Oracle Reports Security

Report Definition File Access

Description: Use this page to create access to a report definition file in WebDB, or find an existing or recently edited report definition file access. WebDB stores information about the report definition files (i.e., how to run the report) as a packaged procedure. Report Definition File Access determines the following in WebDB:

- What Oracle Reports .RDF, .REP, or .XML files you want to make accessible in WebDB.
- Who has access privileges to run this report definition file.
- When this report definition file is available to run.
- How report output is delivered by restricting the report request options (i.e., required and option parameters) that are available to users at runtime. This includes specifying the Reports Servers and printers that are available to users.

For more information about report definition files, refer to the Report Builder online help. Choose Help→Report Builder Help Topics. On the index page, type: file, used by Report Builder. Then click **Display** to view the Files used by Report Builder help topic.

Prerequisites

- The report definition file must exist and the Reports Server must be able to locate it. Refer to the Publishing Reports manual for more information.
- The restricted Reports Server, where this report is run, must be accessible in WebDB. Click Administer, Oracle Reports Security, and then Server Access to add access to a Reports Server in WebDB.
- The users, who require access to run this report, must be created. Ask your DBA to create the user accounts.
- If you want to restrict when this report definition file is available to run, the availability calendar must be defined. Click Administer, Oracle Reports Security, and then Availability Calendars to create a new calendar.
- If you want to restrict parameters on the Optional Parameters page based on a list of values, that LOV must already exist. Click Build, Shared Components, and the Lists of Values (LOV) to create a list of values.

This page contains:

Create a New Report Definition File Access

Create	Click to create access to a report definition file in WebDB.
--------	--

Find an Existing Report Definition File Access

Find In Schema	Type the name of the schema that owns the report definition access.
Find	Click to find the report definition file access.

Select a Recently Edited Availability Report Definition File Access

Name	Displays the name of the five most recently edited report definition file access. Click the name to edit access to the report definition file.
Schema	Displays the schema that owns the report definition file access
Changed	Displays in days, hours, or minutes how long ago access to the report definition file was added or last edited.
By	Displays the user who last made the change.

Report Definition File Access: Report Name and Schema

Description: Use this page to identify the report definition file that is being made accessible in WebDB.

Note that the report definition file must exist and the Reports Server must be able to locate it.

Refer to the *Publishing Reports* manual for more information.

This page contains:

Owner	Choose the schema that will own the report definition file access.
Report Name	Type a descriptive name for the report (e.g., Sales_Report).
Reports Servers	Choose the Reports Server(s) that will be available for users to run this report definition file. Note that the Reports Server must be accessible in WebDB. Click Administer, Oracle Reports Security , and then Server Access to add access to a Reports Server.
Oracle Reports File Name	Type the report definition file name (e.g., sales_report.rdf). The report definition file can be an .RDF, .REP, or .XML file.
Description	Type a description as desired.

Report Definition File Access: Users and Roles

Description: Use this page to specify the users who will have access privileges to run report requests using this report definition file.

Note that the users must already be defined. Ask your DBA to create the user accounts.

This page contains:

- | | |
|----------------|--|
| All Users | Displays a list of the available users. Choose the users that will have access privileges to run this report definition file. Then click  to move the selected users to the Selected Users list. |
| Selected Users | Displays the users who have access privileges to this report definition file. Choose the users whose access privileges to this report definition file you want to revoke. Then click  to remove the selected users from the list. |

Note Click  and  to move all users between the two lists.

Report Definition File Access: Availability Calendar

Description: Use this page to select the availability calendar that determines when this report definition file will be available to run.

Note that if you want to restrict when this report definition file is available to run, the availability calendar must be defined. Click **Administer, Oracle Reports Security**, and then **Availability Calendars** to create a new calendar.

This page contains:

Availability Calendar Name	Type the availability calendar name or click  to choose the availability calendar that determines when this report definition file is available to run.
-------------------------------	--

Note

- If you do not specify an availability calendar, this report will always be available to run.

Report Definition File Access: Required Parameters

Description: Use this page to restrict the values of the required parameters that are available to the users on the Runtime Parameter Form. Required parameters determine the destination choices that will be available to the user when running a report request.

For example, you may want to restrict outputting the report to the Report Server's cache or to a printer. You can use this page to restrict the Destination options available to users on the Runtime Parameter Form to Cache and Printer.

This page contains:

Destination

Types

Choose at least one of the following destinations that will be available to users on the Runtime Parameter Form to receive report output.

Cache Sends the output directly to the Reports Server's cache.

File Saves the output to a file specified as the DESNAME (i.e., destination name) in the Runtime Parameter Form. The DESNAME parameter is automatically added to the Runtime Parameter Form.

Printer Routes the output to the printer specified in the Runtime Parameter Form. The Printer parameter determines the printers that are available for printing report output.

Mail Sends the output to the mail IDs specified as the DESNAME (i.e., destination name) in the Runtime Parameter Form. The report is sent as a file attachment. The DESNAME parameter is automatically added to the Runtime Parameter Form.

Formats

Choose at least one of the following output formats that will be available to users on the Runtime Parameter Form to format this report request. When a report is run in bitmap mode, the destination format specifies the printer driver used when the destination type is FILE.

HTML Means that the report output is sent to a file that can be read by an HTML 3.0 compliant browser (e.g., Netscape 2.2).

HTMLCSS Means that the report output is sent to a file that will include style sheet extensions that can be read by an HTML 3.0 compliant browser that supports cascading style sheets.

PDF Means that the report output is sent to a file that is read by a PDF viewer. PDF output is based upon the currently configured printer for your system. The drivers for the currently selected printer is used to produce the output.

RTF Means that the report output will be sent to a Rich Text Format file that can be read by standard word processors (i.e., Microsoft Word). Note that when you open the file in MS Word, you must choose View→Page Layout to view all the graphics and objects in your report.

Delimited Means that the report output is sent to a file that can be read by standard spreadsheet utilities, such as Microsoft Excel. The DELIMITER parameter value must be included on the Optional Parameters page if you want to give users the option of specifying a delimiter other than tab.

Postscript Means that the report output will be sent to a file that can be read by a Postscript viewer or printer.

Character Means that a report is run in character mode. When the destination type is CHARACTER, an additional DESFORMAT parameter must be defined on the Optional Parameters page (e.g., DESFORMAT= hpl) with one of the following printer definitions: hpl, hplwide, dec, decwide, decland, dec180, dflt, wide, etc. Ask your system administrator for a list of valid destination formats.

Printers

Choose the printer(s) that will be available to users on the Runtime Parameter Form to print the report output. The printers that appear on this list are determined by the Reports Server(s) chosen on the Name and Schema page. All printers associated with the selected Reports Servers are listed. When users choose a Reports Server on the Runtime Parameter Form, only those printers that are associated with the selected Report Server and that are accessible to those users are listed.

Note that to add more printers to a Reports Server access, click **Administer, Oracle Reports Security**, and then **Server Access** to edit an existing Reports Server.

Parameter Form Template

Parameter Form Template Select the template that will be used for the Runtime Parameter Form.

Click **Preview Template** to view the style of the template.

Report Definition File Access: Optional Parameters

Description: Use this page to restrict the values available to users for any additional system or user parameters that are needed to run this report definition file. For example, restricting the number of copies that can be printed or displaying data based on specific values. Consider the following example:

Suppose that you want to run a restricted Sales report for the Northeast sales region, districts one through four. In the Optional Parameters page, you could set the user parameter SALES_DIST with the low value of 1 and the high value of 4 restricting this report to these four sales districts. When users run this report, they will need to choose a sales district from a list of values on the Runtime Parameter Form to view that district's sales information. If users try to view sales information from a district outside of this range, the report will fail to run.

For more information about system and user parameters, refer to the Report Builder online help. Choose Help → Report Builder Help Topics. On the index page, type parameters, about. Then click **Display** to view the help topic About parameters.

This page contains:

Parameter Name	Type the name of the system or user parameter to restrict the values available to users (e.g., SALES_REGION or COPIES).
	Note The parameter must exist and must match the parameter defined in the report.
LOV	Type the name of the list of values, or choose from values from a predefined list of values. Click  to search for the list of values. Note that the list of values must already exist. To create a list of values, click Build, Shared Components , and the Lists of Values (LOV) .
Low Value	Type the low value to select a range of values.
High Value	Type the high value to select a range of values.
More Parameters	Click to set more parameters.

Note

- If you do not specify values in the LOV Name or Low Value and High value fields, the values of the optional parameter are not restricted. However, you can specify the parameters and values that are available to users from the Oracle Reports Parameter Form, a Runtime Parameter Form that is available only to Oracle Reports system administrators. To set the default values and specify the parameters that will be displayed to users in the Runtime Parameter Form, click **Parameters** on the Manage Component page that appears after you finish this wizard.

Report Definition File Access: Validation Trigger

Description: Use this page to create conditional parameter restrictions that cannot be defined on either the Required Parameters page or Optional Parameters page. Validation triggers are PL/SQL functions that are executed when parameter values are specified. A validation trigger is executed when users accept the Runtime Parameter Form.

Validation triggers are also used to validate the Initial Value property of the parameter. The function must return a boolean value (TRUE or FALSE). Depending on whether the function returns TRUE or FALSE, users are returned to the Runtime Parameter Form.

The following validation trigger example allows users to run a report request only when the DESTYPE is PRINTER and EMPNAME is SMITH. The report will run only if the user chooses PRINTER as the DESTYPE and SMITH as the EMPNAME on the Runtime Parameter Form.

```
Is
begin
    IF UPPER(DESTYPE) = 'PRINTER'
      AND EMPNAME = 'SMITH' THEN
      RETURN(TRUE);
    ELSE
      RETURN(FALSE);
    END IF;
end;
```

This page contains:

Validation Trigger

Validation Function	Type the PL/SQL code to define the validation function.
------------------------	---

Report Definition File Access: Create Report Definition File

Description: Use this page to create access to this report definition file in WebDB. Click **OK** to create it. This report definition file is stored in the database as a package procedure in the schema you specified in the Report Name and Schema page of this wizard.

When the package is created, the Manage Component page appears. From this page, you can edit the report definition file access, run the report, or set up defaults for the Runtime Parameter Form.

Server Access

Description: Use this page to add access to a Reports Server in WebDB, or find an existing or recently edited Reports Server access. Reports Server Access defines the following in WebDB:

- What Reports Server is available in WebDB for processing requests.
- What printer(s) are accessible for the Reports Server.
- Who has access privileges to send report requests to this Reports Server.
- When this Reports Server is available to accept report requests.

Prerequisites

- The Reports Server must be installed and configured. Refer to the *Publishing Reports* manual for more information.
- Access to the printers that are accessible to this Reports Server must be defined in WebDB. Click **Administer**, **Oracle Reports Security**, and then **Printer Access** to add a new printer registration.
- The users, who will be given access to this Reports Server, must be created. Ask your DBA to create new user accounts.
- If you want to restrict when this Report Server is available to accept report requests, the availability calendar must be defined. Click **Administer**, **Oracle Reports Security**, and then **Availability Calendars** to create a new calendar.

This page contains:

Add a Server

Create Click to add access to a Reports Server in WebDB.

Find an Existing Server

Name Contains	Type the name of the Reports Server.
Find	Click to find the Reports Server access you specified in the Name Contains field.

Select a Recently Edited Server

Name	Displays the name of the five most recently edited Reports Server access. Click a name to edit access to the Reports Server.
Last Changed	Displays in days, hours, or minutes how long ago the Reports Server access was added or last edited.
Manage	Click Delete to remove the Reports Server access from WebDB.

Server Access: Server Definition

Description: Use this page to identify the Reports Server that you are making accessible in WebDB.

This page contains:

Server Name	Type a descriptive name for the Reports Server (e.g., repserver). This is used to identify the Reports Server in WebDB.
Reports Server TNS Name	Type the TNS name. This is the service entry used to identify the Reports Server in the tnsnames.ora file. Refer to the <i>Publishing Reports</i> manual for more information.
Description	Type a description as desired.
Oracle Reports Web Gateway URL	Type the virtual location of the Oracle Reports executable (e.g., <code>http://your_webserver/cgi-bin/rwcgi60</code>). The executable referenced is based one of the following Reports Server configurations: • a Web CGI configuration references the RWCGI60 executable • a servlet configuration references the RWServlet executable Refer to the <i>Publishing Reports</i> manual for more information about the Reports Server configuration.
Run only Registered Report Definition Files	Check this box if you want only the report definition files that are accessible in WebDB to run when requested. Any report files that are not accessible in WebDB will fail to run when requested. Leave this box unchecked, if you want this restricted Reports Server to accept any report definition file, as long as the requesting user has access privileges to this Reports Server. This option is useful when: • you are first implementing Reports Server security and want to make your report files accessible in phases over a period of time. This option prevents a phased implementation from impacting your production environment.

- a report doesn't require a security check. Users can send unrestricted reports to a restricted Reports Server as long as they have access privileges to that Reports Server.

Printers

Click the printer(s) that you want to make available to this Reports Server for printing output.

Note that the printer must be accessible in WebDB. To add a printer to this list, click **Administer, Oracle Reports Security**, and then **Printers Access** to add a printer to WebDB.

Server Access: Users and Roles

Description: Use this page to make this Reports Server accessible to specific users.

Note that the users who will have access to this Reports Server must already exist in WebDB. Ask your DBA to create the user accounts.

This page contains:

- | | |
|----------------|--|
| All Users | Displays a list of the available users. Choose the users who will have access privileges to run report requests to this Reports Server. Then, click  to move the selected users to the Selected Users list. |
| Selected Users | Displays the users that have access privileges to this Reports Server. Choose the users whose access privileges to this Reports Server you want to revoke. Then, click  to remove the selected users from the list. |

Note Click  and  to move all users between the two lists.

Server Access: Availability Calendar

Description: Use this page to select the availability calendar that determines when this Reports Server is available to accept report requests.

Note that if you want to restrict when this Reports Server is available to accept report requests, the availability calendar must exist. Click **Administer, Oracle Reports Security**, and then **Availability Calendars** to create a new calendar.

This page contains:

Availability Calendar Name	Type the availability calendar name or click  to choose the availability calendar that determines when this Report Server is available to accept report requests.
-------------------------------	--

Note

- If you do not specify an availability calendar, this Report Server will always be available to accept report requests.

Server Access: Add Reports Server

Description: Use this page to add access to this Reports Server to WebDB. Click **OK** to add it to WebDB.

Printer Access

Description: Use this page to add access to a printer in WebDB, or find an existing or recently edited Printer access. Printer Access defines the following in WebDB:

- What printer is accessible in WebDB to print report requests.
- Who has access privileges to print output to this printer.
- When this printer is available to print report requests.

Prerequisites

- A printer that Oracle Reports can recognize must be installed and running. Refer to the *Publishing Reports* manual for more information on RWCLI60 command line arguments DESFORMAT and DESNAME.
- The users, who will be given access privileges to this printer, must be created. Ask your DBA to create the user accounts.
- If you want to restrict when this printer is available for processing, the availability calendar must be defined. Click **Administer, Oracle Reports Security**, and then **Availability Calendars** to create a new calendar.

This page contains:

Add a New Printer Access

Create Click to add access to a printer in WebDB.

Find an existing Printer Access

Name Contains Type the name of the printer access you want to find.

Find Click to find the printer access you specified in the Name Contains field.

Select a Recently Edited Printer Access

Name Displays the name of the five most recently edited printer access controls. Click a name to edit the printer.

Last Changed Displays in days, hours, or minutes how long ago the printer access was added or last edited.

Manage Click **Delete** to remove access to this printer in WebDB.

Printer Access: Printer Name

Description: Use this page to identify the printer that is being made accessible in WebDB.

This page contains:

Printer Name	Type a descriptive name for the printer (e.g., Sales_Printer). This name is used to identify the printer in WebDB.
OS Printer Name	Type the operating system (OS) printer name (e.g., sales_prt_111). This printer must be recognized by Oracle Reports. Refer to the <i>Publishing Reports</i> manual for more information on RWCLI60 command line arguments DESFORMAT and DESNAME.
Description	Type a description as desired.

Printer Access: Users and Roles

Description: Use this page to assign users access privileges to print output to this printer.

Note that the users must already be created. Ask your DBA to create the user accounts.

This page contains:

- | | |
|----------------|---|
| All Users | Displays a list of the available users. Choose the users that will have access privileges to print output to this printer. Then click  to move the selected users to the Selected Users list. |
| Selected Users | Displays the users that have access privileges to this printer. Choose the users whose access privileges to this printer you want to revoke. Then click  to remove the selected users from the list. |

Note

- Click  and  to move all users between the two lists.

Printer Access: Availability Calendar

Description: (Optional) Use this page to select the availability calendar that determines when this printer is available for printing output.

Note that the availability calendar must be defined. Click **Administer, Oracle Reports Security**, and then **Availability Calendars** to create a new calendar.

This page contains:

Availability Calendar Name	Type the availability calendar name or click  to choose the availability calendar that determines when this printer is available for processing.
-------------------------------	---

Note

- If you do not specify an availability calendar, this printer will always be available for processing.

Printer Access: Add Printer

Description: Use this page to add access to this printer to WebDB. Click **OK** to add it to WebDB.

Availability Calendars

Description: Use this page to create an availability calendar in WebDB, or find an existing or recently edited availability calendar. An availability calendar determines when report definition files, Reports Servers, and printers are available for use.

You can create two types of availability calendars:

- A simple availability calendar that defines a single availability rule (e.g., daily, Sunday through Saturday from 12:00 a.m. to 10:00 p.m.).
- A combined availability calendar that combines two or more availability calendars (e.g., combining a daily calendar with a maintenance calendar) into a single availability calendar.

You can apply only one availability calendar to a report definition file, Reports Server, or printer that has been made accessible in WebDB. Combining availability calendars is useful when you want to apply many availability rules to a single availability calendar. For example, your production environment may have a maintenance schedule or holidays when processing is unavailable or available only during reduced hours. You could combine a Daily calendar and Maintenance calendar in to a single availability calendar and then apply this calendar to the report definition file, Reports Server, and printer that you are making accessible in WebDB.

Suppose that you create the following simple availability calendars:

- Daily calendar with an availability period of Sunday through Saturday from 12:00 a.m. to 10:00 p.m.
- Maintenance calendar with an availability period of every Monday from 2:00 a.m. to 6:00 a.m.
- Thanksgiving calendar with an availability period from 12:00 a.m. every fourth Thursday in November to 12:00 a.m. the next day.
- Christmas calendar with an availability period from December 25 at 12 a.m. to December 26 at 12:00 a.m.

Suppose further that you want to allow processing based on the rules established by the Daily calendar, but want to prohibit processing based on the rules established by the Maintenance, Thanksgiving, and Christmas calendars. Your combined availability calendar is compiled by selecting the daily, Maintenance, Thanksgiving, and Christmas calendars, and then excluding the Maintenance, Thanksgiving, and Christmas calendars.

Suppose further that you have many holidays, in addition to Thanksgiving and Christmas, that you want to prohibit processing. You can create simple availability calendars for each

additional holiday, then combine the individual calendars into a single holidays calendar. You could create a single availability calendar by selecting the Daily, Maintenance, and Holidays calendars, and then by excluding the Maintenance and Holidays calendars.

This page contains:

Create a New Availability Calendar

Select the type of calendar that you want to create	Choose Simple Availability Calendar to create a calendar that defines a single availability rule. Choose Combined Availability Calendar to create an availability calendar that combines at least two availability calendars.
Create	Click to create the availability calendar.

Find an Existing Availability Calendar

Name Contains	Type the name of the availability calendar you want to find.
Find	Click to find the availability calendar you specified in the Name Contains field.

Select a Recently Edited Availability Calendar

Name	Displays the name of the five most recently edited availability calendars. Click a name to edit the availability calendar.
Type	Displays the availability calendar type (i.e., simple or combined).
Last Changed	Displays how many days, hours, or minutes the availability calendar was added or last edited.
Manage	Click Delete to remove the availability calendar from WebDB.

Simple Availability Calendar: Simple Availability Calendar

Description: Use this page to name the availability calendar.

This page contains:

Calendar Name	Type a descriptive name for your calendar (e.g., Weekly).
Description	Type a description of the availability calendar as desired.

Simple Availability Calendar: Date/Time Availability

Description: Use this page to determine the date and time duration and repeat pattern that applies to this availability calendar.

Note If a report is running when availability is terminated, the report will continue to run until the request has completed.

This page contains:

Duration	Determines the start and end dates that apply to this availability calendar.
Start	Specify the start month, date, and year, and start hour, minute, and division of day (e.g., starting on January 1, 2000 at 12:00 a.m.).
End	Specify the end month, date, and year, and end hour, minute, and division of day (e.g., ending on January 1, 2001 at 12:00 a.m.).
Repeat	Determines how often the duration pattern is repeated. Choose Occurs only once to repeat the date duration once. Suppose that you want availability every day in the year 2000. You could create a year 2000 Daily availability calendar with a start date of January 1, 2000 and end date of December 31, 2000, and with a repeat pattern that occurs only once (i.e., every day for one calendar year). Choose Yearly to repeat the duration once a year. Suppose you want to prohibit availability on Christmas day. You could create a Christmas holiday calendar starting on December 25, 2000 at 12:00 a.m. and ending on December 26 at 12:00 a.m. with a repeat pattern of yearly. By excluding the Christmas holiday calendar from your combined availability calendar, you can prohibit availability on Christmas day starting in the year 2000. Choose Monthly to repeat the duration every month by the date (e.g., January 3, 2000) or the day (e.g., Monday). Suppose that maintenance is performed on your Reports Servers the third day of every month making them unavailable for processing. You could create a Reports Server maintenance calendar with a date duration that starts and ends on January 3, 2000 and with a repeat pattern of monthly by date. The date duration repeats on the third

day of each month until the repeat pattern is terminated. By excluding the Reports maintenance calendar from your combined availability calendar, the Reports Server becomes unavailable to accept report requests the third day of every month (i.e., February 3, March 3, and so on).

If you choose to repeat the pattern monthly by day, then the repeat starts on the day defined in the duration. In this case, January 3, 2000 falls on the first Monday so this pattern will repeat on the first Monday of every month.

Choose **Weekly** to repeat the duration every week.

Suppose you want availability 24 hours a day on weekdays only. You could create a weekday availability calendar with a start duration of January 3, 2000 at 12:00 a.m., and end duration of January 8, 2000 at 12:00 a.m. (i.e., Monday through Friday), and with a repeat pattern of weekly (e.g., January 10, 17, and so on) until the repeat pattern is terminated..

Choose **Daily** to repeat the date duration every day.

Suppose that you want availability every day of the week (i.e., Sunday through Saturday). You could create a daily calendar with a duration that start on January 3, 2000 at 12:00 a.m. and ends on January 4, at 12:00 a.m. that repeats daily (e.g., January 4, 5, 6 and so on) until the repeat pattern is terminated.

Frequency

Choose how often the repeat pattern occurs.

Suppose that you want to prohibit availability on Thanksgiving day, which occurs on the fourth Thursday in November. You could create a Thanksgiving calendar with a start and end date of November 23, 2000 and with a repeat pattern of monthly by day (i.e., the repeat pattern is on every fourth Thursday in November). You set the frequency to 12 where the repeat pattern will occur every 12 months (i.e., every November) until the pattern is terminated. By excluding the Thanksgiving calendar from your combined calendar, the fourth Thursday in November becomes unavailable for processing.

Repeat Until

Check to terminate the repeat pattern on the specified month, date, and year and specified hour, minute, and division of day.

Suppose that you want to prohibit availability on Christmas day for the next 20 years. If the start duration is December, 25 2000 at 12:00 a.m. and the end duration is December 26 at 12:00 a.m. with

the repeat pattern that occurs yearly. You could set the Repeat Until date to December 25, 2020, then this repeat pattern will occur for the next 20 years.

Simple Availability Calendar: Summary

Description: Use this page to view a summary of the availability calendar you created. Click **Show Calendar** to view a visual representation of the availability calendar. You can drill down and view the calendar by day, week, month, or year. Green indicates availability.

Simple Availability Calendar: Create Simple Availability Calendar

Description: Use this page to create the availability calendar. Click **OK** to create it in WebDB.

Combined Availability Calendar

Description: Use this page to name the combined availability calendar.

This page contains:

Calendar Name	Type a descriptive name of the combined availability calendar (e.g. workdays).
Description	Type a description as desired.

Combined Availability Calendar: Availability Calendars

Description: Use this page to combine at least two availability calendars. You can apply only one availability calendar to a report definition file, Reports Server, and printer access. Combining availability calendars is useful when you want to apply many availability rules to a single availability calendar.

Suppose that you create a Weekday calendar with availability from Monday through Friday 24 hours a day, and a Weekend calendar with availability on Saturday and Sunday at reduced hours (e.g., 6:00 a.m. to 6:00 p.m.). You could create a single availability calendar by combining the Weekday and Weekend availability calendars, and then applying it to the report definition files, Reports Servers, and printers that you are making accessible in WebDB.

Prerequisites The availability calendars that you want to combine must already exist. Click **Oracle Reports Security** and **Availability Calendars** to create an availability calendar.

This page contains:

Availability Calendars	Displays a list of the availability calendars that have been created. Choose the calendars you want to combine, and then click  to move the selected availability calendars to the Select Availability Calendars list.
Selected Availability Calendars	Displays a list of the calendars that are added combined in this availability calendar. Choose the calendar and then click  to remove it from this list.

Note

- Click  and  to move all calendars between the two lists.

Combined Availability Calendar: Exclude Availability Calendars

Description: Use this page to prohibit availability on certain days, such as holidays.

Suppose that you create a Christmas holiday calendar with a start duration of December 25 at 12:00 a.m. and with an end duration of December 26 at 12:00 a.m. (i.e., 24 hours). If you exclude the Christmas Holiday calendar from the Weekly calendar, then availability is prohibited on Christmas day.

Note that if you have several holidays, you could create a holiday calendar for each holiday, and then combine each individual holiday calendar into a single Holidays calendar. If you exclude the holidays calendar from the Weekday and Weekend calendars, then availability is prohibited on each of the holidays defined in the holidays calendar.

This page contains:

Availability Calendars	Displays the availability calendars that are combined in this combined availability calendar. Choose the calendars you want to exclude (i.e., the calendars that specify when an report definition file, Reports Server, or printer is unavailable). Then click  to move the selected availability calendars to the Excluded Availability Calendars list.
Excluded Availability Calendars	Displays a list of the calendars that will be excluded from this combined availability calendar. Choose the calendar and then click  to remove it from this list.

Note Click  and  to move all calendars between the two lists.

Combined Availability Calendar: Summary

Description: Use this page to view a summary of the availability calendar you created. Click **Show Calendar** to view a visual representation of the availability calendar. You can drill down and view the calendar by day, week, month, or year. Green indicates availability.

Combined Availability Calendar: Create Combined Availability Calendar

Description: Use this page to create the availability calendar. Click **OK** to create it in WebDB.

Oracle Reports Parameter Form

Description: Use this page to set the default parameter values and to choose the parameters that will be visible to users on the Runtime Parameter Form. In addition, you can run this report. This page displays the required parameters defined on the Required Parameters page and any additional parameters that were defined on the Optional Parameters page.

This page contains:

Run Report	Click to run this report with the specified parameter values.
Save Parameters	Click to save the parameter value selections.
Server	Choose the Reports Server that you want to receive this report request. Only the servers you chose from the Report Name and Schema page are displayed in the list box.
Printer	Choose the printer that you want to print your report output. Only the printers you chose from the Required Parameters page are displayed in the list box.
Destype	Choose the Destination type. Only the destination types you chose from the Required Parameters page are displayed in the list box.
Desformat	Choose the Destination format. Only the destination formats you chose from the Required Parameters page are displayed in the list box.
Desname	Type the name of the file when the Destype is File or type the mail IDs when the Destype is Mail. The destination name is required when you choose File or Mail as the Destype.
<parameter>	Choose values from a list of values or type a value in text box.
Visible to user	Check each parameter that you want make available in the Runtime Parameter Form when users run this report request from a WebDB site. If the box is not checked, then the parameter is not displayed to users.

CAUTION

- When setting the parameter values for debugging purposes, be sure to delete (or not save) any confidential parameter values, such as social security numbers, from this Oracle Reports Parameter Form. Otherwise, this confidential information will be displayed to other users when you add this report request to a WebDB site.

Monitoring

Browse Activity Log

Description The Browse Activity Log page is a Query by Example form that enables you to browse the contents of the Activity Log. The Activity Log contains a record of logged end user requests for WebDB components. A request is logged if the developer who created the component selected the **Log Activity** option in the Display Options page of the component build wizard.

To browse the log, specify values for each column entry field below that you want to include in your query of the log. You can use wildcards and conditions such as <, >, or IN in your search criteria. When you finish, click **Query**.

This page contains:

Query	Click to display data based on the criteria you specified on this page.
Reset	Click to clear or reset all entry fields on this page to their default values.
Column entry field check box	Check to include the table column in your query results.
TIME STAMP	Type criteria to search for component requests based on the date they were logged. Type your search criteria in the format DD-MMM-YY.
COMPONENT TYPE	Type criteria to search for requests based on the type of component that was requested; for example, MENU or REPORT.
COMPONENT NAME	Type criteria to search for component requests based on the package that contains the component; for example, SCOTT.MY_REPORT.SHOW.
COMPONENT ATTRIBUTE	Type criteria to search for requests based on

	component attributes. Component attributes are the options specified by the developer during the creation or most recent edit of the component; for example, the name of the table on which the component is based, column and condition parameters, and look and feel options.
INFORMATION	Type criteria to search for component requests based on information defined by the creator of the component.
ELAP TIME	Type criteria to search for component requests based on the time it took for the server to fulfill the request for the component.
NUMOF ROWS	Type criteria to search for component requests based on the number of database table or view rows returned by the request.
USERID	Type criteria to search for component requests based on the user who originated the request for the component.
IP ADDRESS	Type criteria to search for component requests based on the IP address of the user who originated the request for the component.
USER AGENT	Type criteria to search for component requests based on the web browser type and machine used by the requester of the component; for example, Mozilla.
Order by	Choose column values that will be used to order table rows returned by a query. This option is equivalent to specifying a SQL ORDER BY clause. Choose Ascending to sort query results according to the alphabetic (A-Z) or numeric (starting with the lowest number) order of column values, depending on the datatype for the column. Choose Descending to sort in the reverse order.
Output Format	Choose a display format for the query results: • HTML Format

Formats the results using HTML tables and displays results on a new page in the web browser. Results that contain large amounts of data may take longer to display in this format.

- **Excel**

Downloads the results for display in Microsoft Excel.

- **ASCII text**

Formats the results using the HTML PRE tag to display results as ASCII text. This option is useful for displaying large amounts of data.

Maximum rows

Choose the maximum number of rows you want to display in the query results.

Query options

Choose one or more options for formatting your query results.

Note Windows users can choose more than one option by clicking it while pressing the Ctrl key.

Set Log Attribute

Description Use this page to set how many days the current Activity Log will log requests for components until switching to a new log. You can also view information about the Activity Log such as:

- When the first and most recent entries in the current log were made.
- The name of the database object containing the log.
- Names of the other database objects that support the log such as indexes.

This page contains:

View Log Information	Click to view additional information about the Activity Log.
Log Switch Interval	Type the number of days you want to log requests for components in the current log before switching to a new log.
Apply	Click to set the interval with the value you specified in the Log Switch Interval entry field.

Index

A

Advanced Form Building page, 88
Advanced PL/SQL code examples, 221

B

Batch Results Manager page, 356
Batch Results page, 359
Browse Activity Log, 395
Browse and Build Privileges Manager page, 334
Browse Database page, 17
Browse Object Type page, 21
Browse Objects page, 19
Browse Schemas page, 20
Build Database Object Finish page, 294
Build Database Objects page, 293

C

Calendar build wizard
 Advanced PL/SQL Code, 51
 Display Options, 43
 Name and Schema, 40
 Parameter Entry Form Options, 46
 SQL Statement, 41
 Text Options, 49
Calendar Building page, 39
Change Password page, 335
Change WebDB Listener Settings page, 353
Chart (from SQL query) build wizard
 Advanced SQL Code, 82
 Button Options, 75
 Chart Name and Schema, 70
 Display Options, 73
 Parameter options, 81
 SQL Query, 71
 Summary options, 78

 Text Options, 79
Chart build wizard
 Advanced PL/SQL Code, 69
 Button Options, 64
 Chart Name and Schema, 54
 Column Conditions, 59
 Display Options, 61
 Table/View Columns, 56
 Tables and Views, 55
 Text Options, 67
Chart Building page, 52
Component status codes, 37
Copy Component page, 246
Create Dynamic List of Values page, 276
Create JavaScript page, 288
Create Static List of Values page, 282
Create Structured U/I Template page, 259
Create Unstructured UI Template, 263

D

Drop Component page, 248
Dynamic Page build wizard
 Dynamic Page Contents, 85
 Log Activity Option, 87
 Page Name and Schema, 84
 Refine Page Contents, 86
Dynamic Page Building page, 83

E

Edit Color Definition page, 251
Edit Dynamic List of Values page, 279
Edit Font Definition page, 256
Edit Image Definition page, 254
Edit JavaScript page, 291
Edit Static List of Values page, 284
Edit Stored Results page, 360
Example reports formatted with column breaks, 222

Execute Procedure Page, 28, 32

F

- Find Component page, 35
- Find Database Object page, 22
- Finish Component page, 38
- Form from Procedure wizard
 - Advanced PL/SQL Code, 120
 - Argument Formatting and Validation, 111
 - Button Options, 117
 - Display Options, 100
 - Procedure Name, 108
 - Procedure Name and Schema, 107
 - Text Options, 118
- Frame Driver build wizard
 - Advanced PL/SQL Code, 162
 - Display Options, 157
 - Initial Page in Target Frame, 155
 - Name and Schema, 151
 - Parameter Form Display Options, 154
 - SQL Query for Frame Driver, 152
 - Text Options, 160
- Frame Driver Building page, 150
- Function wizard
 - Arguments page, 296
 - Body page, 297
 - Schema and Script Name, 295
- FWT
 - Advanced PL/SQL Code, 106
 - Button Options, 103
 - Column Formatting and Validation, 93
 - Form Name and Schema, 90
 - Table or Views, 91
 - Text Options, 104

G

- Grant Manager 2, 332
- Grant Manager page, 331
- Guidelines for writing JavaScripts, 289

H

- Hierarchies Building page, 163
- Hierarchies wizard

- Advanced PL/SQL Code, 179
- Column Conditions, 168
- Display Options, 170
- Name and Schema, 164
- Parameter Entry Form Display Options, 173
- Parameter options, 176
- Table/View Columns, 166
- Tables or Views, 165
- Text Options, 177

I

- Index build wizard
 - Schema and Script Name, 298
 - Storage Parameters, 301
 - Table Columns, 300

L

- Links wizard
 - Finish page, 273
 - Link Name page, 267
 - Target Inputs page, 270
 - Target Type and Name page, 268

M

- Manage Colors page, 249
- Manage Component page, 240, 244, 245
- Manage Execute Component Privileges page, 242
- Manage Fonts page, 255
- Manage Images page, 252
- Manage JavaScripts page, 286
- Manage Links page, 265
- Manage List of Values page, 274
- Manage U/I Templates page, 257
- Master-detail Form build wizard
 - Advanced PL/SQL Code, 138
 - Detail Row Formatting, 127
 - Display Options, 131
 - Join Conditions, 123
 - Master Row Finder, 135
 - Master Row Formatting, 124
 - Master-Detail Name and Schema, 121

- Parameter Entry Form Display Options, 134
- Tables or Views, 122
- Text Options, 136
- Menu build wizard
 - Menu Name and Schema, 182
- Menu Building page, 180

P

- Package wizard
 - Package Body and Specification, 303
 - Schema and Script Name, 302
- PL/SQL Gateway Settings page, 349
- Procedure wizard
 - Arguments, 305
 - Schema and Script Name, 304

Q

- Query and Update Table page, 23
 - Query options, 27
- Query by Example Form build wizard
 - Add Advanced PL/SQL code, 149
 - Button Options, 146
 - Display Options, 143
 - QBE Name and Schema, 139
 - QBE Results Formatting, 142
 - Table/View Columns, 141
 - Tables and Views, 140
 - Text Options, 147

R

- Rename Component page, 247
- Report (from SQL Query) build wizard
 - Advanced SQL Code, 239
 - Column Formatting, 227
 - Display Options, 229
 - Name and Schema, 224
 - Parameter Entry Form Display Options, 233
 - SQL Statement, 225
 - Text Options, 237
- Report build wizard
 - Advanced PL/SQL Code, 220
 - Column Conditions, 206

- Column Formatting, 208
- Display Options, 210
- Join Conditions, 204
- Name and Schema, 202
- Parameter Entry Form Display Options, 214
- Table and Views, 203
- Table/View Columns, 205
- Text Options, 218
- Report Building page, 200

S

- Select Demos page, 324
- Sequence wizard
 - Schema Name and Script, 306
- Set Activity Log Attribute page, 398
- Site Building page, 318
- Site Creation wizard
 - Finish page, 325
 - Schema Owner page, 320
 - Tablespaces page, 323
- Site Creation wizard
 - Language page, 322
- Site Creation wizard
 - Name page, 319
- Synonym wizard
 - Database Object and Remote Link, 308
 - Schema and Script Name, 307

T

- Table wizard
 - Columns, 310
 - Schema and Script Name, 309
 - Tablespace, 311
- Test JavaScript page, 292
- Trigger wizard
 - Table and Options page, 313
 - Trigger Body, 315
 - View Body, 317

V

- View wizard
 - Schema and Script Name, 316

W

WebDB Manage Roles page, 330
WebDB Role Manager page, 328
WebDB User Manager
 Browse In Privileges tab, 347

 Build In Privileges tab, 346
 Grants tab, 342
 Roles tab, 344
 User tab, 339
WebDB User Manager page, 336