Table of Contents

[CHAPTER 26: SQL DEVELOPER 2](#_Toc417706364)

[Theory 2](#_Toc417706365)

[AIM 3](#_Toc417706366)

[Lab Exercise 26: SQL DEVELOPER 4](#_Toc417706367)

[1. Installing SQL Developer 5](#_Toc417706368)

[2. Connecting to Database 16](#_Toc417706369)

[3. Exploring SQL Developer 25](#_Toc417706370)

[4. Debugging subprograms 41](#_Toc417706371)

[SUMMARY 57](#_Toc417706372)

[REFERENCES 58](#_Toc417706373)

[INDEX 59](#_Toc417706374)

# CHAPTER 26: SQL DEVELOPER

## Theory

SQL\*Plus is a powerful tool to connect and work with Oracle Database. However, it requires a lot of typing and, in some cases, frustrating. For example, you have to type and run several queries just know all Functions you have and what the source code of your selected method is. Moreover, it is difficult to some higher-level tasks such as debugging methods and building ER diagram of your database. For all of that and for other reasons, it is useful to consider a graphical development tool to develop PL/SQL code. Fortunately, Oracle presents its free tool: Oracle SQL Developer.

Oracle SQL Developer is developed not only for PL/SQL developers; it is comprehensive tool than includes very good features for SQL Developers and Oracle DBAs. It is possible also to use Oracle SQL Developer to connect to other databases (non-Oracle vendors).

The features of Oracle SQL Developer can be categorized into three main categories:

1. Development: Develop, test, debug, tune, report, ...
2. Administration: backup, diagnoses, scheduling, migration, ...
3. Designing: data modeler.

## AIM

The AIM of the following exercise is to be able to use Oracle SQL Developer.

The steps involved will include:

* Installing SQL Developer
* Connecting to Database
* Exploring SQL Developer
* Debugging Subprograms

In general, lab exercises are done in sequential order. Thus, it is assumed that you successfully completed the previous labs. However, not all previous labs are required. Please be sure to run the following lab before proceeding:

* Installing Oracle Database 12c.

Estimated Completion Time:

30 minutes

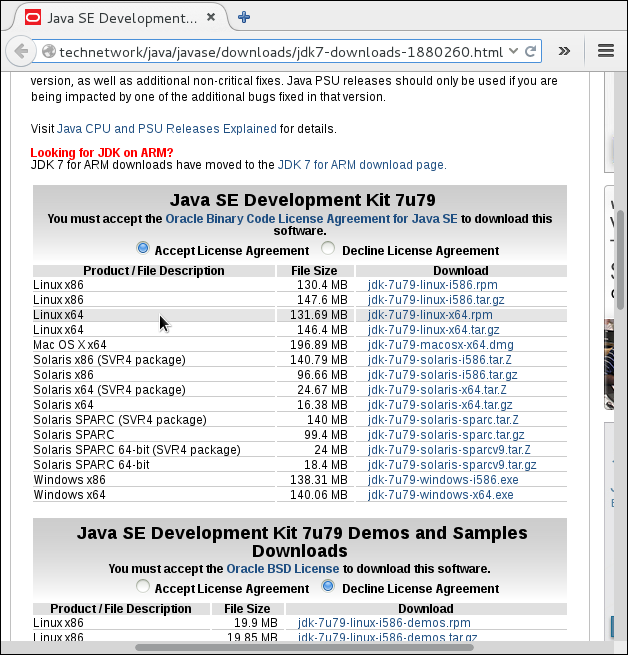
# Lab Exercise 26: SQL DEVELOPER

|  |
| --- |
|  |

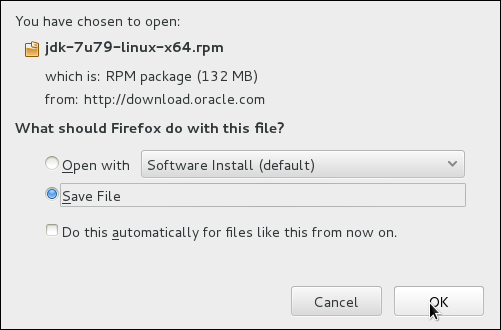
## Installing SQL Developer

**Step 1:** You will first need to install JDK. Open the internet browser and enter the following link. Accept the License Agreement and download "**jdk-7u79-linux-x64.rpm**"

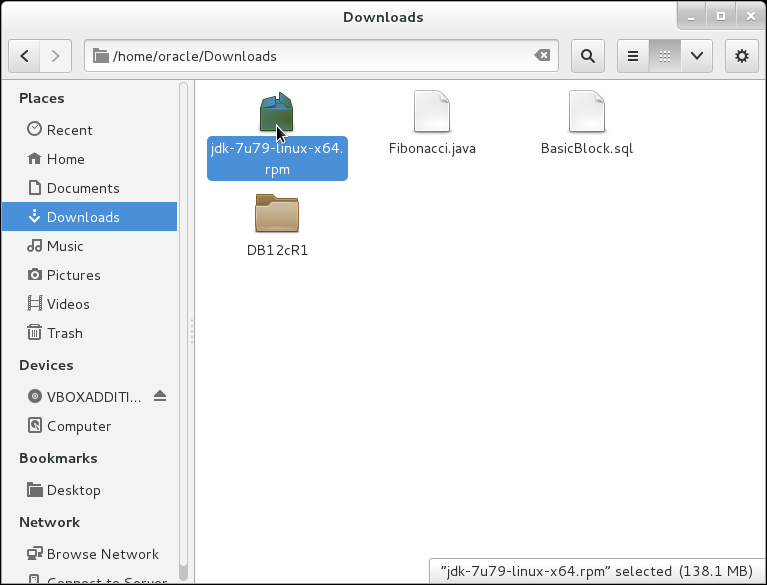
|  |
| --- |
| line |
| http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html |

****

**Step 2:** Select "**Save file**" and click "**OK**":

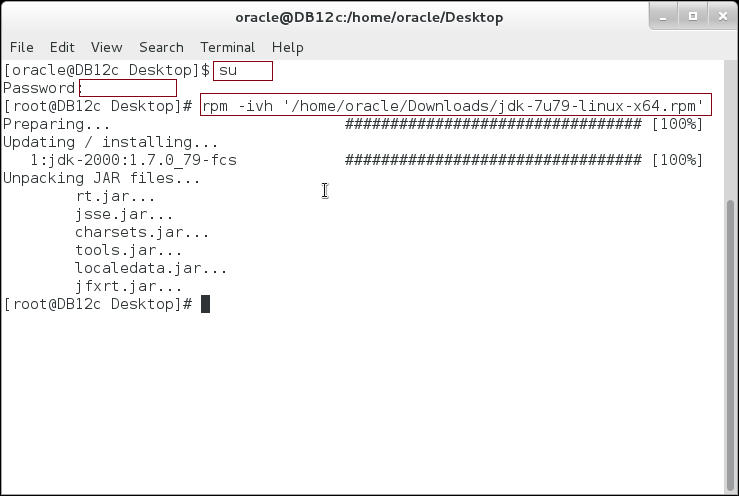
****

**Step 3:** By default, the file is located in a download folder:

****

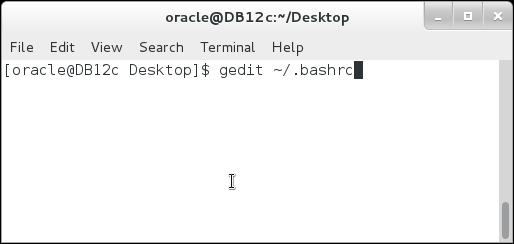
**Step 4:** Become root and use rpm package manager to install JDK as the following:

|  |  |
| --- | --- |
| Command | Description |
| su | switch user |
| oracle321 | root password |
| rpm -ivh '/home/oracle/Downloads/jdk-7u79-linux-x64.rpm' | Install JDK |



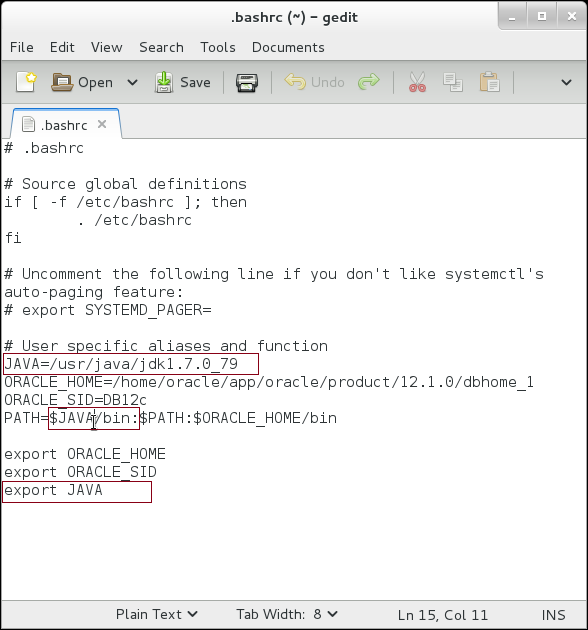
**Step 5:** Set the JAVA path as the following:

|  |  |
| --- | --- |
| Command | Description |
| gedit ~/.bashrc |  |

****

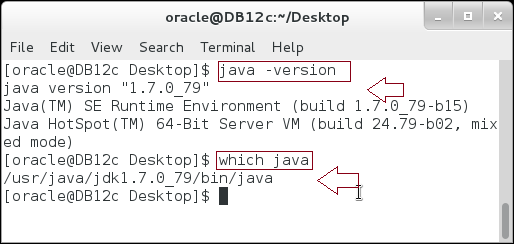
**Step 6:** Set the JAVA environment variable and add JAVA/bin to PATH as shown:

|  |  |
| --- | --- |
| Command | Description |
| JAVA=/usr/java/jdk1.7.0\_79 |  |
| PATH=$JAVA/bin:$PATH:$ORACLE\_HOME/bin |  |
| export JAVA |  |

****

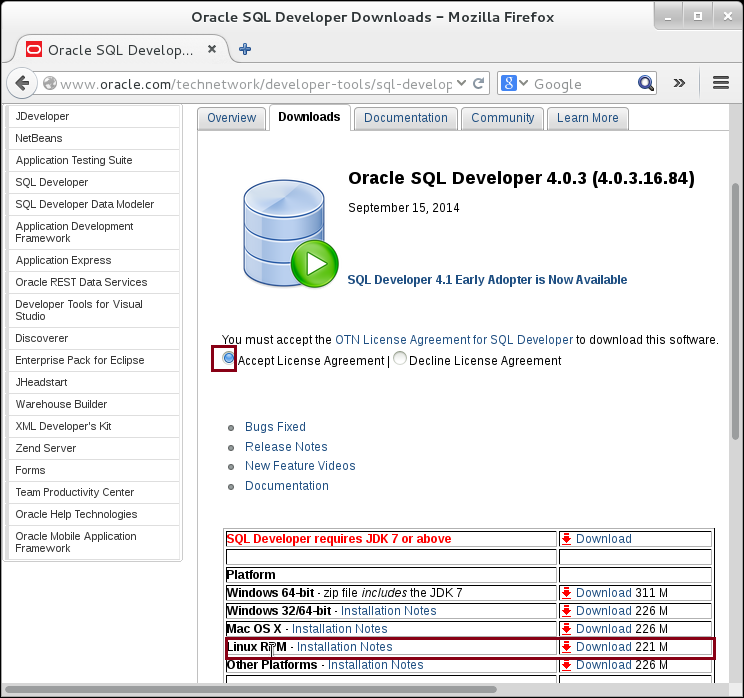
**Step 7:** Re-open the terminal and test if the JAVA path has been test correctly:

|  |
| --- |
| Command |
| java -version |
| which java |

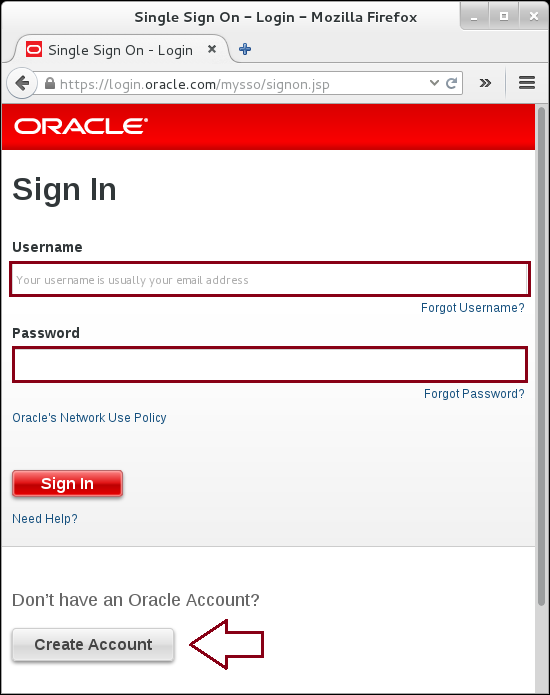
****

**Step 8:** Download "Oracle SQL Developer" from the following link. Accept the license and download "Linux RPM" :

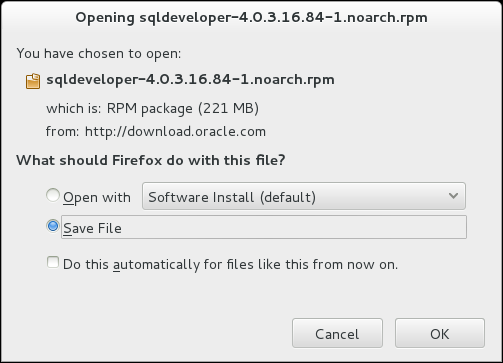
|  |
| --- |
| line |
| http://www.oracle.com/technetwork/developer-tools/sql-developer/downloads/index.html |

****

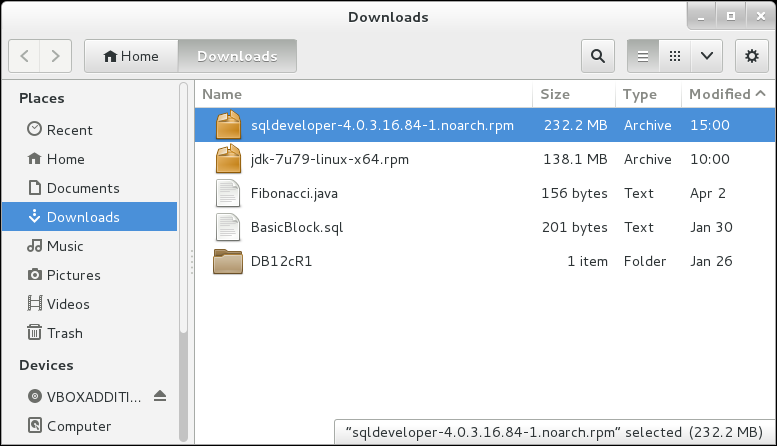
**Step 9:** Login with your user account or create one if you don't have:

****

**Step 10:** Download the file:

****

**Step 11:** By default, it is located in the Downloads directory:

****

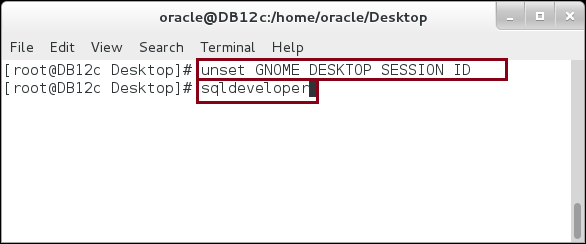
**Step 12:** Open the terminal and install SQL Developer:

|  |
| --- |
| line |
| su |
| oracle321 |
| rpm -Uhv '/home/oracle/Downloads/sqldeveloper-4.0.3.16.84-1.noarch.rpm' |

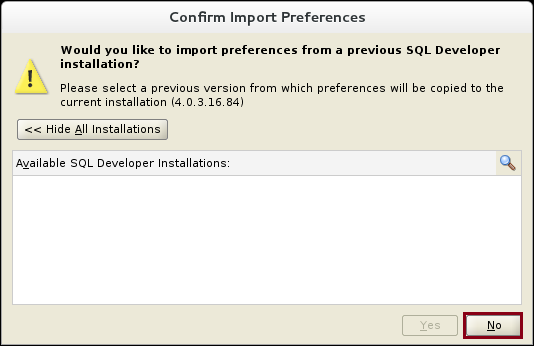


**Step 13:** Start the SQL Developer by the following command:

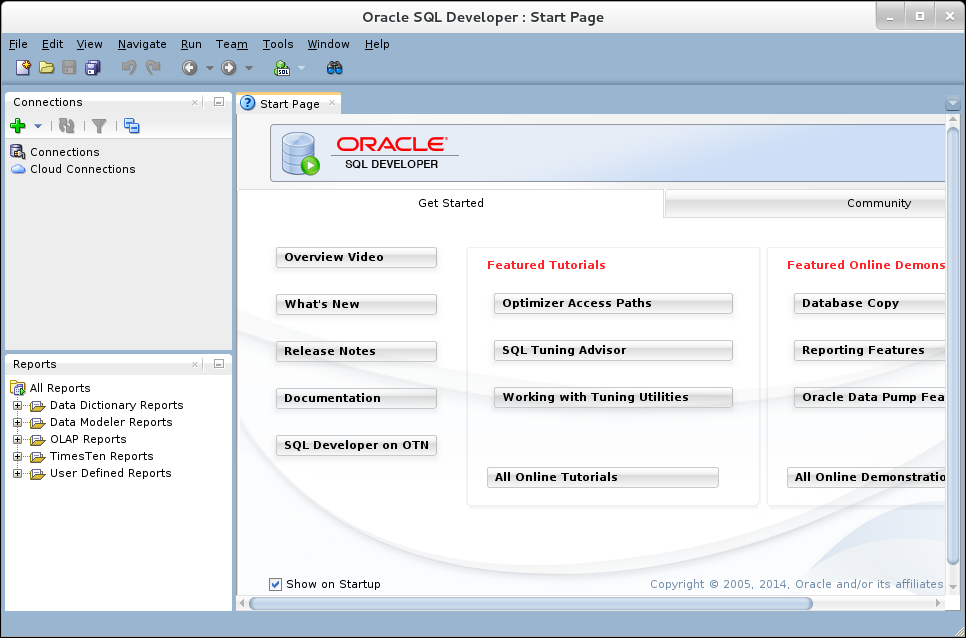
|  |
| --- |
| Command |
| unset GNOME\_DESKTOP\_SESSION\_ID |
| sqldeveloper |



**Step 14:** If the wizard ask you to import preference from other application, click "No".

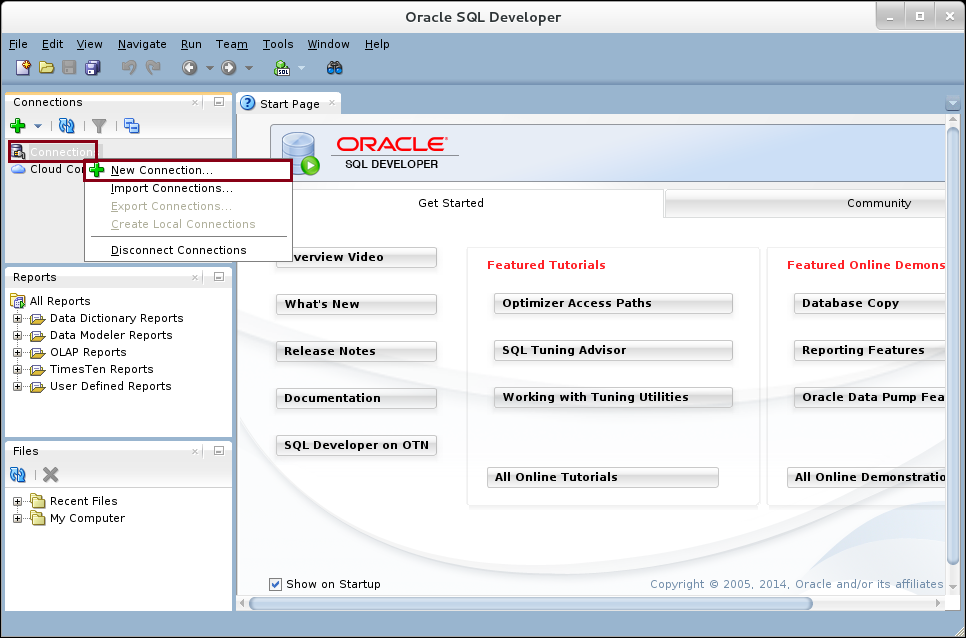


**Step 15:** Finally, here we are. Oracle SQL Developer is running.



## Connecting to Database

**Step 1:** Before starting this section, be sure that your database is up and running and the listener is configured (as in Chapter 1), and also up and running. In this section, you will learn that you have many options to do the same job. In our case, create a connection to the database. Right click on "**Connections**" list element and click on "**New Connection**" as show:

****

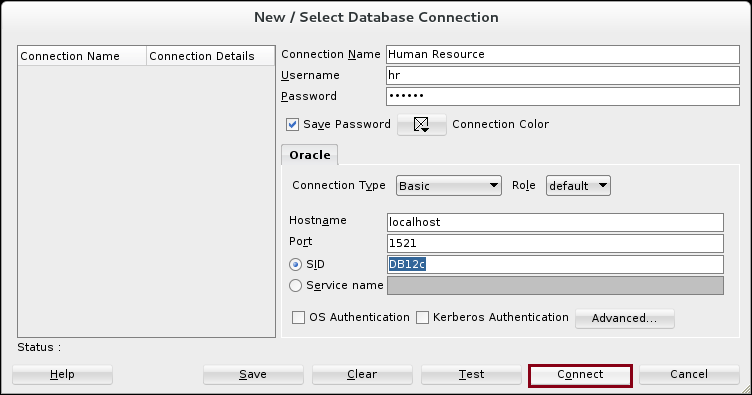
**Step 2:** Fill in the form as shown below:

|  |  |
| --- | --- |
| Field | Value |
| Connection name | Human Resource |
| username | hr |
| password | oracle |
| Save password | CHECKED |
| Host name | localhost |
| Port | 1251 |
| SID | DB12c |

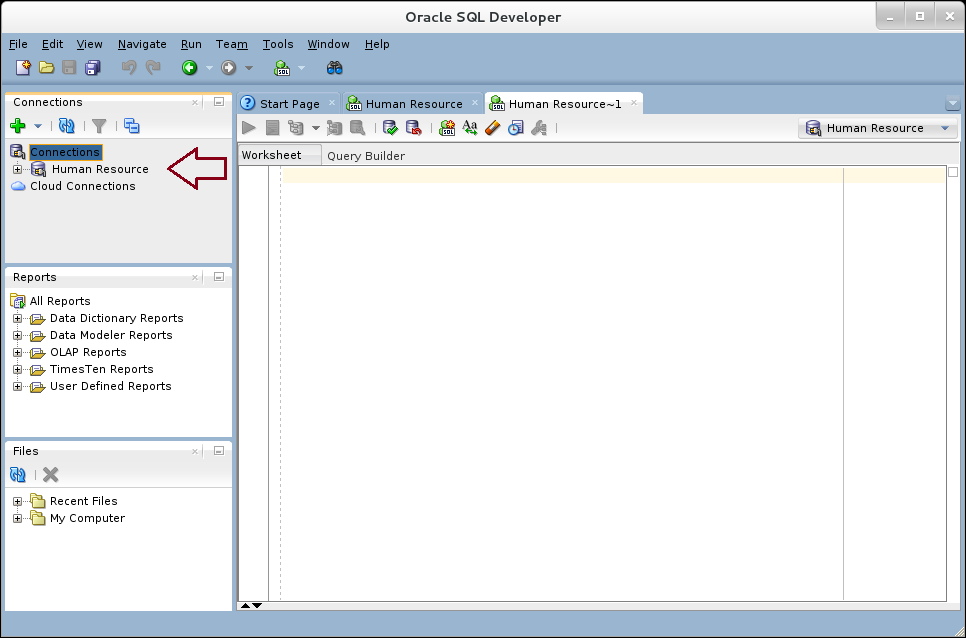
Click on "**Test**" button to test the connection.



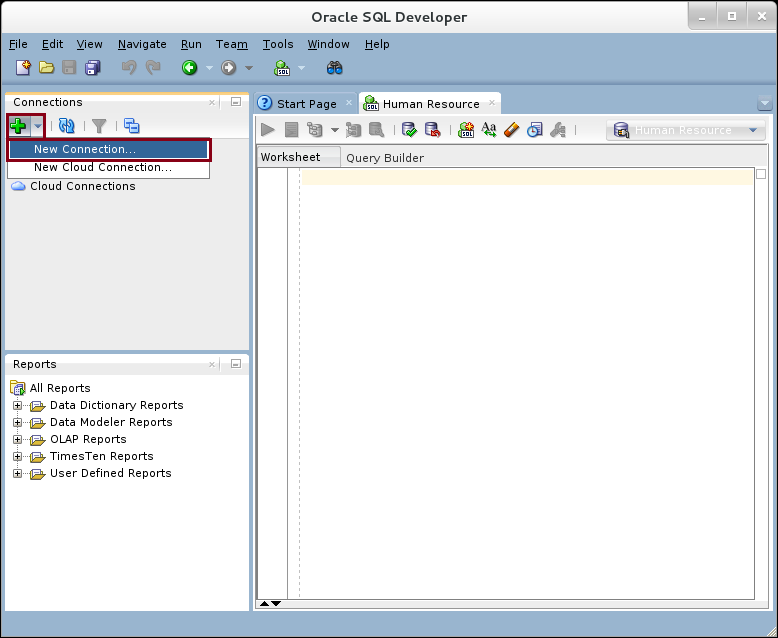
**Step 3:** If no error showed up after clicking on "**Test**" button, you can move forward and click on "**Connect**" button to create the connection.

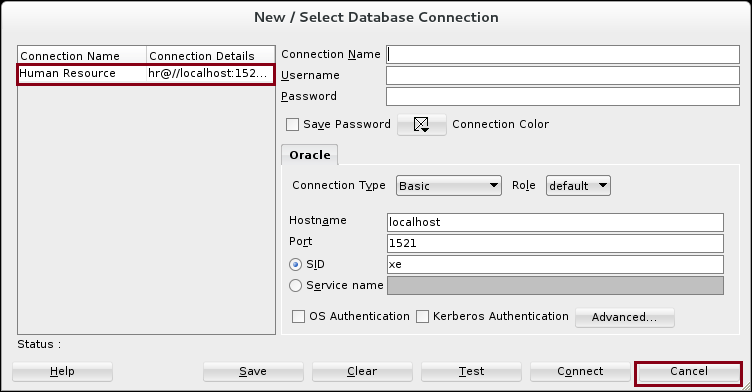
****

**Step 4:** The connection now appears on the connections list:

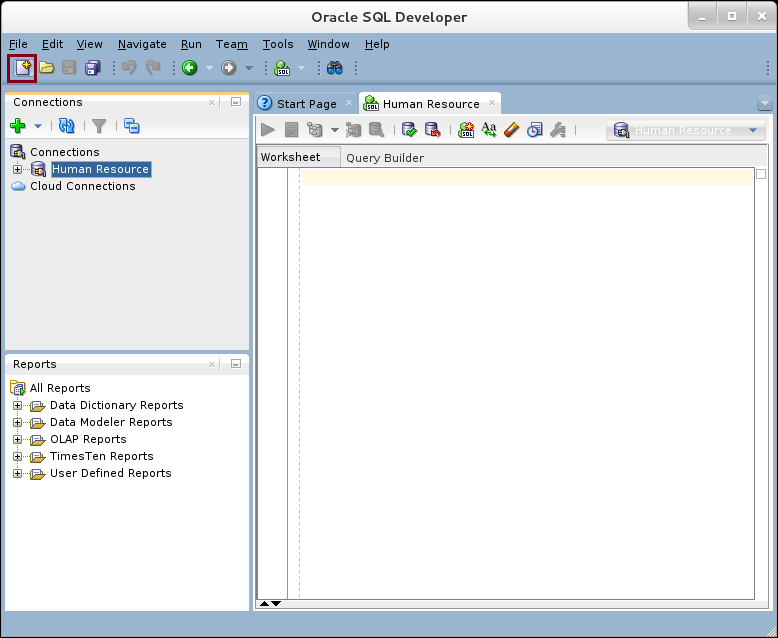


**Step 5:** Now, you will explore many other options to do the same as the previous task just to be familiarized with Oracle SQL Developer. Click on "**+**" icon in "**Connections**" window and then click "**New Connection**".

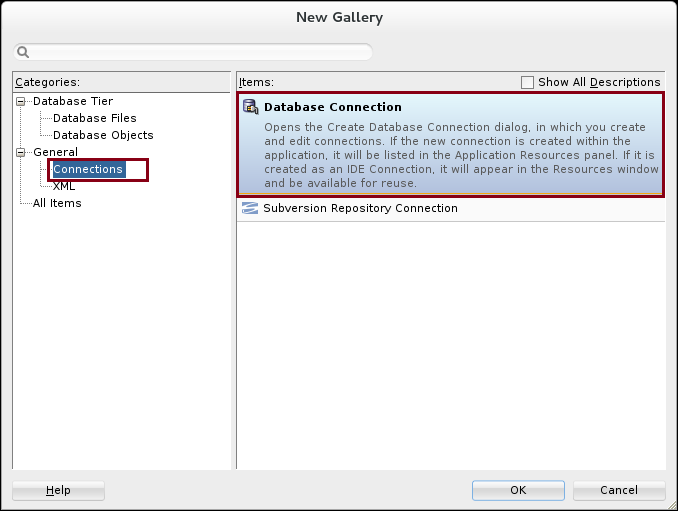


**Step 6:** You can create new connection as well. Click "**Cancel**" button.

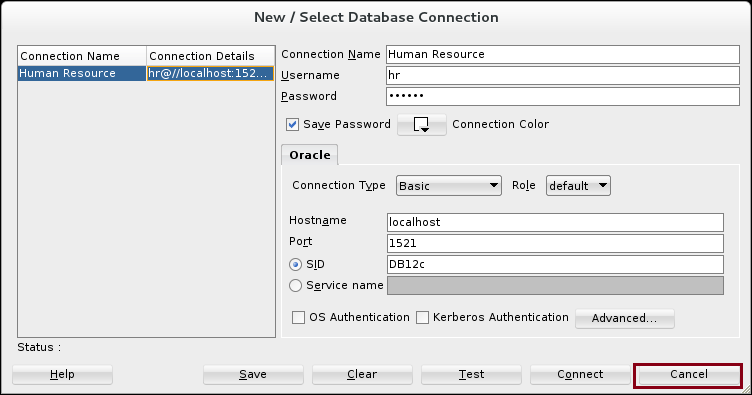
**Step 7:** Click on "**NEW**" icon in the icons menu.



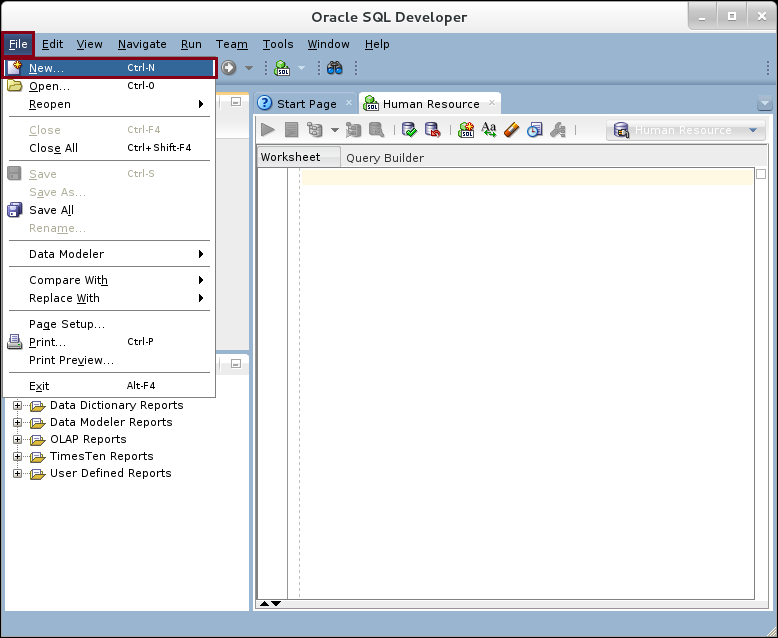
**Step 8:** Under "**General**" list item, click on "**Connections**". Double click on "**Database connections**" item list.



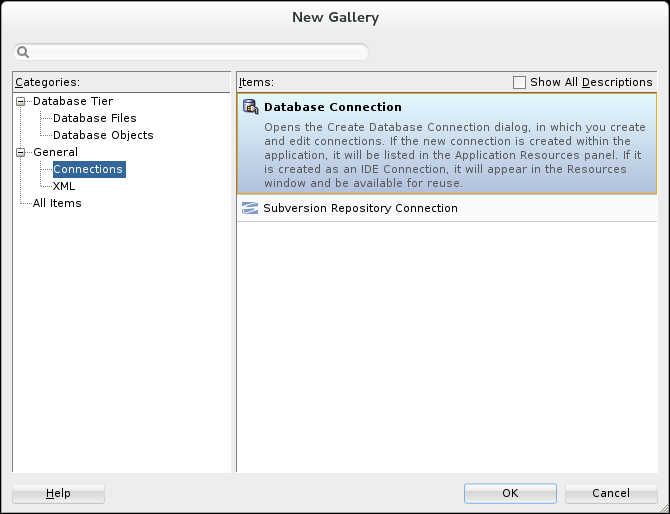
**Step 9:** Here you are, you can create new connection. Click on "**Cancel**" button.



**Step 10:** Expand on "**File**" menu and click on "**New**" menu list.

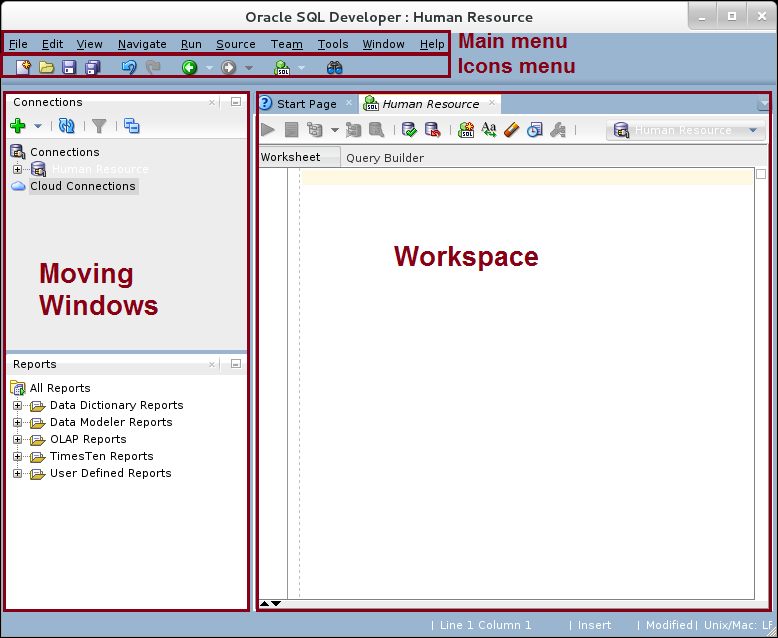
****

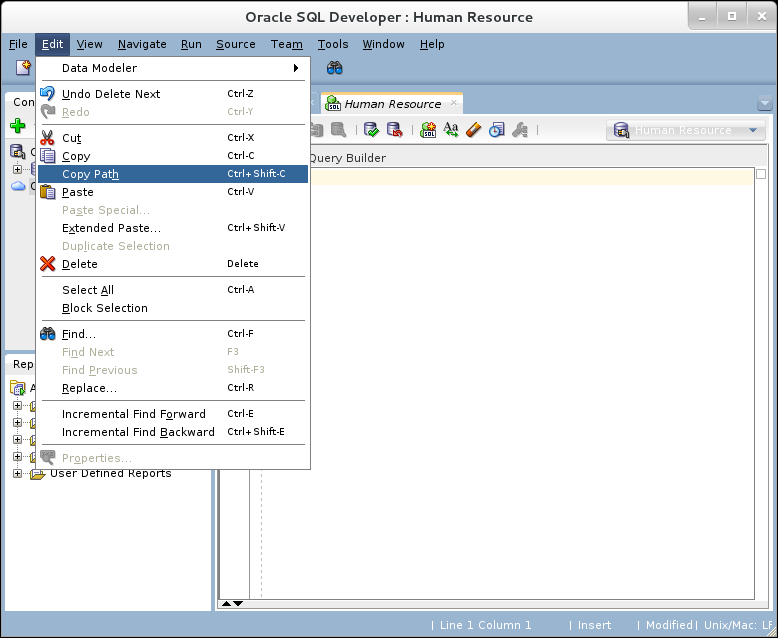
**Step 11:** Again, you can create new connection.



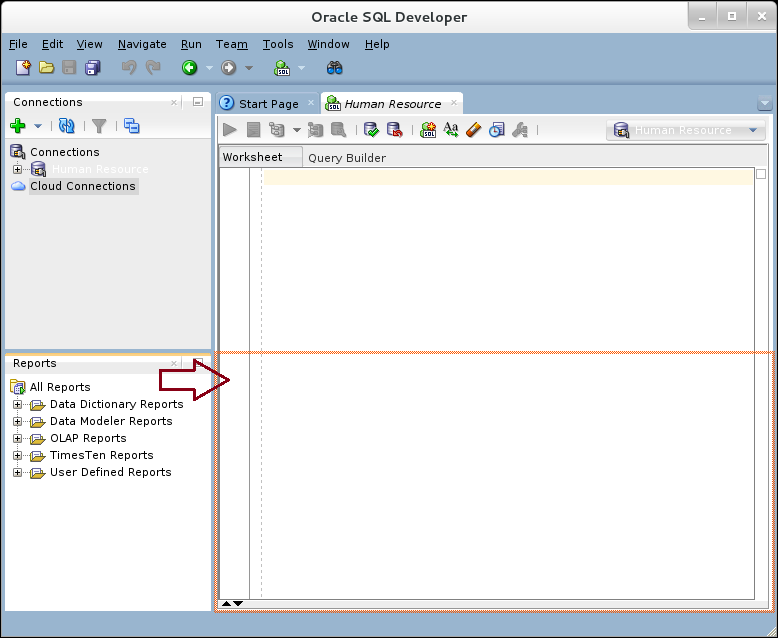
## Exploring SQL Developer

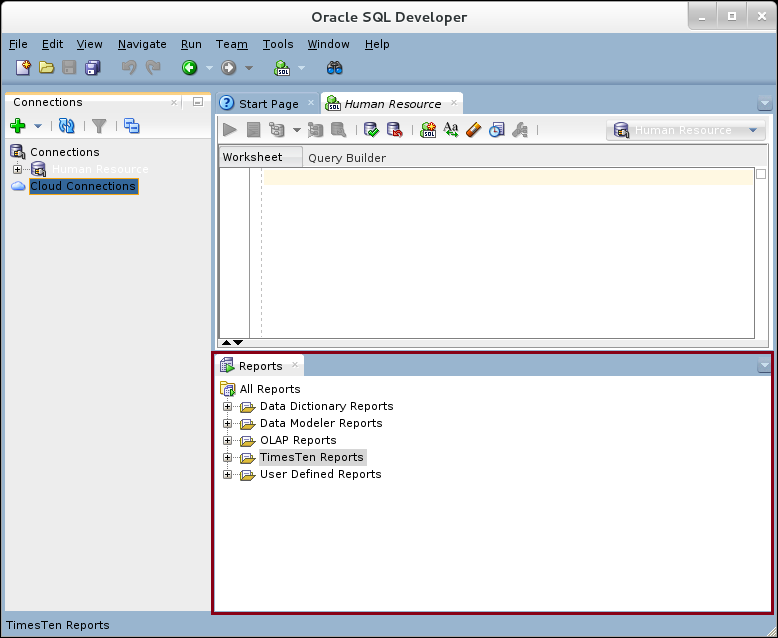
**Step 1:** SQL Developer interface consists of four main categories: Main menu, icons menu, moving windows, and workspace.



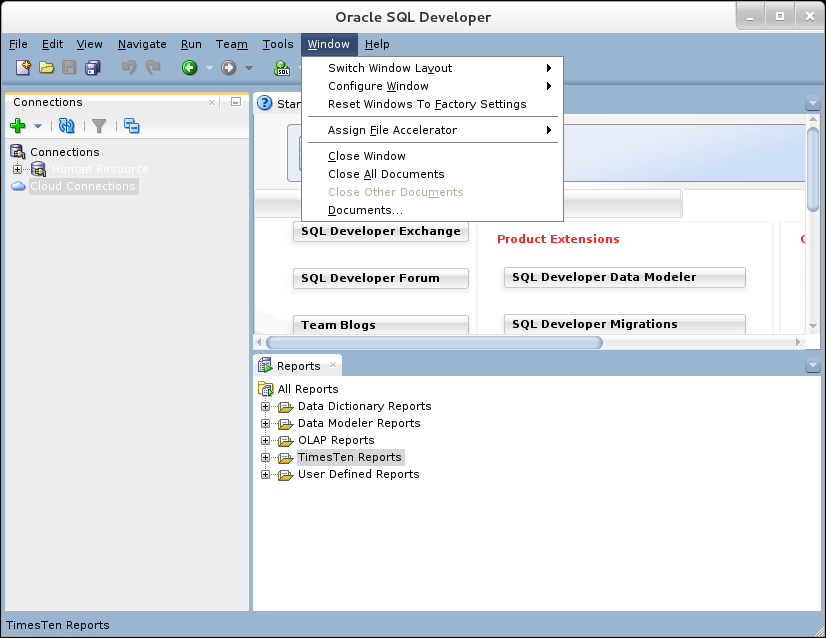
**Step 2:** The main method of navigation is the main menu. Take three minutes to familiarize yourself with each menu list. 

**Step 3:** You mainly use the moving windows to view the common items you regularly use. You can also customize your windows location by dragging and dropping them.

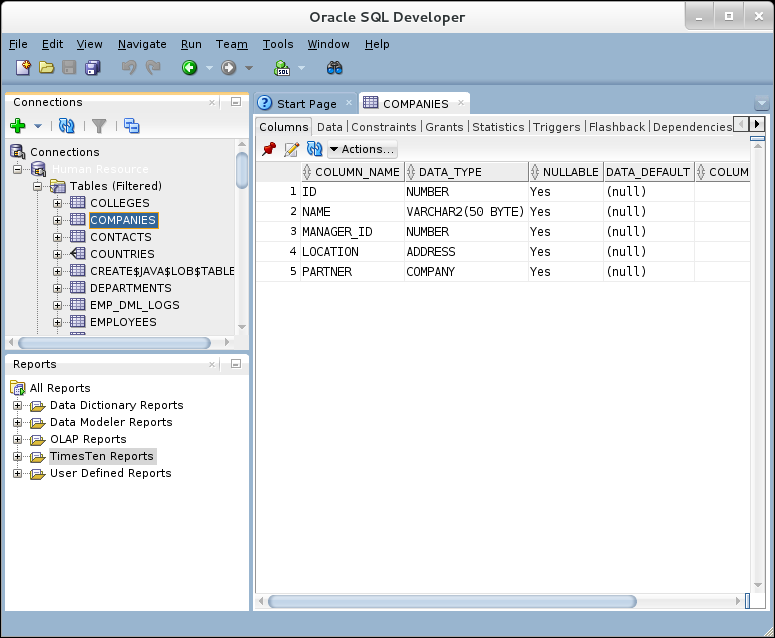
****

****

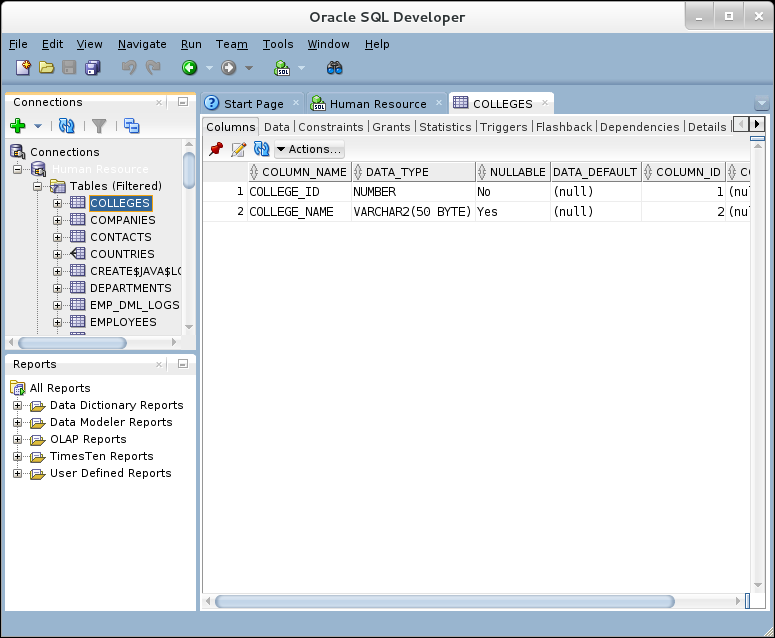
**Step 4:** Use "Windows" main menu to re arrange windows**:**



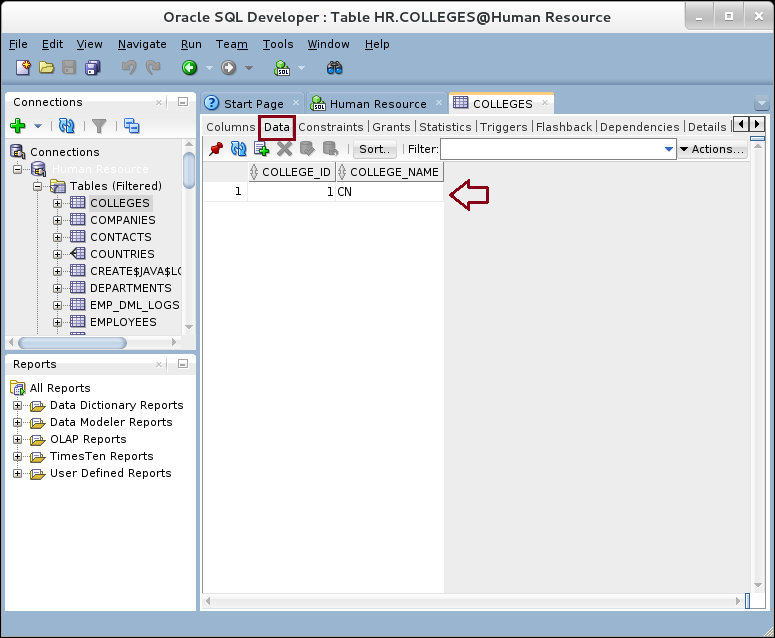
**Step 5:** To view "HR" schema objects, expand "Human Resource" connection. Expand "Tables" list item. Click on "COMPANIES" for example. The workspace area should show "COMPANIES" table's column.



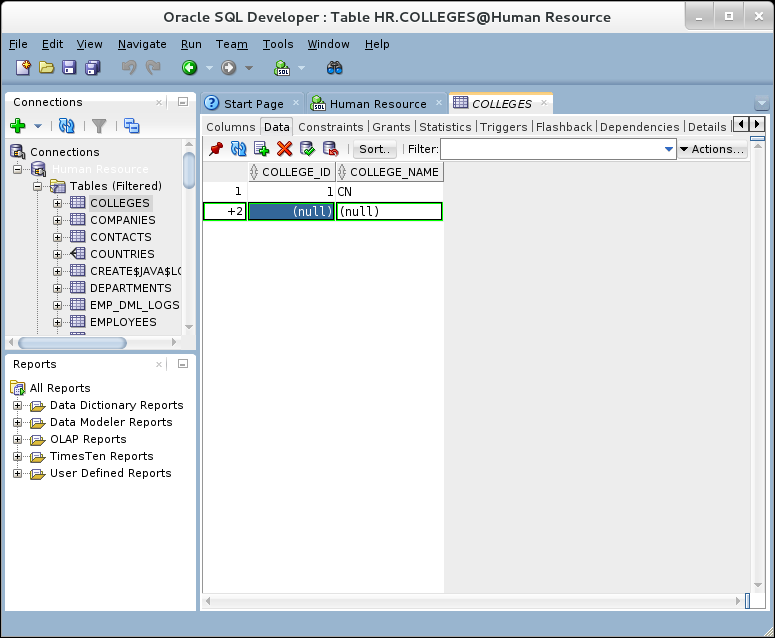
**Step 6:** Navigate through others objects, for example "COLLEGES" table**.**



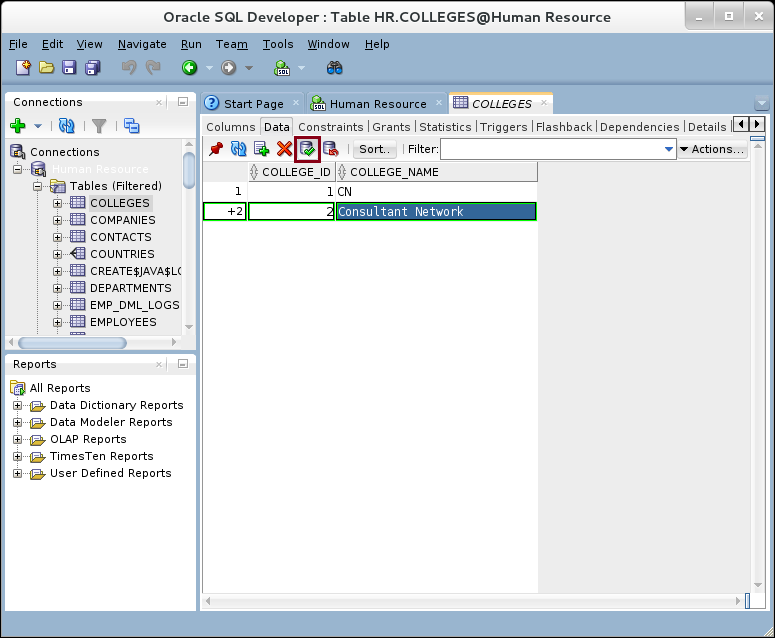
**Please note:** It is very easy to navigate through "schema" objects and view their structures in comparison with SQL\*Plus.

**Step 7:** You can also view the table data, constraints, accessing table privileges were granted to whom, which other objects that depends on this table and many other things using the tab pages as shown below:

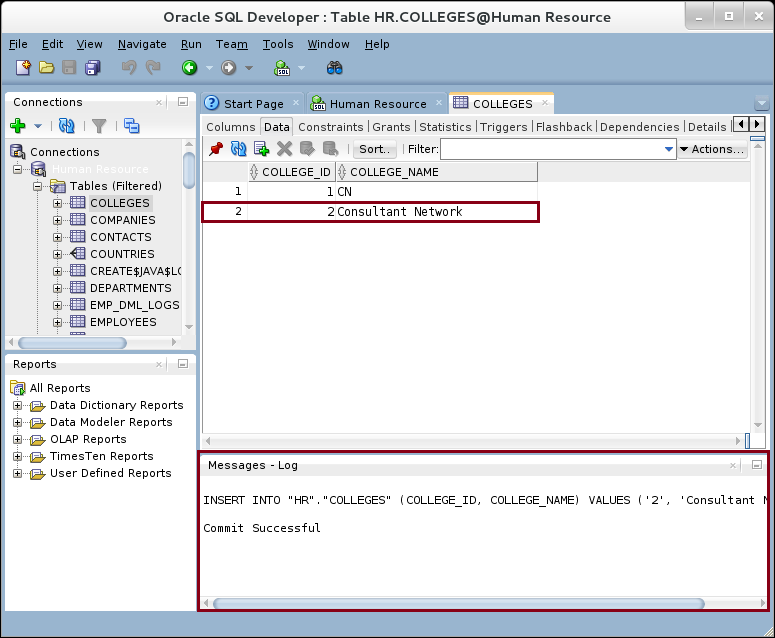
**Step 8:** You can also use "Data" tab page to insert new record into the table

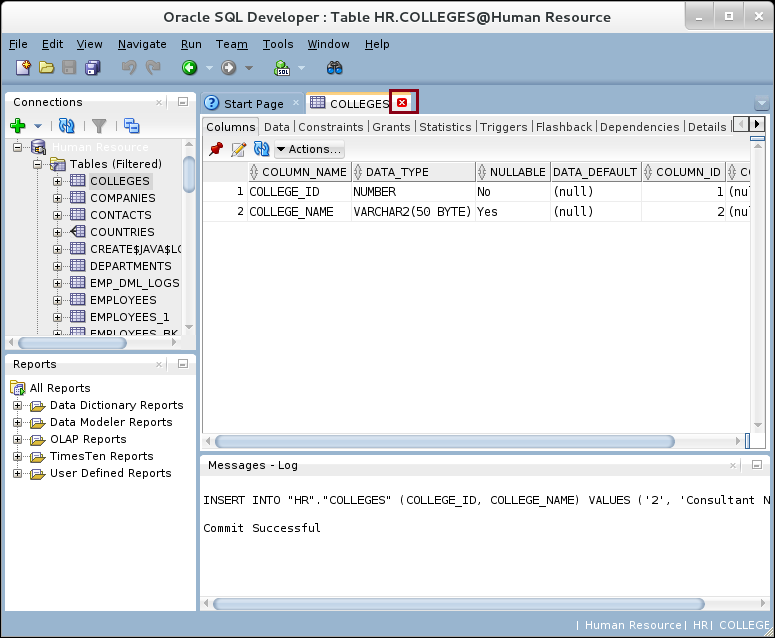


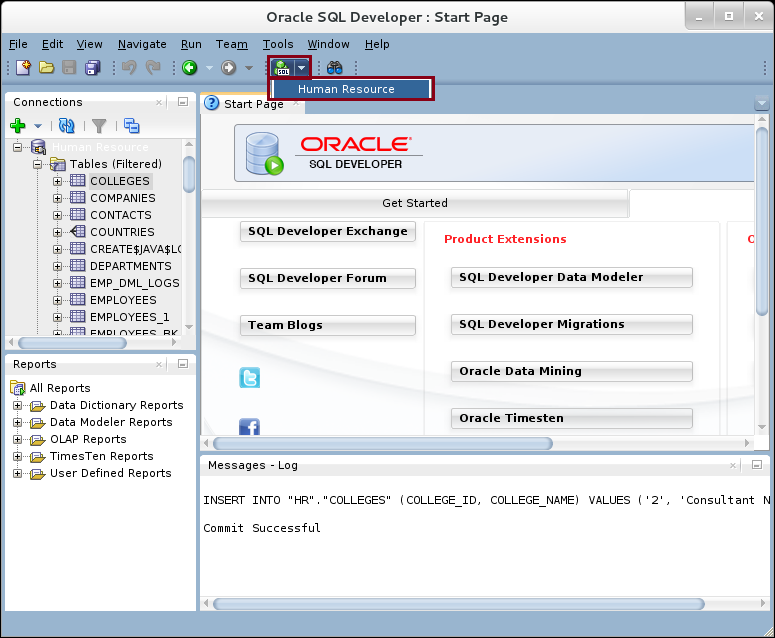
**Step 9:** Use "**Commit**" icon button to commit the transaction as shown



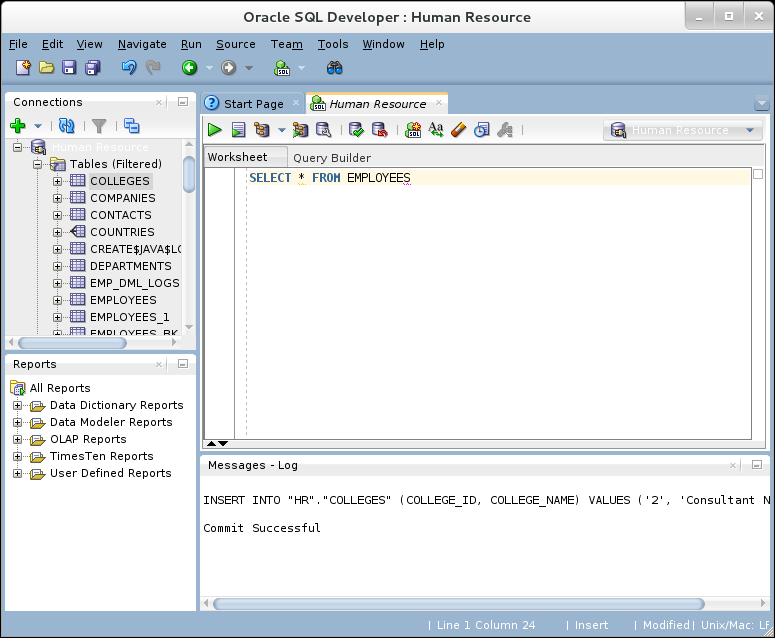
**Step 10:** You will see the SQL command used to insert the record into the table into the message log window:

****

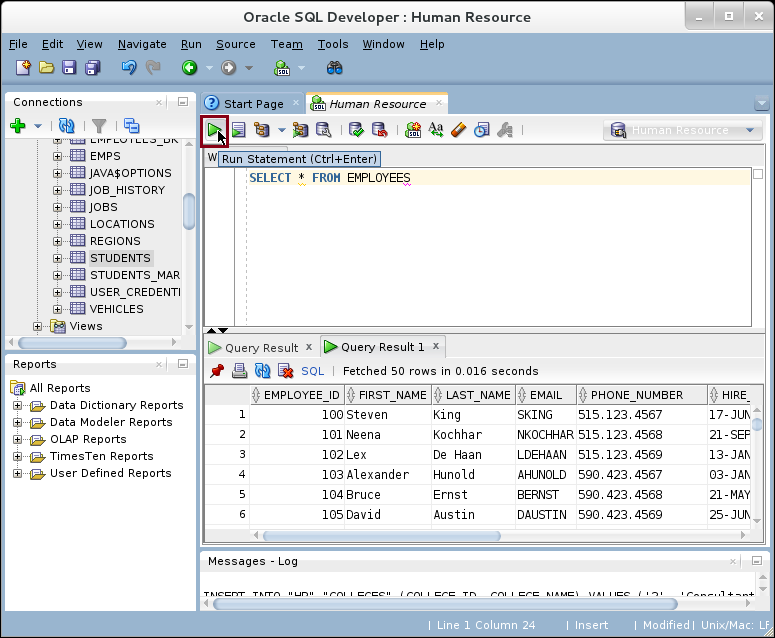
**Step 11:** If you no longer need to view "COLLEGES" table, close its tab page as shown below**.**

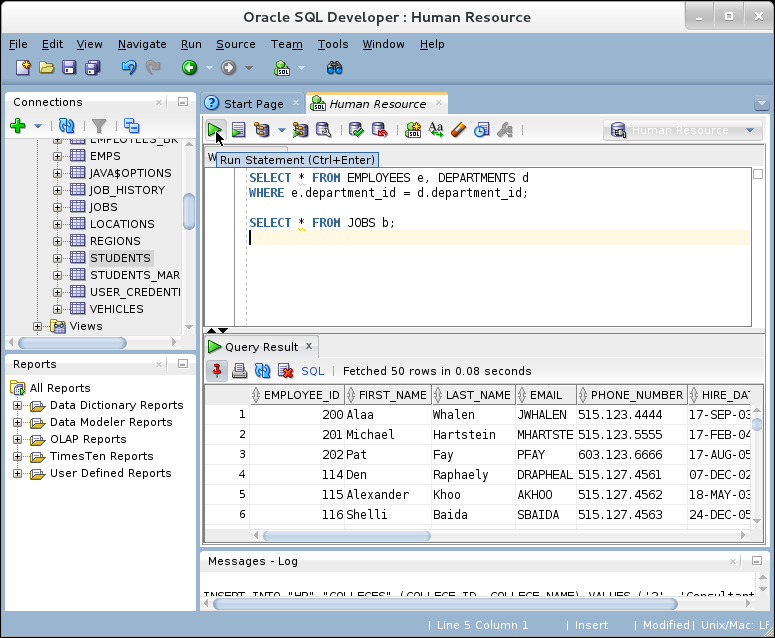
**Step 12:** You still have the power to write your own SQL. Expand on "**SQL Worksheet**" icon and select the connection "**Human Resource**"

**Step 13:** Type your SQL commands. For example "SELECT \* FROM EMPLOYEES". Please note the editor inspector recommendation.



**Step 14:** Use "Run" icon to execute the query.

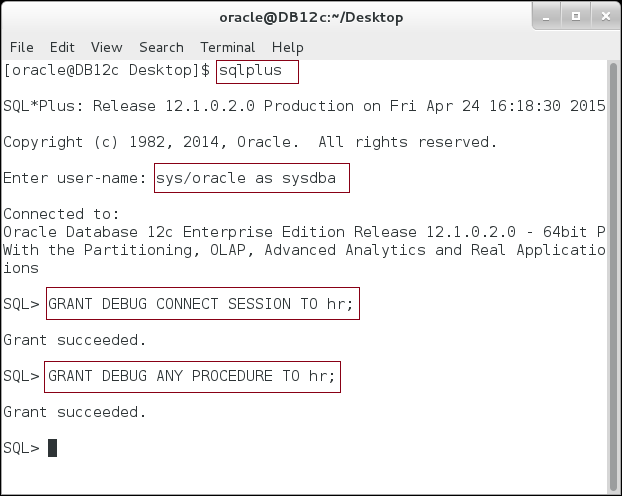


**Step 15:** You can write as many SQL statements as you want. Each SQL statement should end by ";". To execute one of the SQL statement, select it first and click on "Run" icon. 

## Debugging subprograms

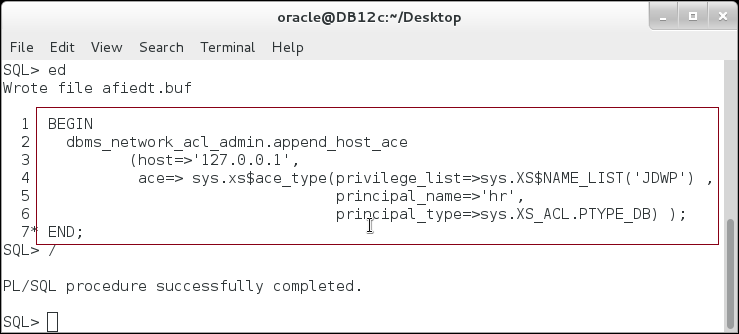
**Step 1:** Before start debugging any subprogram, you must grant DEBUG CONNECT SESSION and DEBUG ANY PROCEDURE privileges to the user. Connect as "sys" and grant "HR" these privileges as shown below:

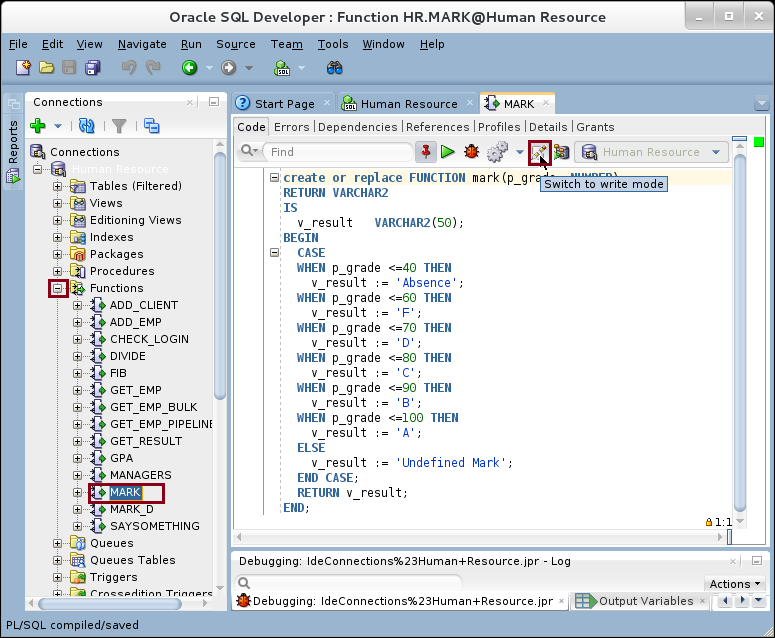
|  |  |
| --- | --- |
| Command | Description |
| sqlplus | Run SQL |
| sys/oracle as sysdba | Connect as sys |
| GRANT DEBUG CONNECT SESSION TO hr; |  |
| GRANT DEBUG ANY PROCEDURE TO hr; |  |

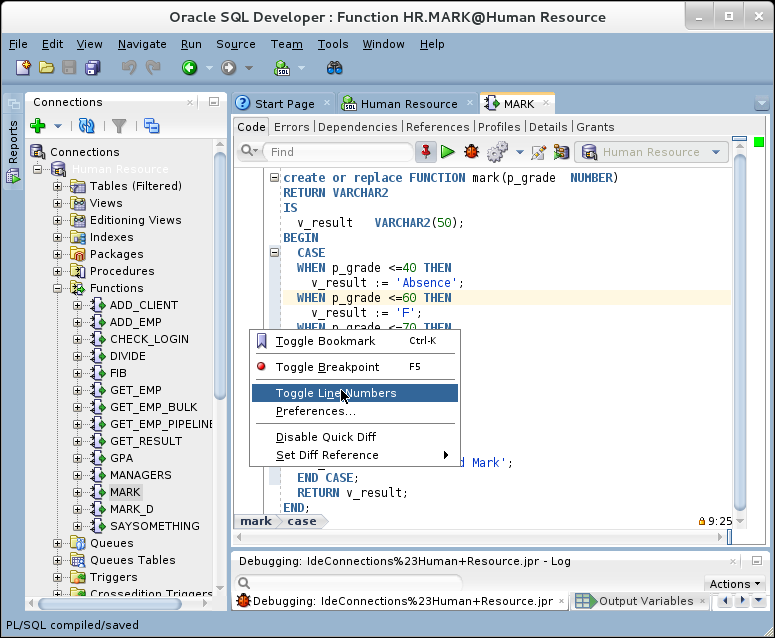
****

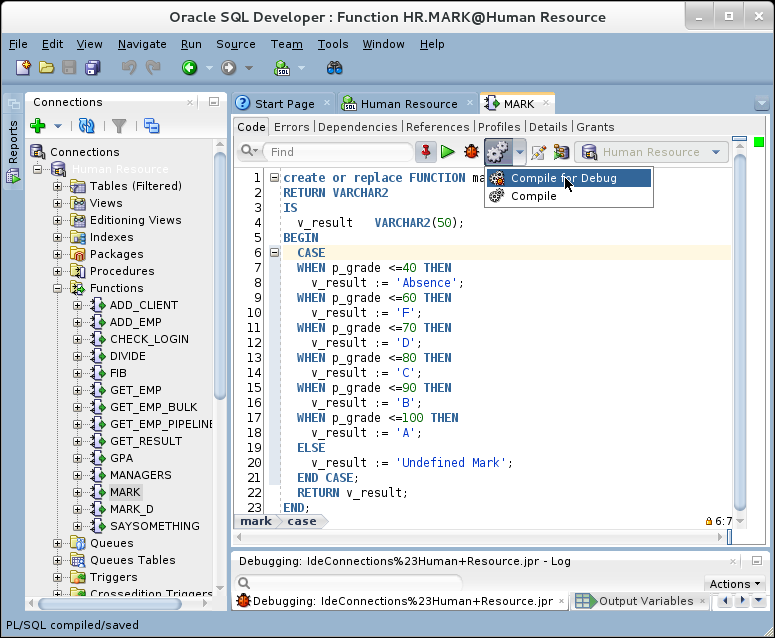
**Step 1:** In Oracle 12c, you have to add entry in ACL to allow SQL Developer using "DBMS\_DEBUG\_JDWP". Execute the following block:

|  |
| --- |
| Command |
| BEGIN |
| **dbms\_network\_acl\_admin.append\_host\_ace** |
| (host=>'127.0.0.1', |
| ace=> sys.xs$ace\_type(privilege\_list=>sys.XS$NAME\_LIST('JDWP') , |
| principal\_name=>'hr', |
| principal\_type=>sys.XS\_ACL.PTYPE\_DB) ); |
| END; |
| / |

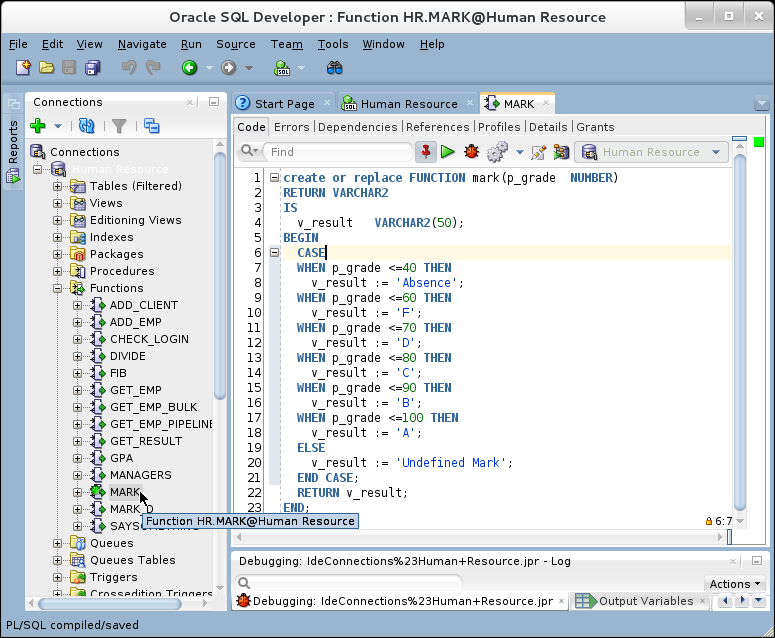
****

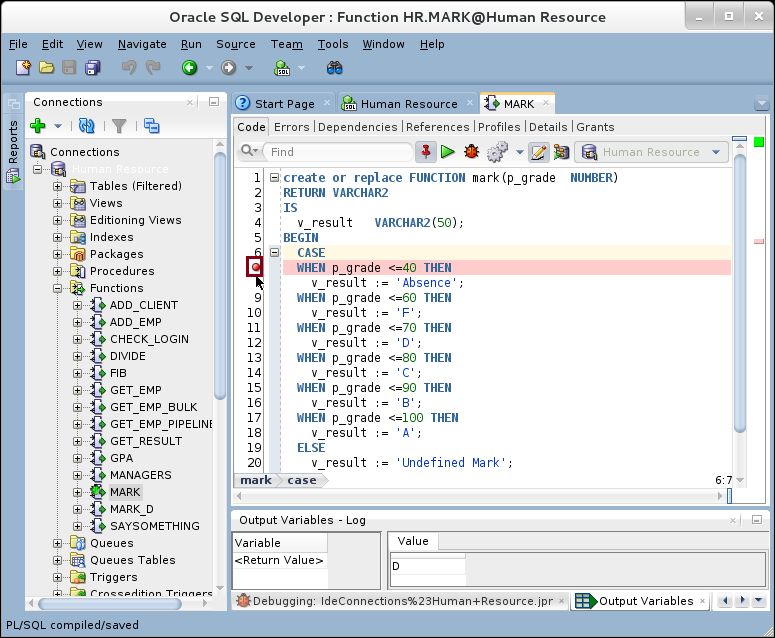
**Step 1:** Return back to SQL Developer. Expand "**Functions**" menu list and click on "**MARK**" function. The source code is automatically shown in the workspace are. By default, SQL Developer opens in read-only mode. Click on "**write mode**" icon. 

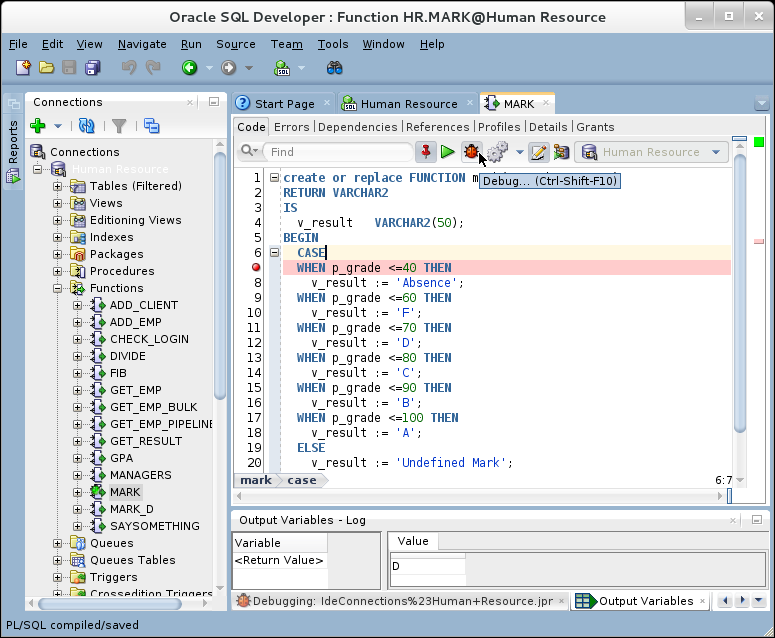
**Step 2:** You can now, add line number to method. Right click on free are beside the method block. Click on "**Toggle Line Numbers**" option from the popup menu. 

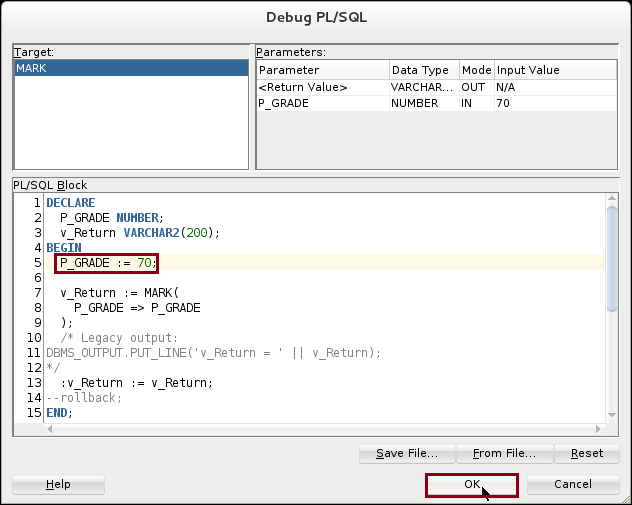
**Step 3:** Before start debugging, you should add debug information to the subprogram. Expand "**Compile**" menu icon and click on "**Compile for Debug**" menu item. 

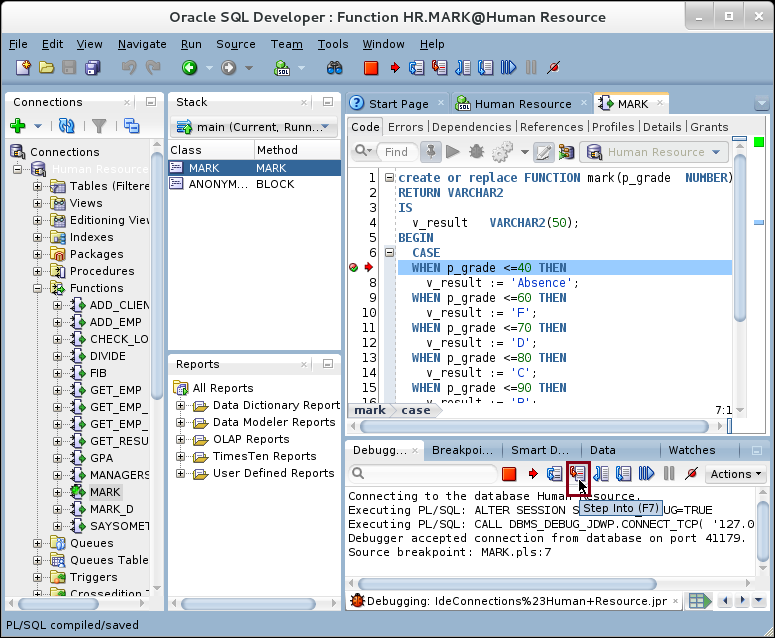
**Step 4:** You should notice the green thing appearing on the method icon; this means the debug information has been added to the method.

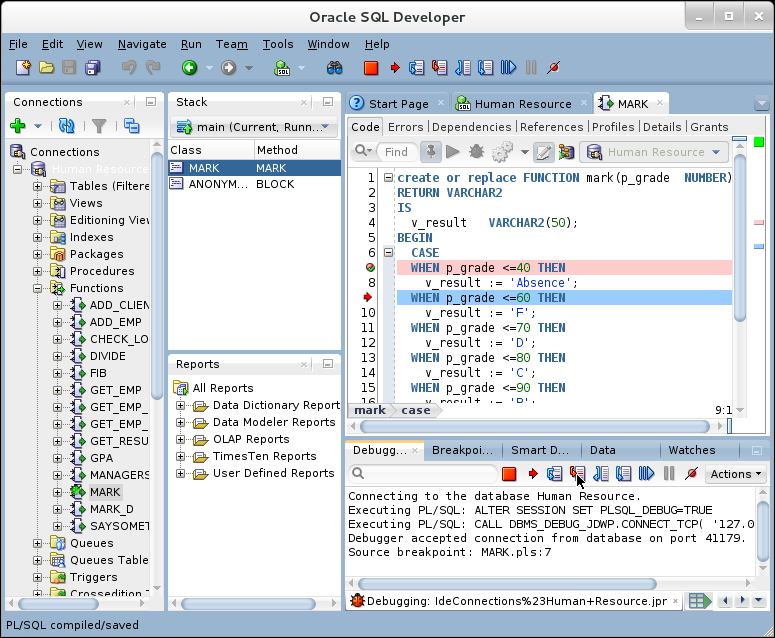


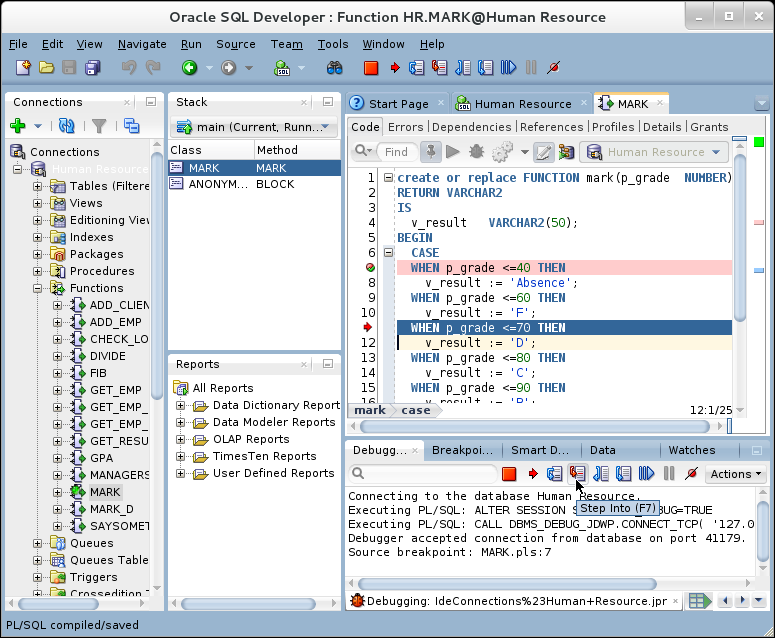
**Step 5:** Click on line "7" to add a debugging point as shown. ****

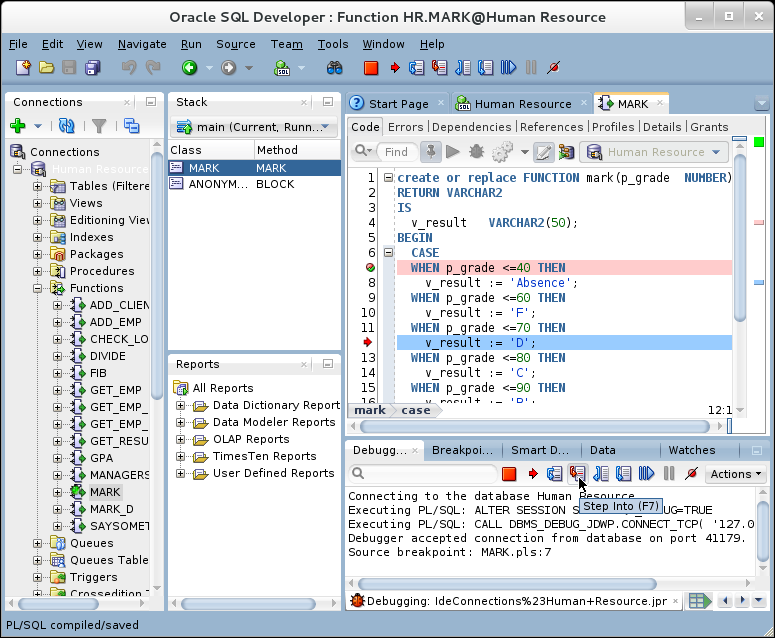
**Step 5:** Click on "**Debug**" icon to start debugging. 

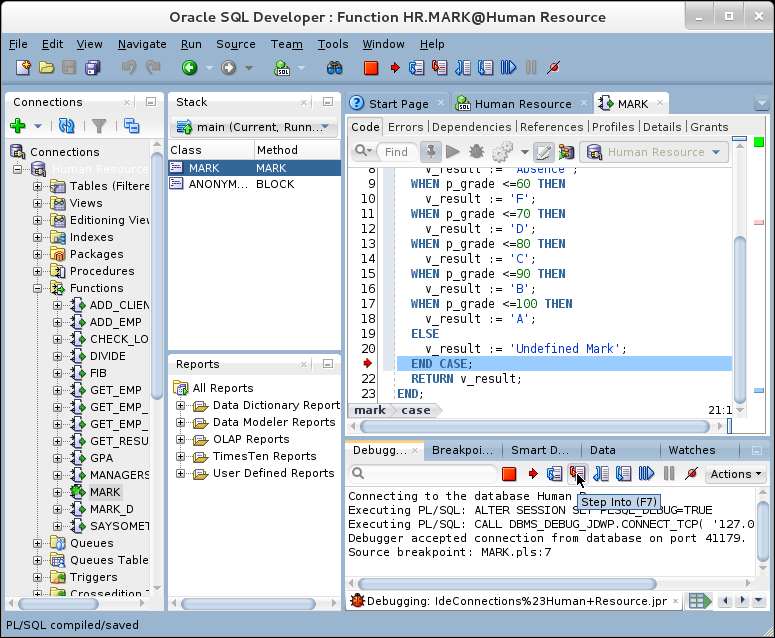
**Step 6:** SQL Developer build a very simple block to call you method, MARK. Edit the inputs of the method as you want, **P\_GRADE=70** for example, and click "**OK**" button. 

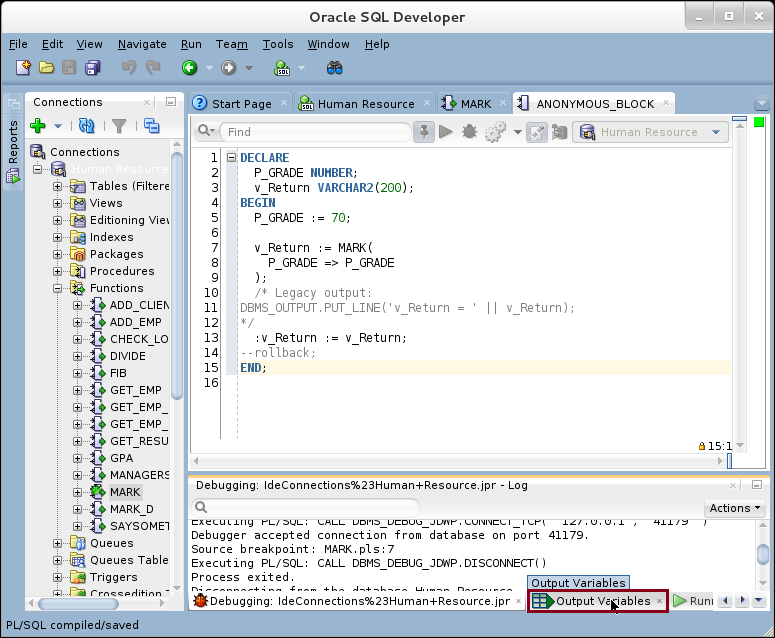
**Step 7:** The debugging stop on the breaking point you added earlier. You can debug navigation buttons to "**Step Over**", "**Step Into**", "**Step Out**", "**Step to End of the Method**" and "**Stop debugging**". Examine each one by yourself; we will use "Step Into" to move in more details through the method. Click on "**Step Into**" icon. 

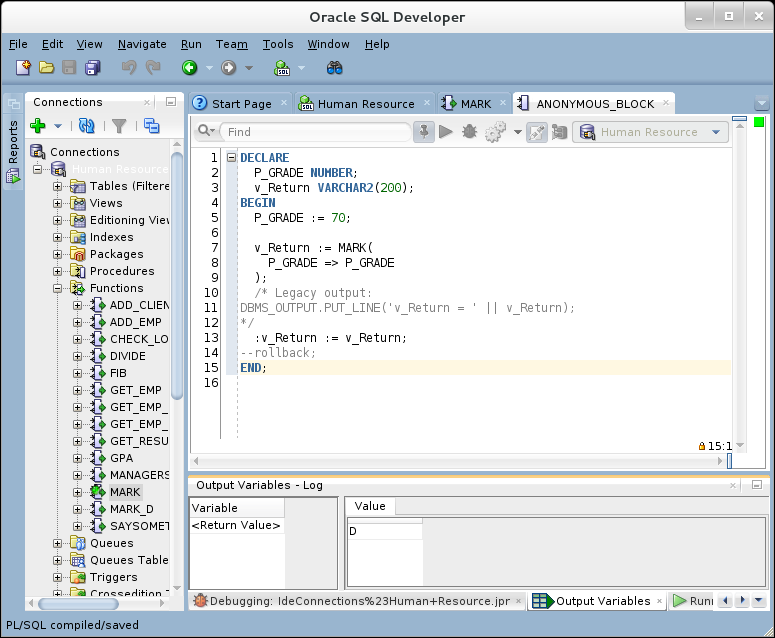
**Step 8:** The debugger move you one-step. Since our value is "**70**", then it is not less than 40. Therefore, it moves to the next condition "**<= 60**". Keep clicking on "**Step Into**" icon. 

**Step 9:** Again, our value is not less than 60, so the execution pointer moves to next condition **"<=70**". Click on "**Step Into**" icon. 

**Step 10:** Our value is less than or equal 70, so the execution pointer enters inside the condition. **v\_result** value is now equal "**D**". Click on "**Step Into**" icon. 

**Step 11:** The execution pointer moves to the end of the **CASE** statement. Keep clicking on "**Step Into**" icon until you exist from the method.

**Step 12:** Finally, you will ends with the test PL/SQL block. To view the output value, click on "**Output Variables**" tab page as shown below. 

**Step 13:** The output value is "**D**". 

# SUMMARY

SQL Developer is a complete PL/SQL development tool. You can easily use SQL Developer to do task which it is very difficult to do with SQL\*Plus. In this chapter, you just have very brief view of SQL Developer debugging feature. Even though, SQL Developer is full of other useful features for both developers and DBAs.

After completing this lab exercise, you should be able to use Oracle SQL Developer in executing SQL and debugging subprograms.

# REFERENCES

* http://www.oracle.com/webfolder/technetwork/tutorials/obe/db/devdays2012/mod2\_sqldev/mod2\_sqldev.html

# INDEX

SQL Developer 2, 3, 5, 10, 13, 14, 15, 18, 25, 41, 42, 48, 57