

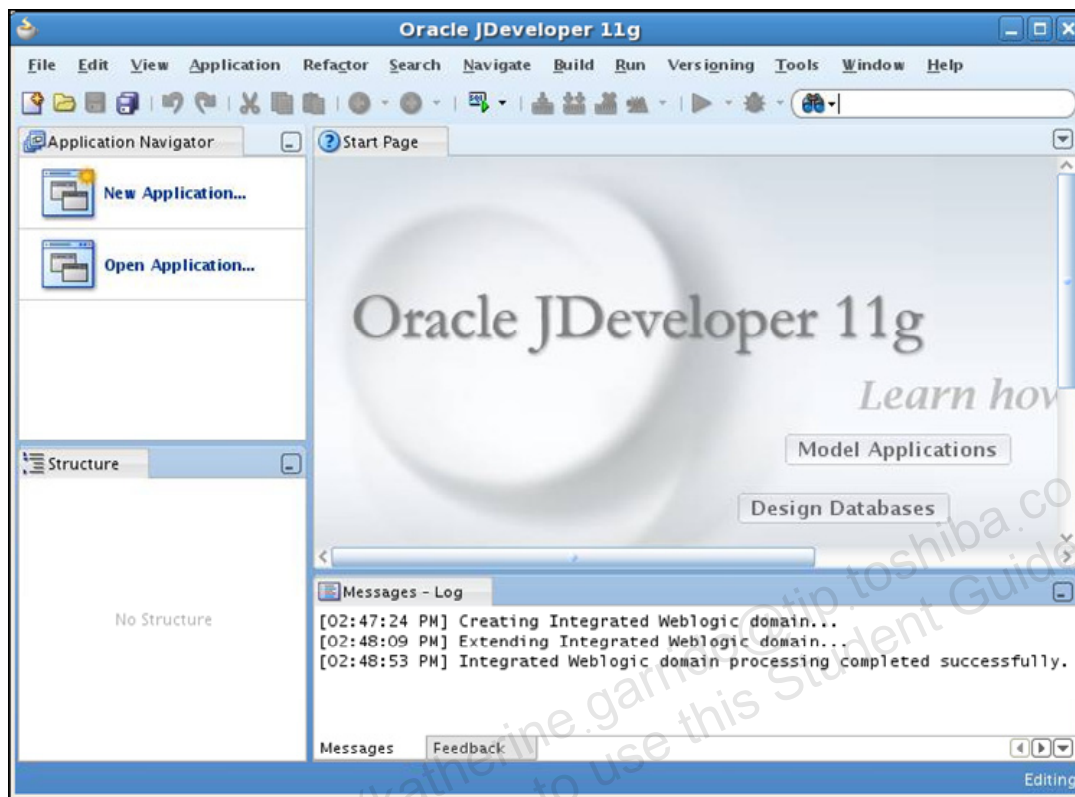
D

Using JDeveloper

ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

Oracle JDeveloper



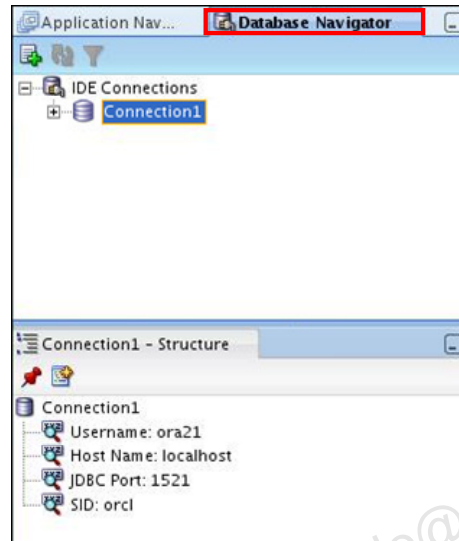
ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

Oracle JDeveloper is an integrated development environment (IDE) for developing and deploying Java applications and Web services. It supports every stage of the software development life cycle (SDLC) from modeling through deploying. It has the features to use the latest industry standards for Java, XML, and SQL while developing an application.

Oracle JDeveloper 11g initiates a new approach to J2EE development with features that enable visual and declarative development. This innovative approach makes J2EE development simple and efficient.

Database Navigator

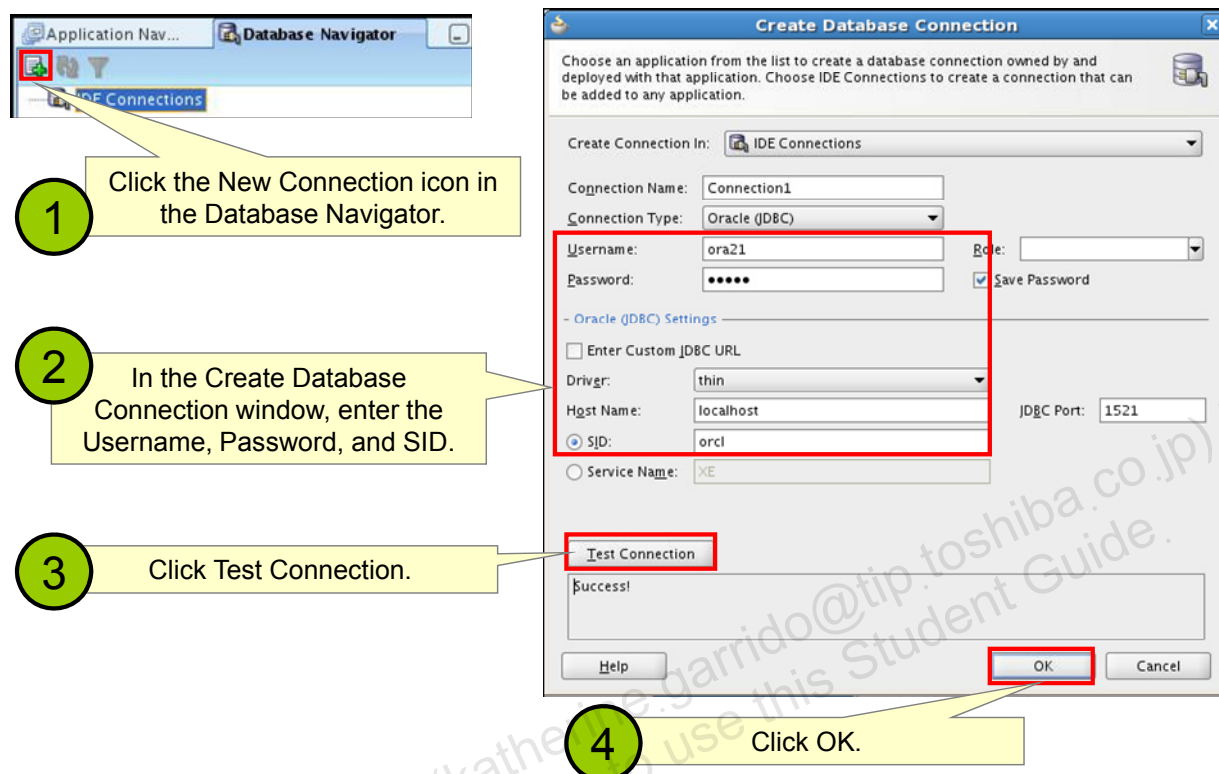


ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

Using Oracle JDeveloper, you can store information that is necessary to connect to a database in an object called “connection.” A connection is stored as part of the IDE settings, and can be exported and imported for easy sharing among groups of users. A connection serves several purposes from browsing the database and building applications, all the way through to deployment.

Creating a Connection



Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

A connection is an object that specifies the necessary information for connecting to a specific database as a specific user of that database. You can create and test connections for multiple databases and for multiple schemas.

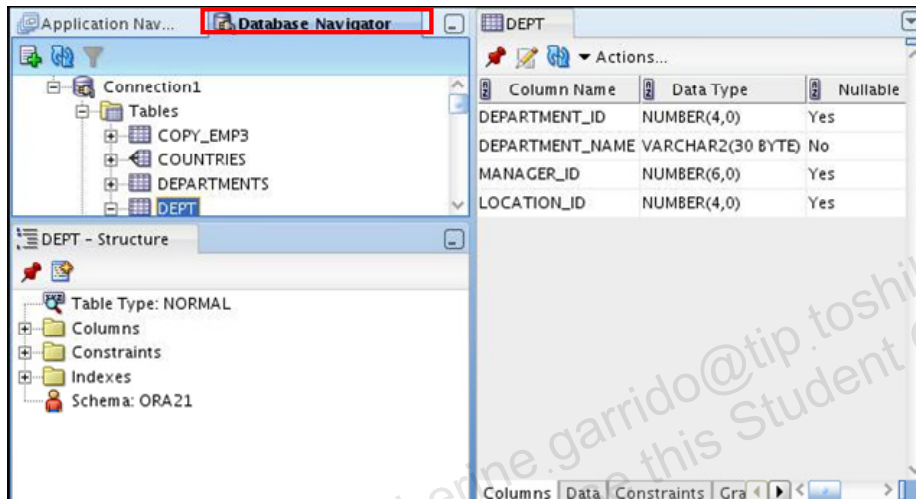
To create a database connection, perform the following steps:

1. Click the New Connection icon in the Database Navigator.
2. In the Create Database Connection window, enter the connection name. Enter the username and password of the schema that you want to connect to. Enter the SID of the database that you want to connect.
3. Click Test to ensure that the connection has been set correctly.
4. Click OK.

Browsing Database Objects

Use the Database Navigator to:

- Browse through many objects in a database schema
- Review the definitions of objects at a glance



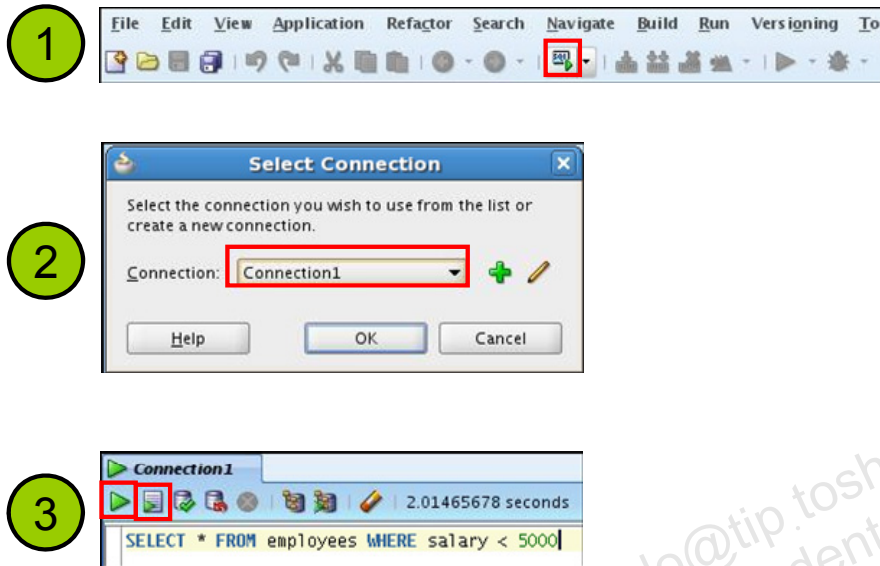
ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

After you create a database connection, you can use the Database Navigator to browse through many objects in a database schema, including tables, views, indexes, packages, procedures, triggers, and types.

You can see the definition of objects broken into tabs of information that is pulled out of the data dictionary. For example, if you select a table in the Navigator, details about columns, constraints, grants, statistics, triggers, and so on are displayed on an easy-to-read tabbed page.

Executing SQL Statements



ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

To execute a SQL statement, perform the following steps:

1. Click the Open SQL Worksheet icon.
2. Select the connection.
3. Execute the SQL command by clicking either of the following:
 - The **Execute statement** button or by pressing F9. The output is as follows:

The screenshot shows the 'Results' tab in Oracle SQL Developer. The output is a table with the following data:

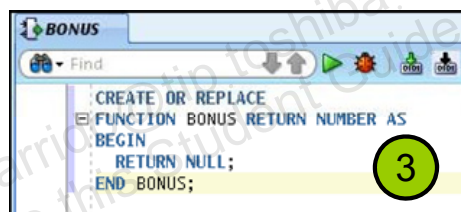
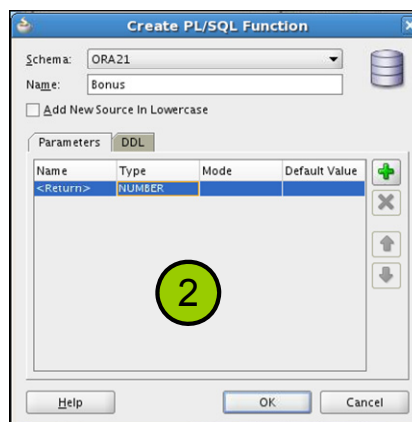
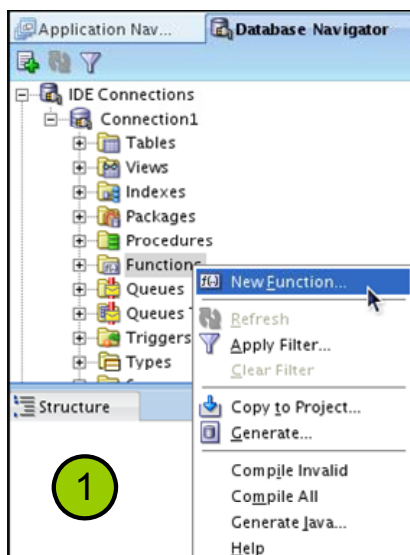
	EMPLOYEE_ID	FIRST_NAME	LAST_NAME
1	100	Steven	King
2	101	Neena	Kochhar

- The **Run Script** button or by pressing F5. The output is as follows:

The screenshot shows the 'Script Output' tab in Oracle SQL Developer. The output is a table with the following data:

EMPLOYEE_ID	FIRST_NAME	LAST_NAME
100	Steven	King

Creating Program Units



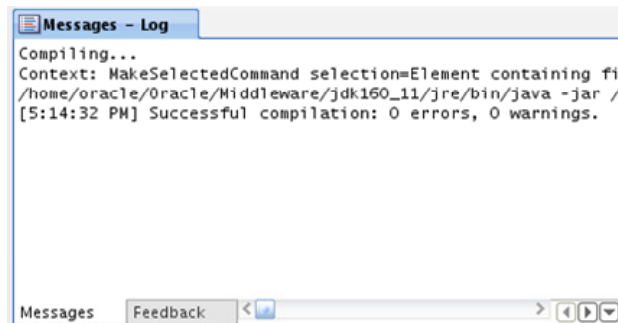
Skeleton of the function

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

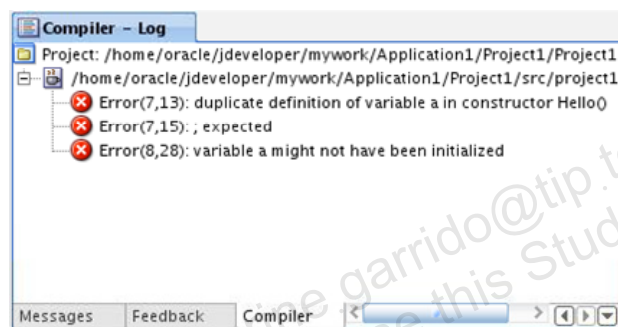
To create a PL/SQL program unit, perform the following steps:

1. Select View > Database Navigator. Select and expand a database connection. Right-click a folder corresponding to the object type (procedures, packages, or functions). Select "New [Procedures|Packages|Functions]".
2. Enter a valid name for the function, package, or procedure, and click OK.
3. A skeleton definition is created and opened in the Code Editor. You can then edit the subprogram to suit your need.

Compiling



Compilation with errors



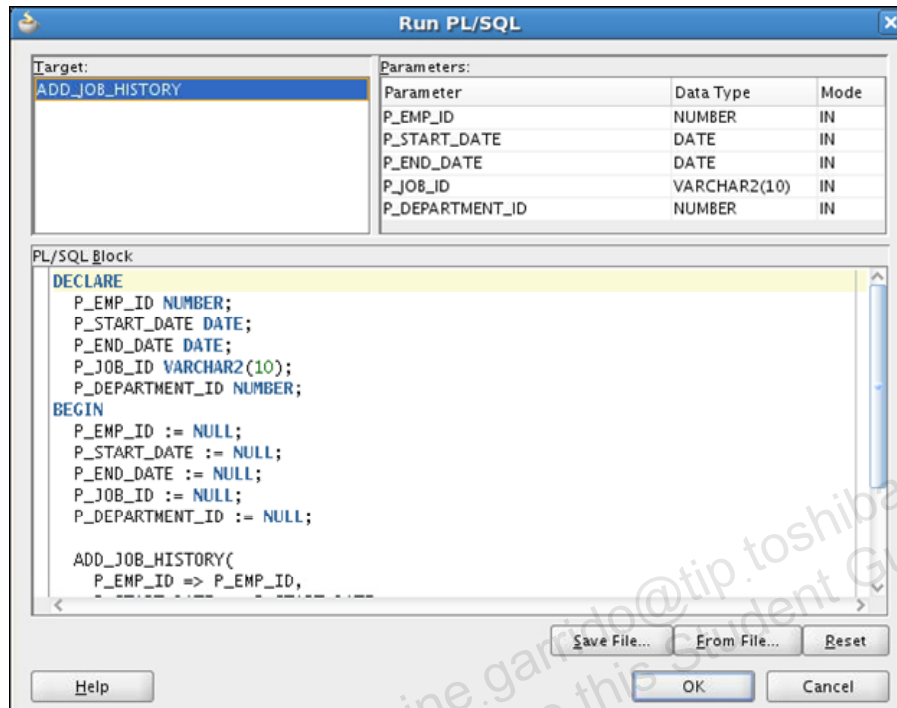
Compilation without errors

ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

After editing the skeleton definition, you need to compile the program unit. Right-click the PL/SQL object that you need to compile in the Connection Navigator, and then select Compile. Alternatively, you can press CTRL + SHIFT + F9 to compile.

Running a Program Unit

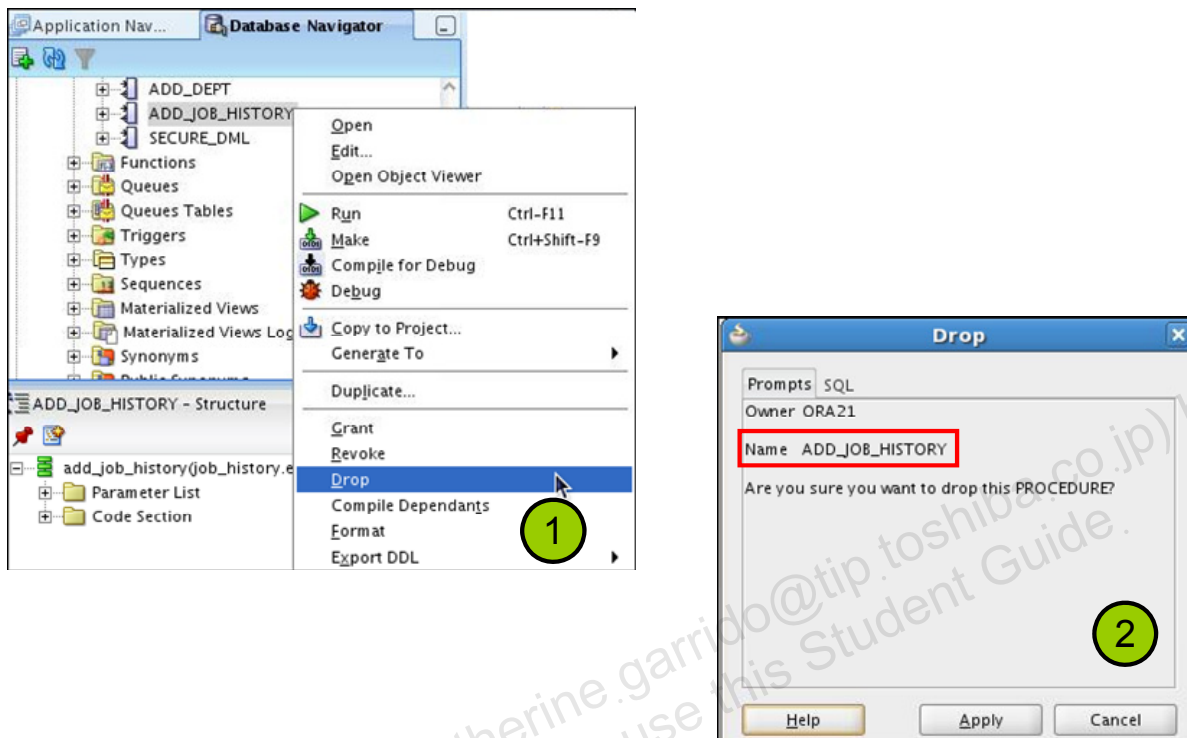


ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

To execute the program unit, right-click the object and select Run. The Run PL/SQL dialog box appears. You may need to change the NULL values with reasonable values that are passed into the program unit. After you change the values, click OK. The output is displayed in the Message-Log window.

Dropping a Program Unit

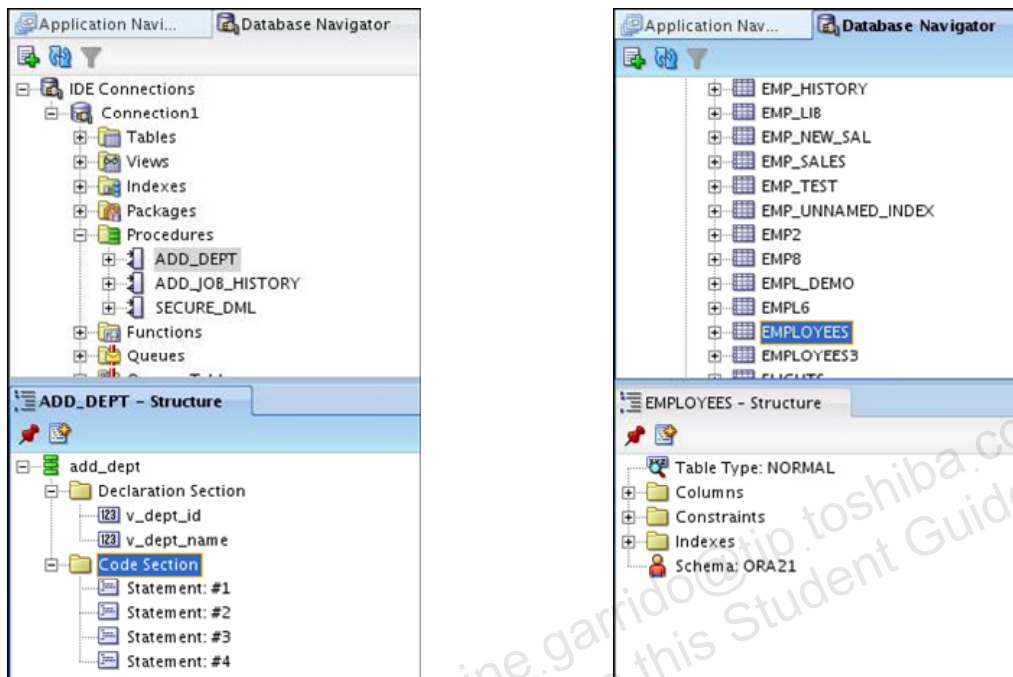


ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

To drop a program unit, right-click the object and select Drop. The Drop Confirmation dialog box appears. Click **Apply**. The object is dropped from the database.

Structure Window



ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

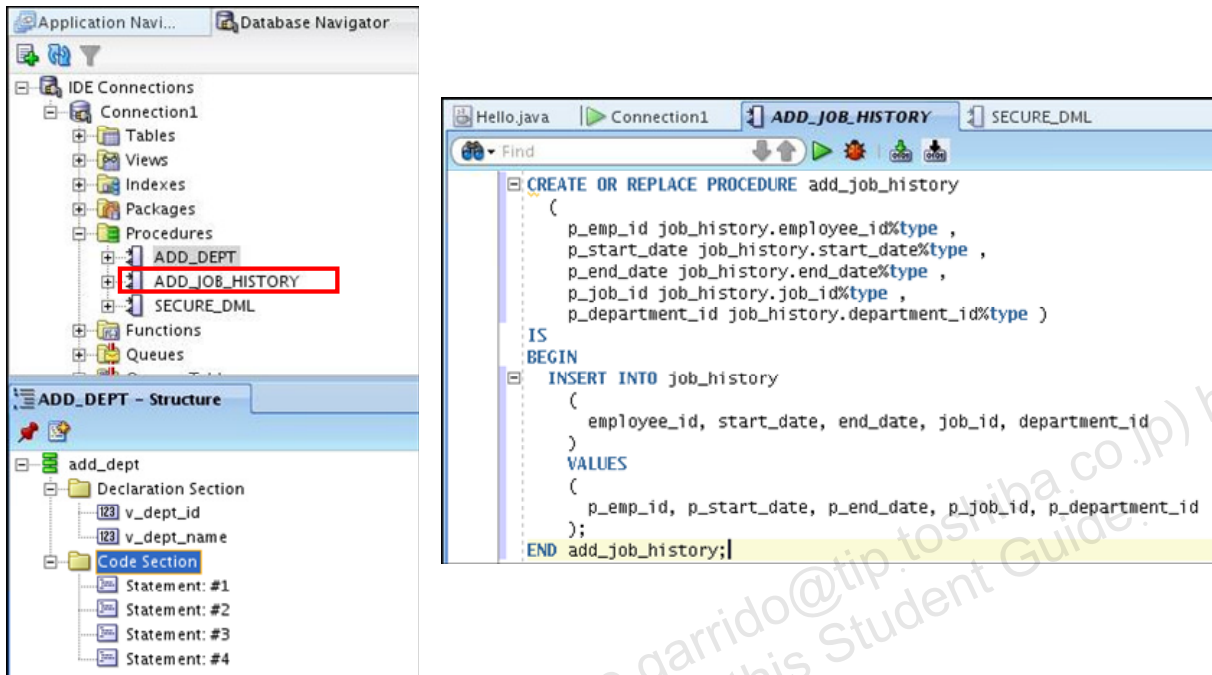
The Structure window offers a structural view of the data in the document that is currently selected in the active window of those windows that participate in providing structure: the navigators, the editors and viewers, and the Property Inspector.

Select View > Structure window to view the Structure window.

In the Structure window, you can view the document data in a variety of ways. The structures that are available for display are based on the document type. For a Java file, you can view code structure, UI structure, or UI model data. For an XML file, you can view XML structure, design structure, or UI model data.

The Structure window is dynamic, tracking always the current selection of the active window (unless you freeze the window's contents on a particular view), as is pertinent to the currently active editor. When the current selection is a node in the navigator, the default editor is assumed. To change the view on the structure for the current selection, click a different structure tab.

Editor Window



ORACLE

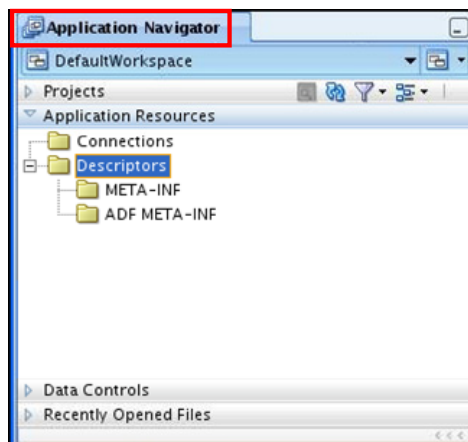
Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

When you double-click the name of a program unit, it opens in the Editor window. You can view all your project files in one single editor window, you can open multiple views of the same file, or you can open multiple views of different files.

The tabs at the top of the editor window are the document tabs. Clicking a document tab gives that file focus, bringing it to the foreground of the window in the current editor.

The tabs at the bottom of the editor window for a given file are the editor tabs. Clicking an editor tab opens the file in that editor.

Application Navigator



ORACLE

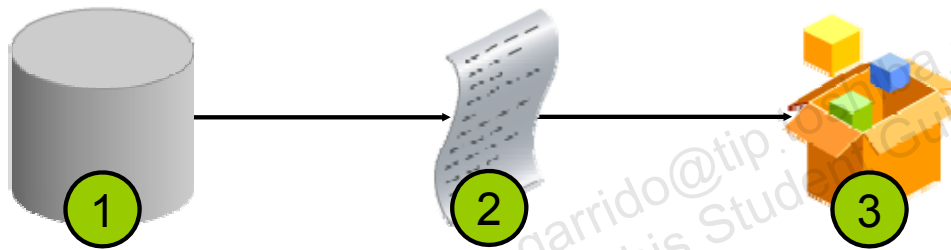
Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

The Application Navigator gives you a logical view of your application and the data that it contains. The Application Navigator provides an infrastructure that the different extensions can plug in to and use to organize their data and menus in a consistent, abstract manner. Although the Application Navigator can contain individual files (such as Java source files), it is designed to consolidate complex data. Complex data types such as entity objects, Unified Modeling Language (UML) diagrams, Enterprise JavaBeans (EJBs), or Web services appear in this navigator as single nodes. The raw files that make up these abstract nodes appear in the Structure window.

Deploying Java Stored Procedures

Before deploying Java stored procedures, perform the following steps:

1. Create a database connection.
2. Create a deployment profile.
3. Deploy the objects.



ORACLE

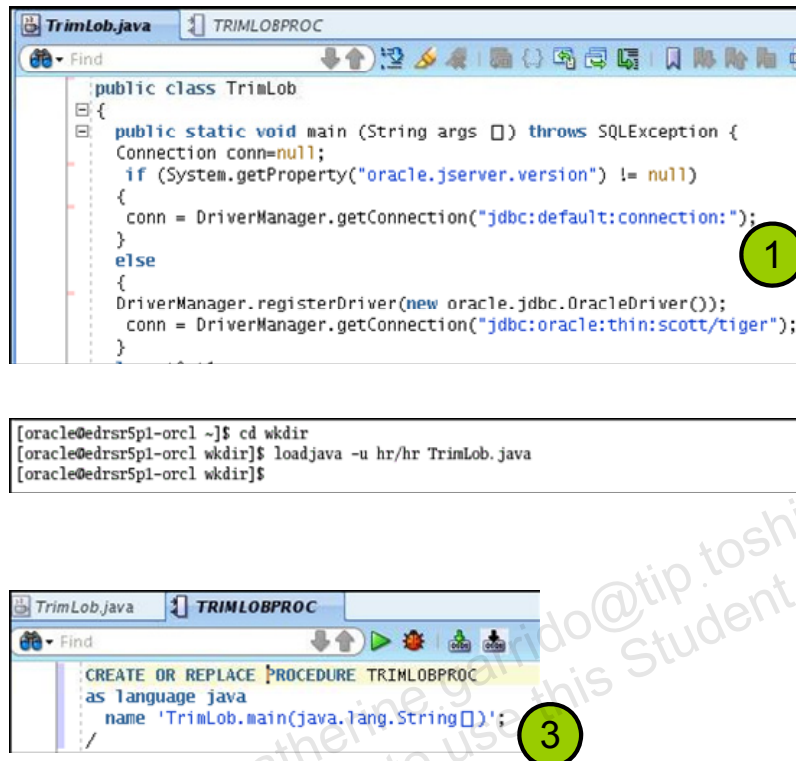
Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

Create a deployment profile for Java stored procedures, and then deploy the classes and, optionally, any public static methods in JDeveloper using the settings in the profile.

When the database is deployed, the information provided in the Deployment Profile Wizard and two Oracle Database utilities are used:

- `loadjava` loads the Java class containing the stored procedures to an Oracle database.
- `publish` generates the PL/SQL call-specific wrappers for the loaded public static methods. Publishing enables the Java methods to be called as PL/SQL functions or procedures.

Publishing Java to PL/SQL



ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.

The slide shows the Java code and illustrates how to publish the Java code in a PL/SQL procedure.

1. Create and compile a Java routine.
2. Execute the command shown in the slide to load the TrimLob Java routine into the database.
3. Publish the Java class method by creating the TrimLobProc PL/SQL call specification that references the Java class methods. Within the TrimLobProc PL/SQL specification, you identify the name of the Java class method and its parameters. This is called "Publishing" the Java class method.

For more details on how to run Java routines in the database, refer to the following OBE:
http://www.oracle.com/technology/obe/08/11gR1_JDBC_Java/java_otn.htm

How Can I Learn More About JDeveloper 11g?

Topic	Website
Oracle JDeveloper Product Page	http://www.oracle.com/technology/products/jdev/index.html
Oracle JDeveloper 11g Tutorials	http://www.oracle.com/technology/obe/obe11jdev/11/index.html
Oracle JDeveloper 11g Product Documentation	http://www.oracle.com/technology/documentation/jdev.html
Oracle JDeveloper 11g Discussion Forum	http://forums.oracle.com/forums/forum.jspa?forumID=83

ORACLE

Copyright © 2012, Oracle and/or its affiliates. All rights reserved.