



Introducing Oracle's SQL Developer

John Jay King

King Training Resources

john@kingtraining.com

Download this paper and code examples from:

<http://www.kingtraining.com>



- Become aware of Oracle's SQL Developer product and its features
- Be able to create, modify, and test both SQL and PL/SQL using SQL Developer
- Use SQL Developer's Object Browser to examine and/or alter existing objects and to create new objects
- Debug stored PL/SQL Procedures and Functions using SQL Developer
- Customize SQL Developer to your personal preferences



- Oracle has released the free SQL Developer tool (formerly known as Project Raptor) and it is super!
- If you have ever used the Free Toad tool or PL/SQL Developer this tool will feel familiar; plus it has more features and supports Oracle 10g too!
- SQL Developer is a Java-based GUI tool (works in Windows, Unix, and Linux)
 - Power of SQL*Plus without a local Oracle client installation
 - Intuitive software; not completely new, based upon JDeveloper
 - Provides features often found only in expensive third-party software including:
 - GUI browsing of database objects, creation, deletion, alteration
 - Debugging complete with breakpoints
 - Query execution
 - Database reporting
 - Multiple connections
 - Tuning features



- SQL Developer is freely downloadable from Oracle using the following URL:
http://www.oracle.com/technology/products/database/sql_developer/index.html
 - Current release as of March 2007 is 1.1.2
 - Release 1.2 scheduled for mid-2007
- Two versions are available:
 - Product with Java 1.5 embedded (about 65mb)
 - Product using installed Java 1.5 (about 39mb)



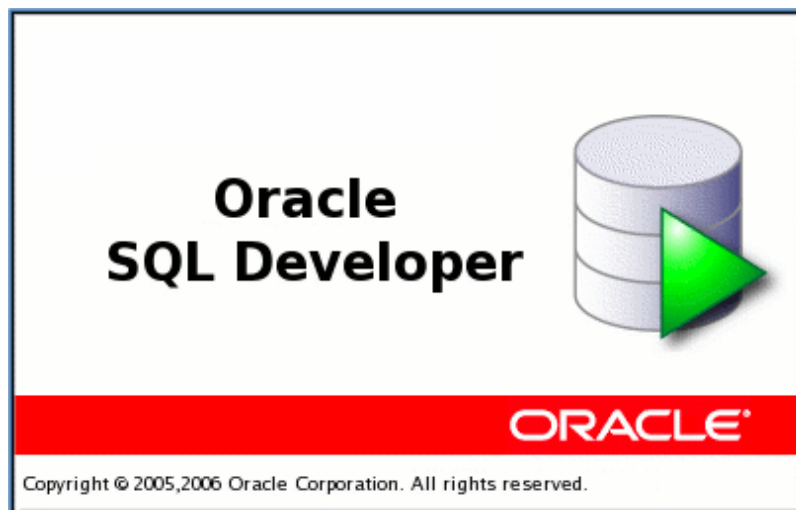
- Taking a page from many open-source products, SQL Developer installs by simply “unzipping” the file downloaded from Oracle
 - No registry entries in Windows
 - No desktop/startbar manipulation in Windows
 - No need for Oracle Client or “tnsnames.ora” file
 - Not an “installed windows program” no need to uninstall or back-out anything, simply delete the install directory
- I find it useful to create a Windows shortcut but it is not necessary



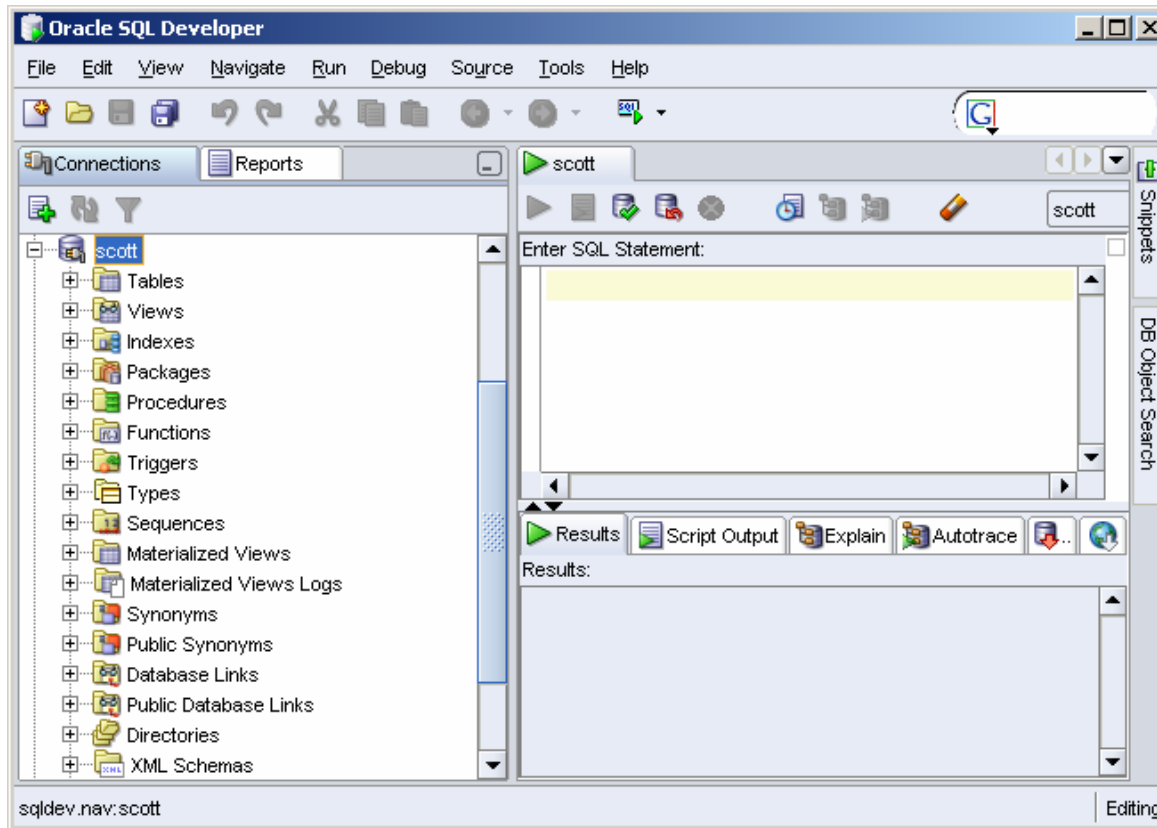
- Major features of SQL Developer include:
 - SQL Worksheet for writing/testing SQL, PL/SQL, and script files
 - GUI object navigator allowing creation, deletion, and alteration of Oracle objects (yours and those you have access to)
 - PL/SQL debugging including breakpoints and manipulation of variables
 - Multiple connections
 - Explain Plan



- Execute the “sqldeveloper.exe” program (found in the unzipped sqldeveloper directory)



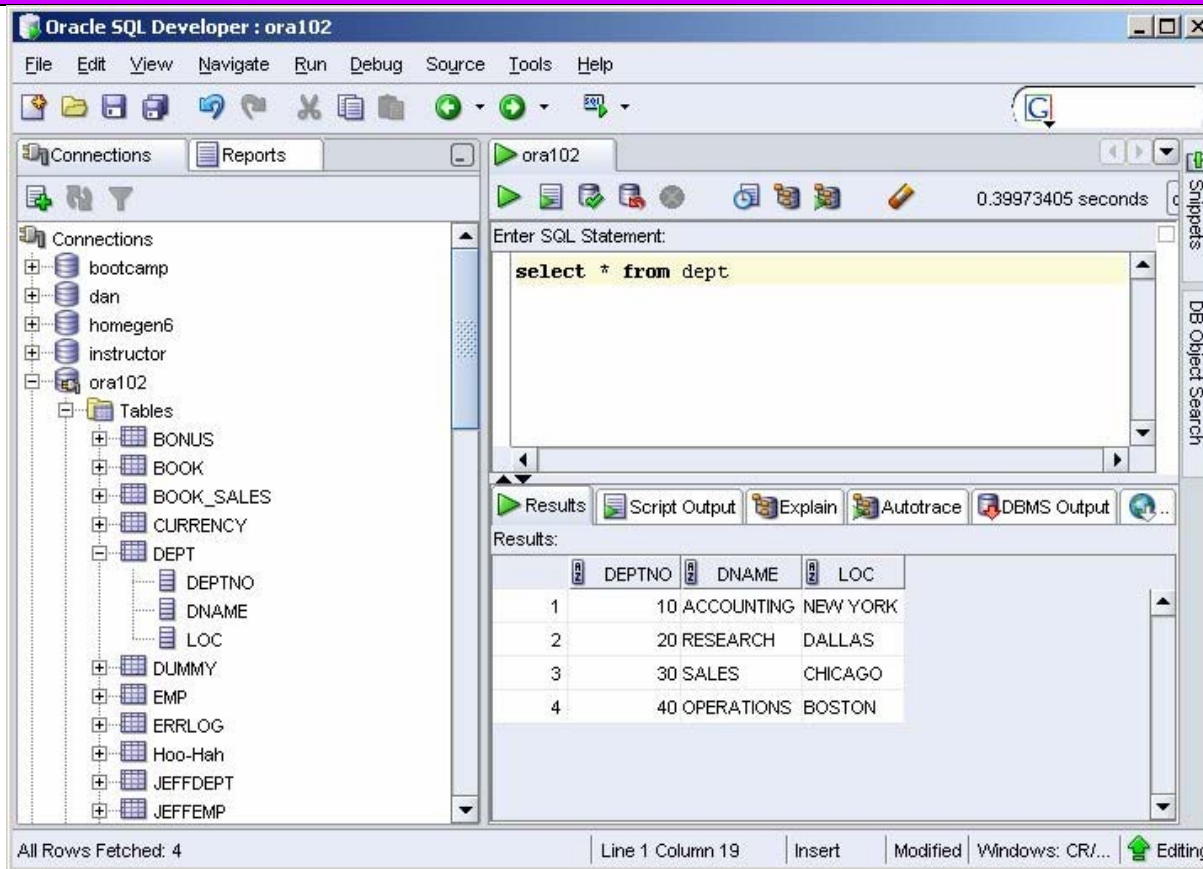
- Requires Java 1.5 (or later) for execution
- Excellent tutorials and help are part of the install and also available on Oracle’s website
- First execution asks if you want to migrate settings from earlier installation
- First execution also asks which file types to associate with SQL Developer



- Upon entry the screen above appears
- Right-click on “Connections” and choose “New Database Connection”



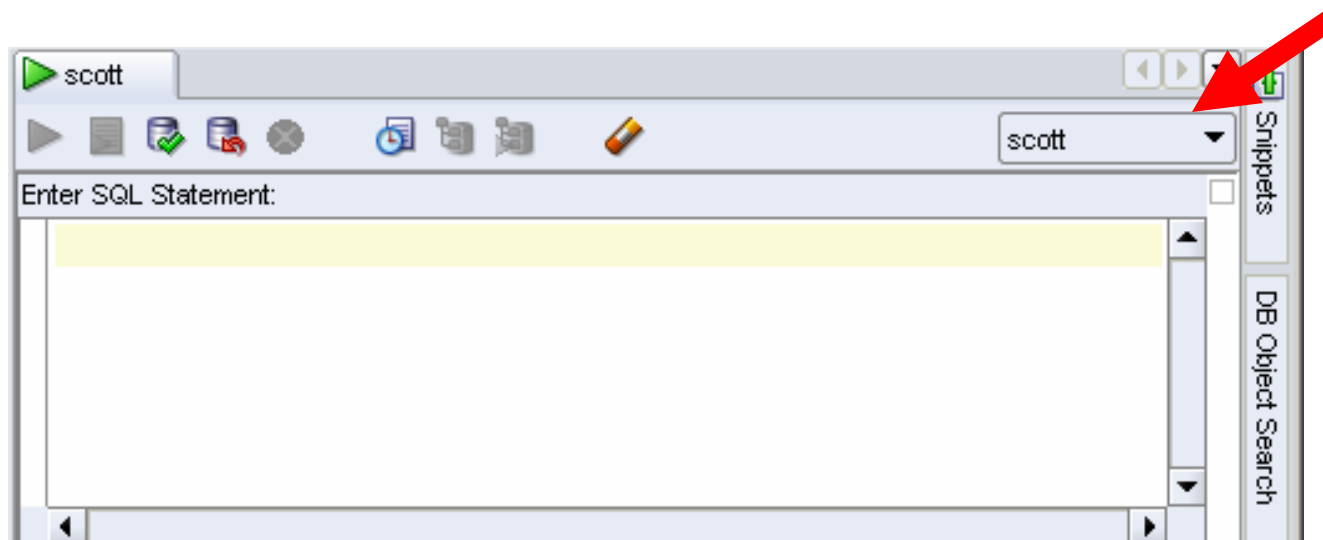
- Enter a connection name (you choose)
- Enter Userid and Password (if shop security permits)
- Enter Hostname, Port, and SID/Service name (get from DBA)
- Choose “Test” to test, then “Connect”



- After logging in to a connection, SQL Developer shows three subpanels:
 - Expandable list of the objects associated with the connected Schema/Userid
 - SQL Worksheet
 - Results area

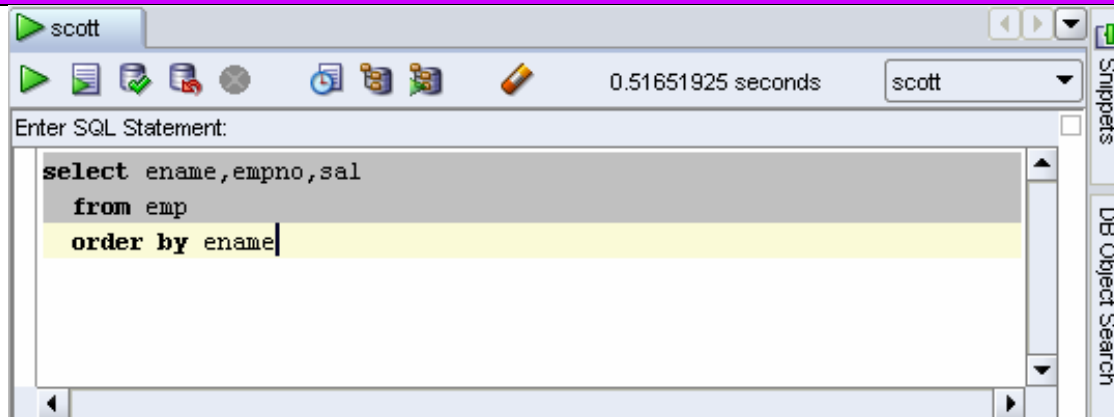


- The SQL Worksheet area has a dialog box where SQL may be keyed, viewed, and modified along with an expandable “snippets” panel providing code fragments
- Note the “pull-down” showing the Connection that will be used to execute any SQL, PL/SQL, or script in the Worksheet














- Enter SQL to test



- SQL Toolbar



-  – Execute Statement (F9)
-  – Run Script (F5)
-  – Commit (F2)
-  – Rollback (F3)
-  – Cancel (Ctrl+Q)
-  – SQL History (F9)
-  – Execute Explain Plan (F6)
-  – Clear (Ctrl+D)

-  – Current Connection pulldown (if no Connection is selected, other icons are disabled/grayed-out)



- If only one statement is in SQL Worksheet, select “Execute Statement” to run it
- If more than one statement is coded, position the cursor in the statement to be executed (or highlight the statement), then select “Execute Statement”

scott

0.51651925 seconds

scott

Enter SQL Statement:

```
select ename, empno, sal
from emp
order by ename
```

Results

	ENAME	EMPNO	SAL
1	ADAMS	7876	1100
2	ALLEN	7499	1600
3	BLAKE	7698	2850
4	CLARK	7782	2450

- To expand the SQL area to a full-screen; double-click the “Connection” tab (double-click again to shrink back)

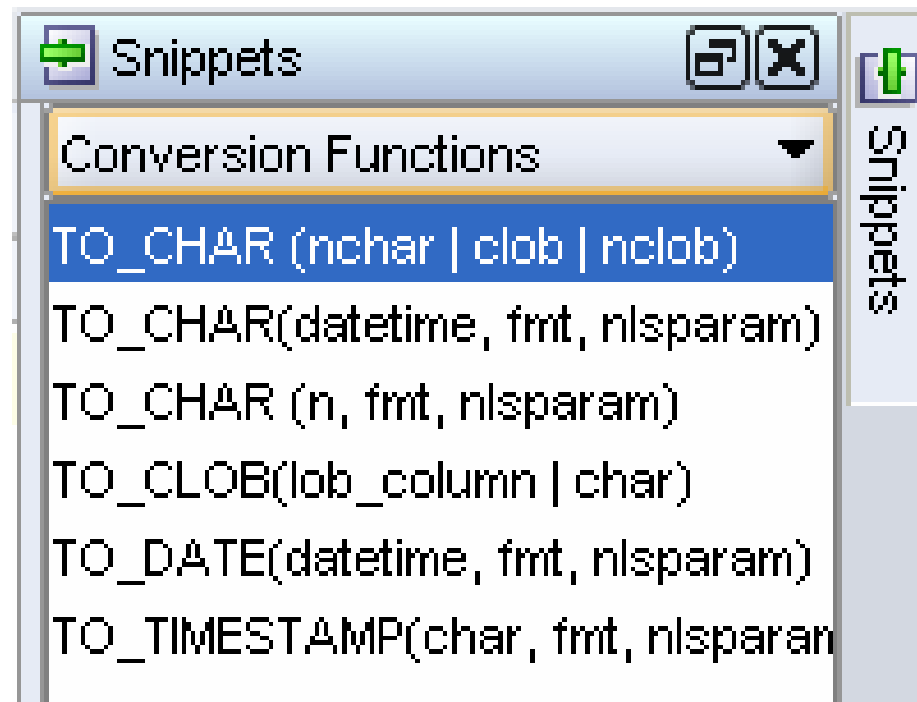


- The “Results” tab shows the results of a single-statement execution

	ENAME	EMPNO	SAL
1	ADAMS	7876	1100
2	ALLEN	7499	1600
3	BLAKE	7698	2850
4	CLARK	7782	2450



- SQL Developer provides many “snippets” of code that may be included via drag & drop or code completion





- If multiple SQL, PL/SQL, or script statements are to be executed together, select “Run Script”
- Results show up in the “Script Output” tab

The screenshot shows the Oracle SQL Developer interface. At the top, there is a toolbar with various icons, a timer showing 0.51792616 seconds, and a dropdown menu set to 'scott'. Below the toolbar is a text area labeled 'Enter SQL Statement:'. The text area contains two SQL queries. The first query is highlighted in yellow and reads: `select ename,empno,sal
from emp
order by ename;`. The second query is `select deptno,dname
from dept`. The interface also includes a scroll bar on the right side of the text area.



- When a script is executed, the results show up in the “Script Output” tab

The screenshot shows a database query results window with the following elements:

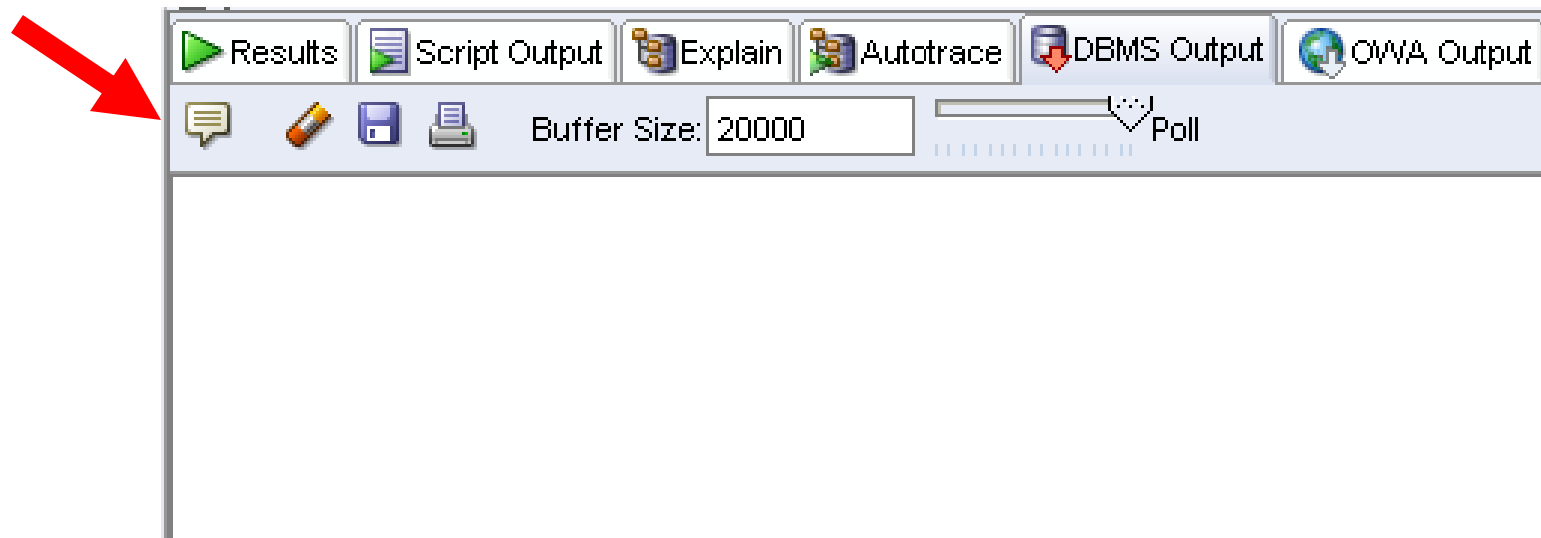
- Navigation tabs: Results, Script Output (selected), Explain, Autotrace, DBMS Output, OWA Output.
- Toolbar: Edit, Save, Print icons.
- Table data:

Test5555	5555	1234.56
WARD	7521	1250
- Text: 16 rows selected
- Header section:

DEPTNO	DNAME
10	ACCOUNTING



- If DBMS_OUTPUT.PUT_LINE was used in executed SQL (or a script) it will show up in the DBMS_OUTPUT tab
- DBMS_OUTPUT must be enabled prior to running your test (you may wish to increase the buffer size too)





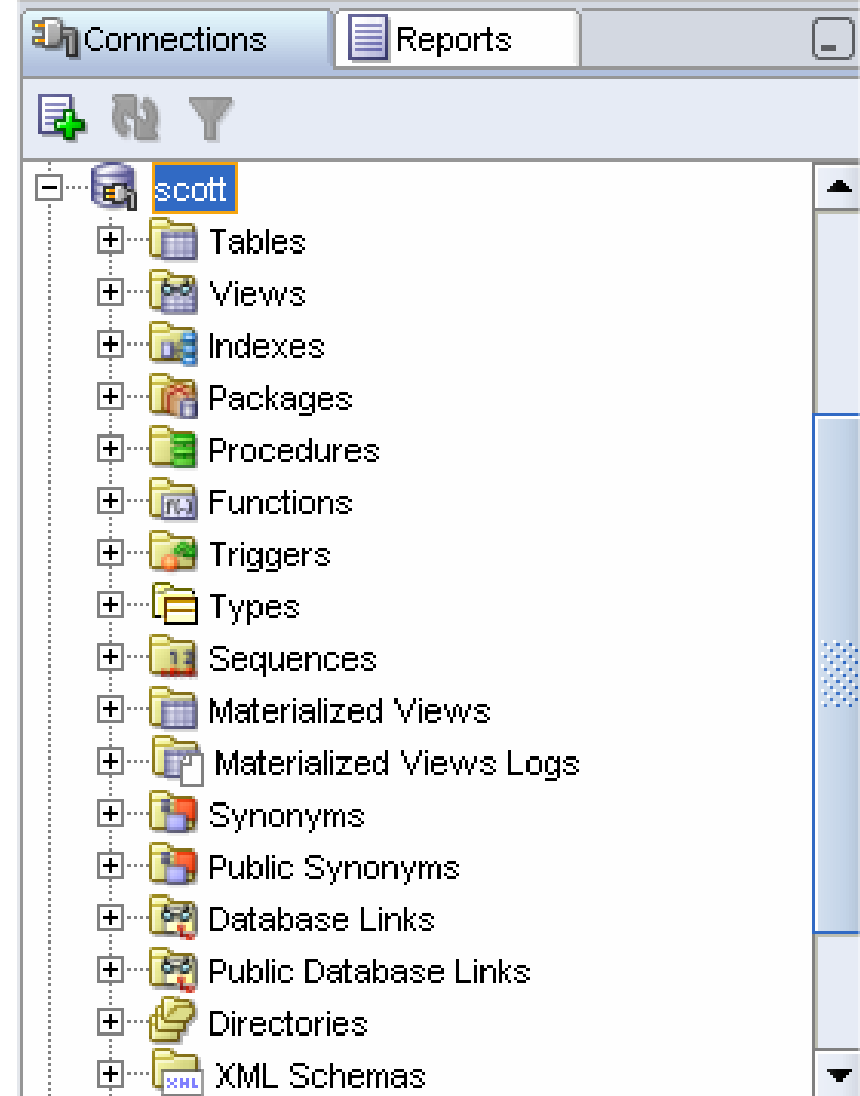


- OWA output will show up in the OWA OUTPUT tab
- OWA OUTPUT must be enabled prior to running your test





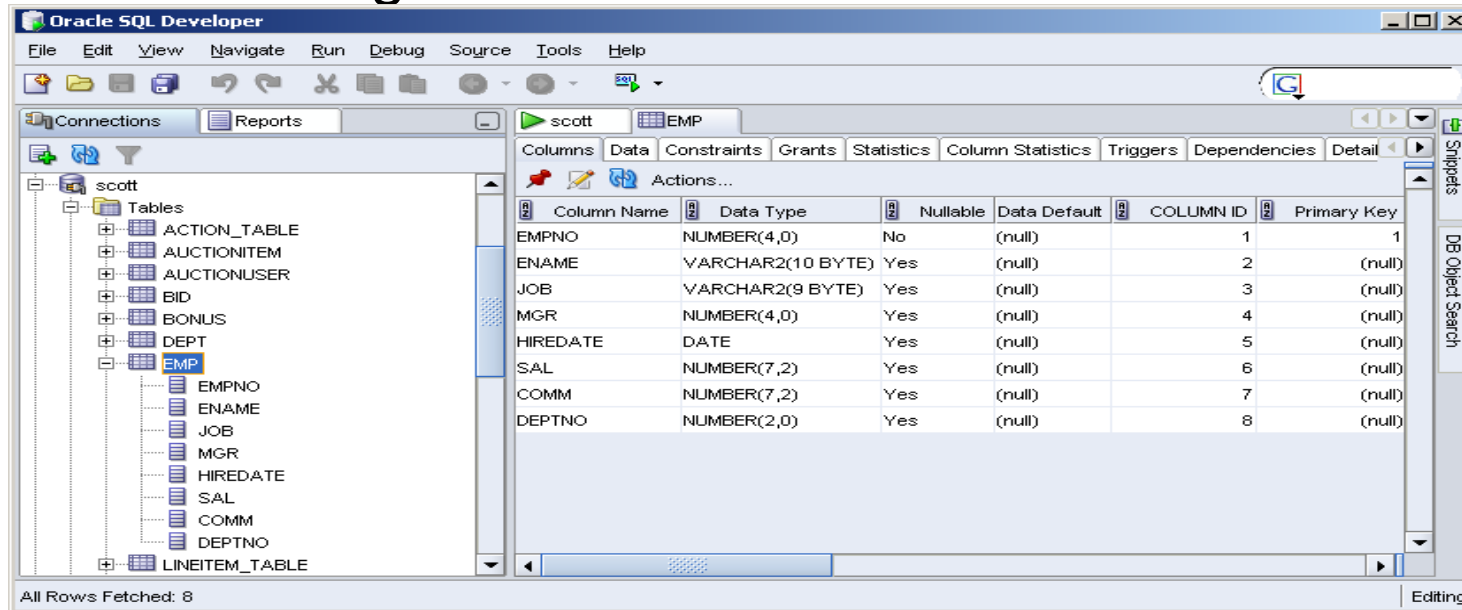
- The objects accessible to the Userid/Schema used to make a connection are available for browsing via the object navigator
- This navigator follows the usual pattern
 - Expand 
 - Contract 
- Objects available are those belonging to the connection Userid/Schema (user_) and objects available in some way to the Userid/Schema (all_)



“Browsing” a Table



- Selecting an object (EMP table below) displays a multi-tab panel containing various bits of information about the table



- Columns
- Data
- Indexes
- Constraints
- Grants
- Statistics
- Column Statistics
- Triggers
- Dependencies
- Details
- SQL

(double-click Table Name (EMP) tab to expand/contract display)



- Right-click on an object to display its “Context Menu” (table menu shown)
 - The Edit option opens a dialog to view/modify the table definition
 - Other options open additional menus allowing the user to issue commands graphically rather than from the command line





Edit Table

Schema: SCOTT

Name: EMP

Table Type: Normal External Index Organized Temporary (Transaction) Temporary (Session)

Columns

- Primary Key
- Unique Constraints
- Foreign Keys
- Check Constraints
- Indexes
- Storage Options
- Lob Parameters
- Comment
- DDL

Columns:

- EMPNO
- ENAME
- JOB
- MGR
- HIREDATE
- SAL
- COMM
- DEPTNO

Column Properties

Name: EMPNO

Datatype: Single Complex

Type: NUMBER

Precision: 4

Scale: 0

Default:

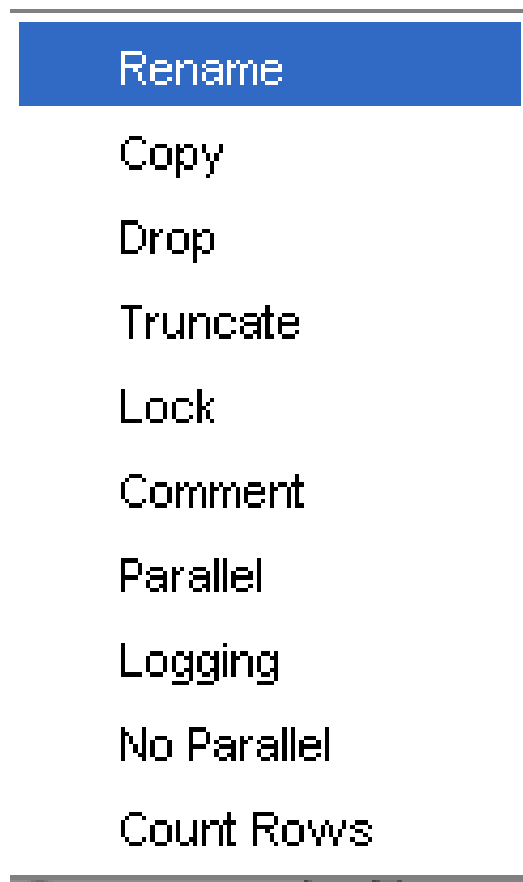
Cannot be NULL

Comment:

Help OK Cancel

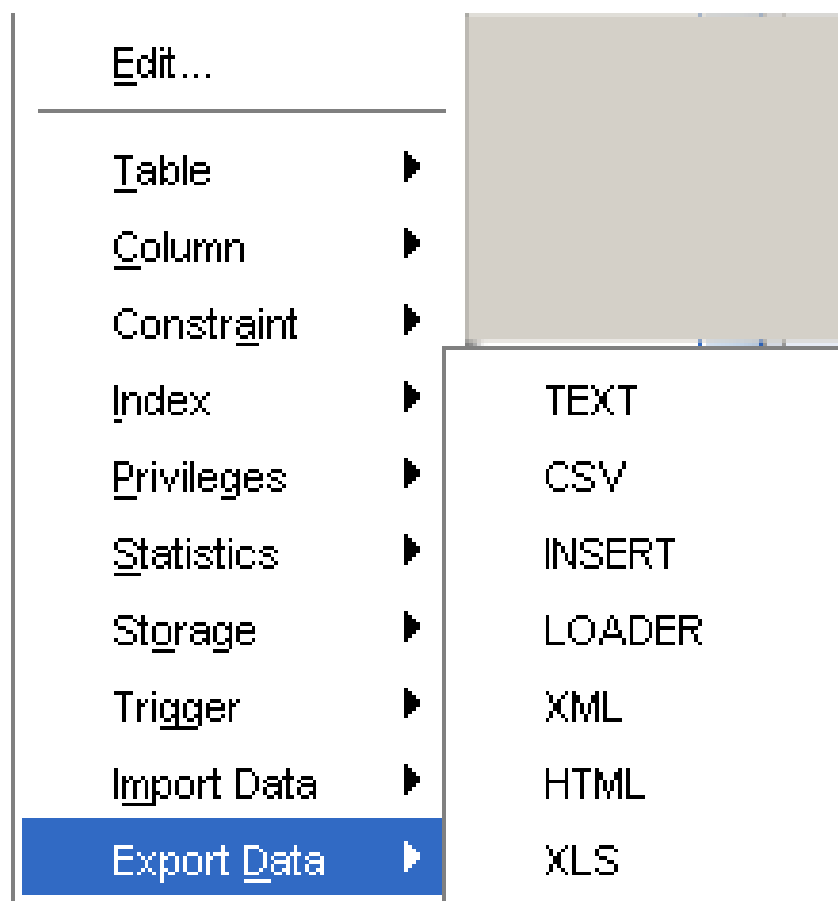


- Choose “Table” to see the list of table manipulation options



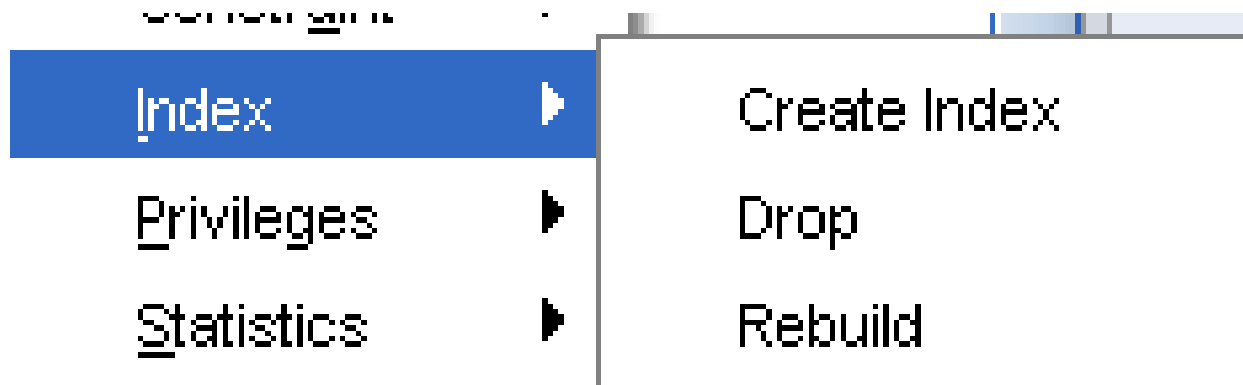


- Choose “Export Data” to list table export options





- Choose “Index” to create or modify indexes

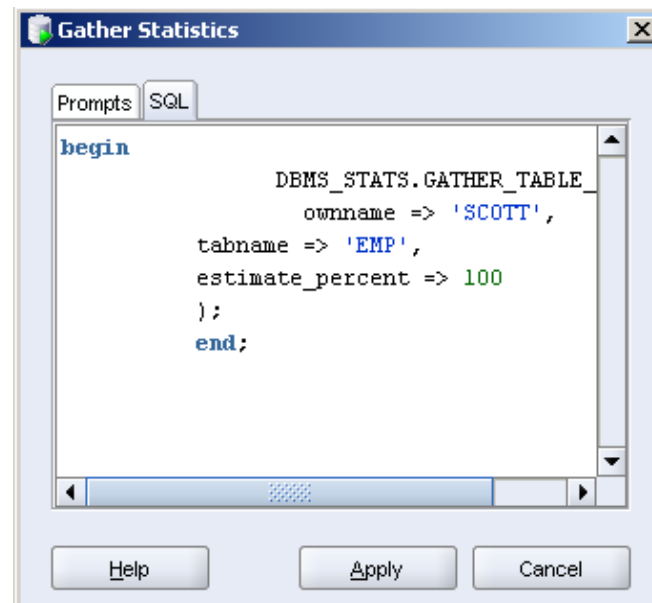
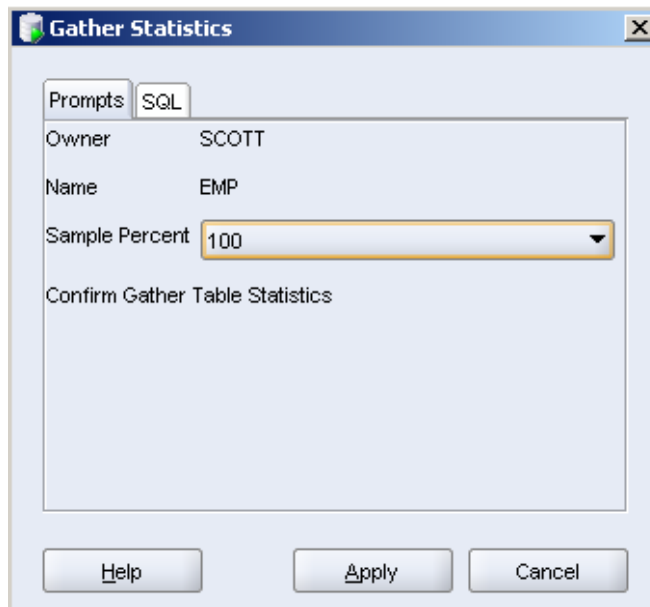




- Choose “Statistics” to use DBMS_STATS to gather statistics or validate structures

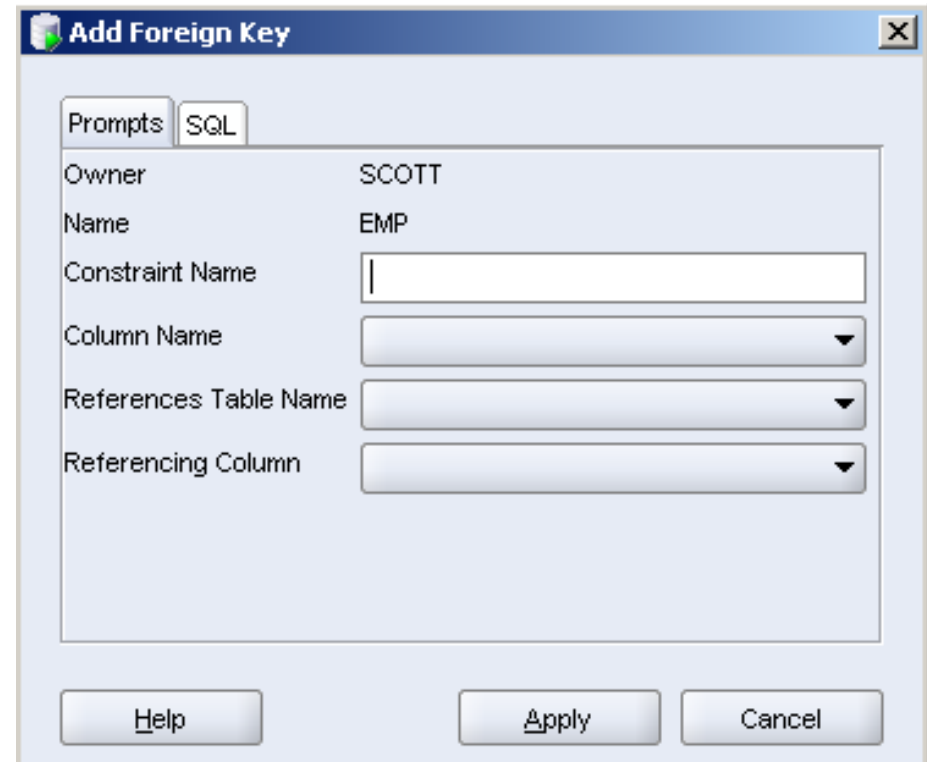
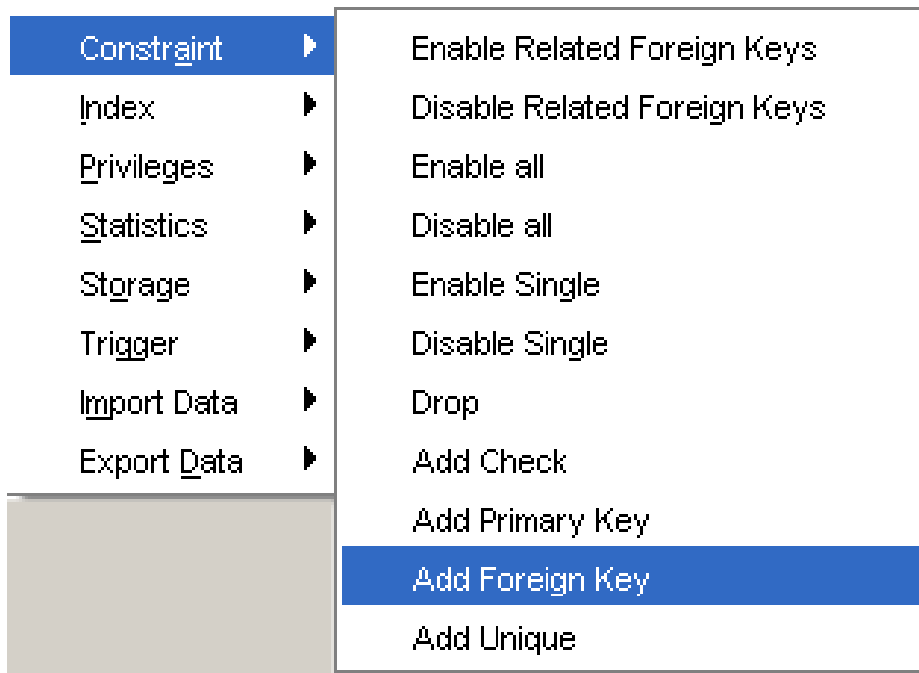


- After choosing Gather Statistics a dialog appears allowing you to change the sample size and view/copy the PL/SQL code



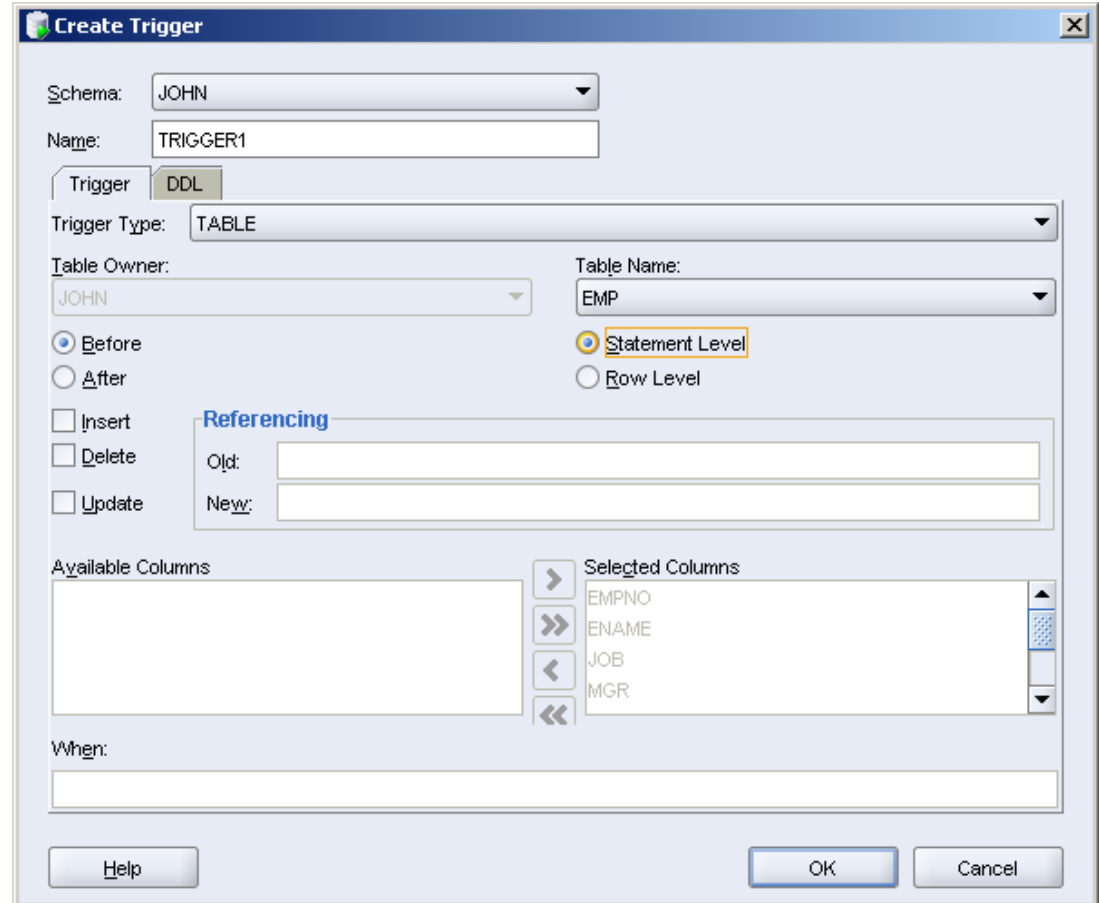
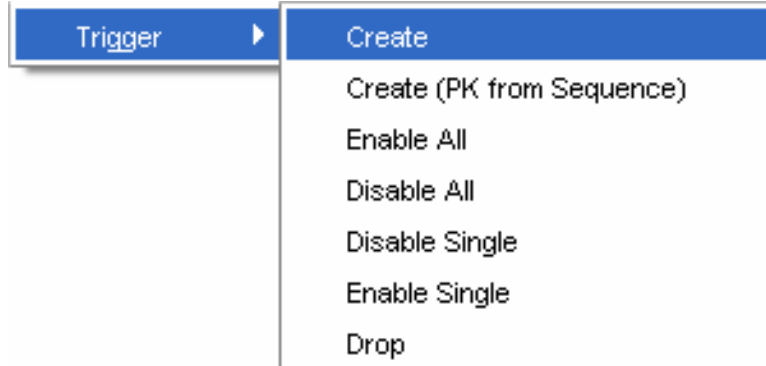


- Choose “Constraint” to view or alter table constraints





- Choose “Triggers” to build or alter table triggers
(builds skeleton DDL that may be modified in SQL Workspace)





- Right-click on an object type to display its “Context Menu”
- Here is the “Context Menu” for Tables





- Choose “Create Table” to display a panel for creating a standard table
- Choose “Add Column” or “Remove Column” as desired

Schema: SCOTT Advanced

Name: TABLE1

Table DDL

Column Name	Type	Size	Not Null	Primary Key
COLUMN1	VARCHAR2	4000	<input type="checkbox"/>	<input type="checkbox"/>

Add Column Remove Column

Help OK Cancel



- Choose “Create Table” to display a panel for creating a standard table, then mark the “Advanced” box to display other options including:
 - External tables
 - IOT
 - Global Temporary Table(panel changes for each table type)

The screenshot shows the 'Create Table' dialog box with the following details:

- Schema:** SCOTT
- Name:** TABLE1
- Table Type:** Normal External Index Organized Temporary (Transaction) Temporary (Session)
- Column Properties:**
 - Name:** COLUMN1
 - Datatype:** Simple Complex
 - Type:** VARCHAR2
 - Size:** 4000
 - Units:** (empty)
 - Default:** (empty)
 - Cannot be NULL
 - Comment:** (empty)



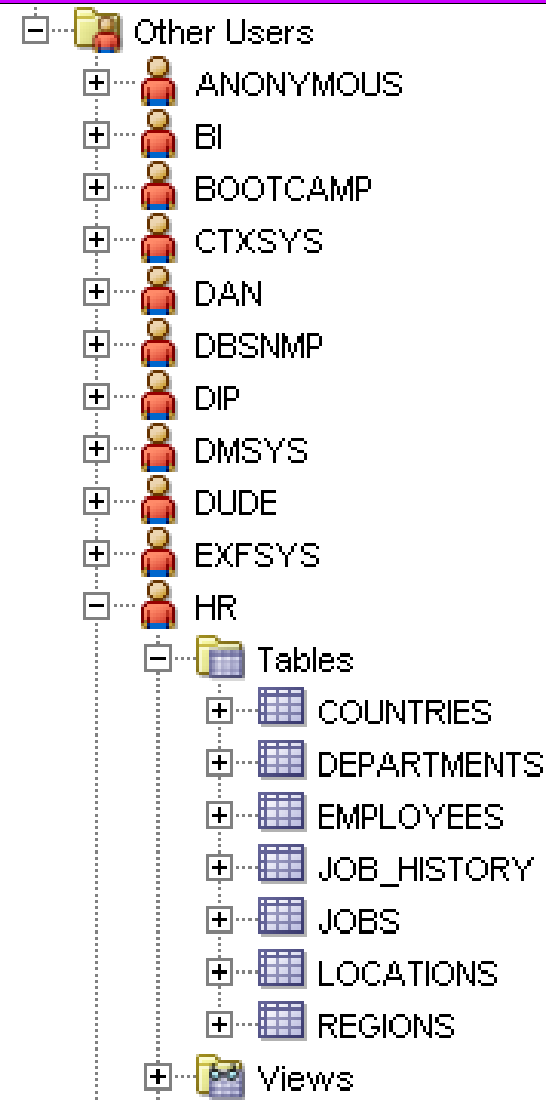
- From the Materialized View context menu choose “Create Materialized View” to build an MVIEW by filling in the blanks!

The screenshot shows the 'Create Materialized View' dialog box with the following settings:

- Schema: SCOTT
- Name: MVIEW1
- Advanced:
- SQL Query: Properties: DDL
- Refresh Options:
 - When: DEMAND
 - Method: COMPLETE
 - Type: (blank)
 - Start on: 17-Mar-2007 15:08:36
 - Next: 17-Mar-2007 15:08:36
 - Use Rollback Segment
 - Master Local
 - Default Use (blank)
 - Constraints
 - Enforced Trusted
- Materialized View Options:
 - Parallel 0
 - Enable Cache
 - Enable Query Rewrite
 - Prebuilt Option
 - Reduced Precision
 - No Reduced Precision
 - Build Type: Immediate
- Using Index Clause:
 - Add USING INDEX Clause
 - Index Tablespace: (blank)



- At the bottom of the object navigator, the “Other Users” option lists the Userid/Schema names that have one or more objects available to you
- Expand a Userid/Schema to see the objects exposed to you





- As you might imagine, selecting Packages, Procedures, Functions, or Triggers allows you to display and/or alter existing stored PL/SQL
- Context menus for each type of stored PL/SQL allow you to create new objects using the same “paint by the numbers” approach used for Tables and Materialized Views



The screenshot shows the Oracle SQL Developer interface. On the left, the Object Browser displays a tree view of database objects, including a procedure named TESTIT with three parameters: INVAR1, INVAR2, and OUTVAR. The main editor window shows the source code for the TESTIT procedure. The code is as follows:

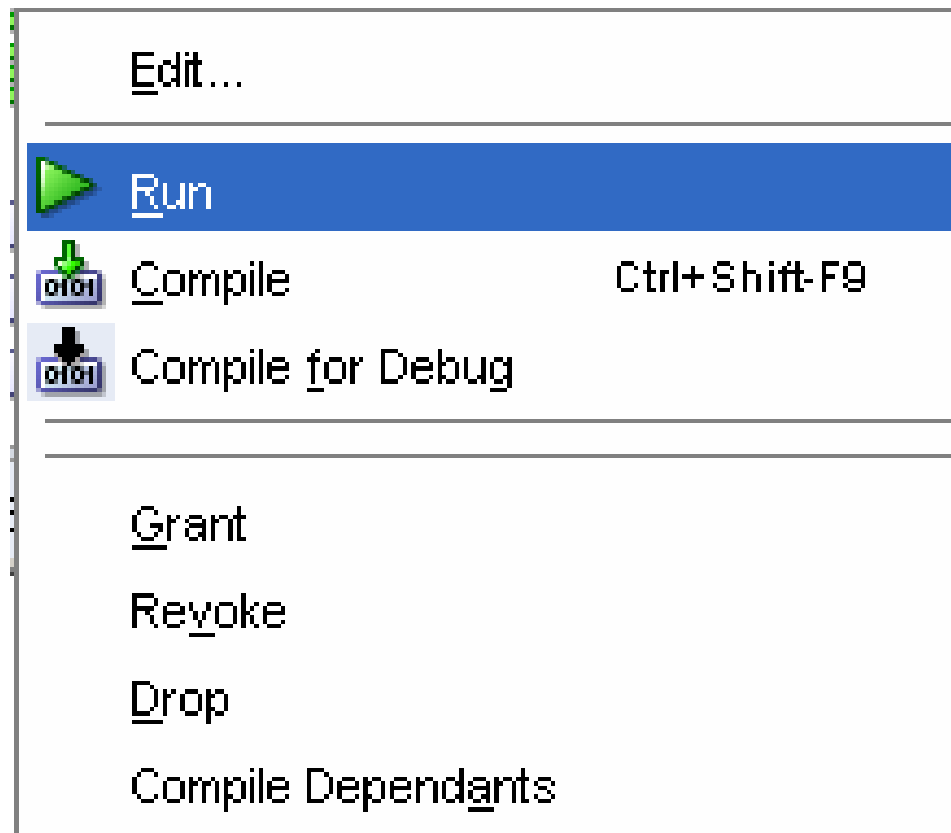
```
create or replace PROCEDURE testit(invar1 IN NUMBER, invar2 IN NUMBER, outvar OUT NUMBER)
mycounter NUMBER := 0;
delimit_value NUMBER := 10;
BEGIN
  dbms_output.put_line('Hip-hip-hooray! Values: invar1=' || invar1
    || ' invar2=' || invar2);
  FOR loopctr IN invar1 .. invar2
  LOOP
    mycounter := mycounter + loopctr;
    if mycounter > delimit_value then
      mycounter := invar1;
    end if;
    dbms_output.put_line('Value is ' || mycounter);
  END LOOP;

  outvar := mycounter;
END;
```

The status bar at the bottom of the window indicates the current context: PROCEDURE JOHN.TESTIT@ora102 and the mode is Editing.



- Right-Click on a PL/SQL object to display the following context menu





- SQL Developer allows:
 - Setting and/or removal of breakpoints
 - Monitoring and manipulation of variables
- “Compile for Debug” must be executed to make an object available for debugging
- Once compiled for Debugging, whenever executed in “Debug” mode the code will stop where directed



- When editing PL/SQL code the icons shown here appear



Run



Debug



Compile



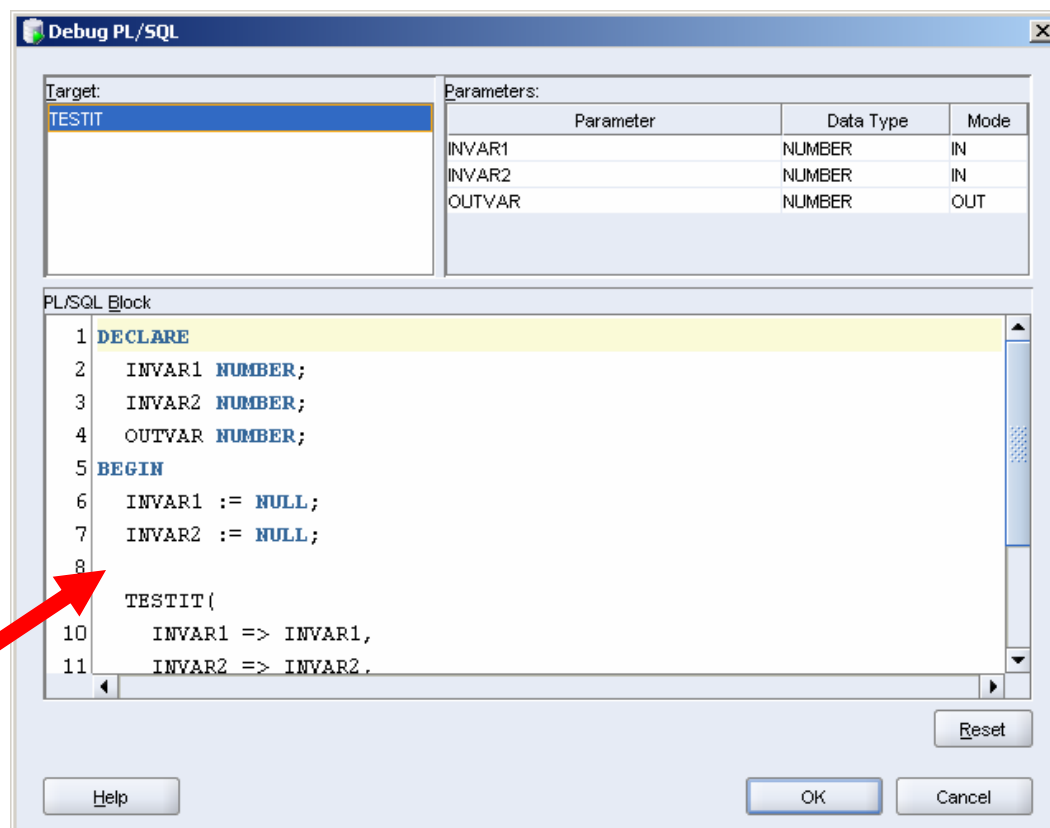
Compile for Debug

(PL/SQL name in navigator is italicized after Compile for Debug)





- The debugging “Run” option creates an anonymous PL/SQL to test the selected PL/SQL
- Parameter values are initially set to “null” so you must provide the values you wish to test





Oracle SQL Developer - DefaultWorkspace.jws

File Edit View Navigate Run Debug Source Tools Help

Connections Reports

Ora101 TESTIT TESTIT

Procedures

- ADD2
- ADDEM
- ADDEMUP
- ADDEMUP2
- BUILD_IDX
- BUILD_SEQ

Stack

Subroutine

- JOHN.TESTIT
- JOHN.ANONYMOUS_BLOCK

Run Manager

Processes

```

7          || ' invar2=' || invar2);
8  FOR loopctr IN invar1 .. invar2
9  LOOP
10         mycounter := mycounter + loopctr;
11         if mycounter > delimit_value then
12             mycounter := invar1;
13         end if;

```

Debugging - ... Breakpoints

Description	Type
Oracle exception, Exception Breakpoint	
\$Oracle.ProcedureSource Breakpoint	
\$Oracle.ProcedureSource Breakpoint	
\$Oracle.Trigger.JCSource Breakpoint	
\$Oracle.Trigger.JCSource Breakpoint	

Smart Data

Name	Value	Type
loopctr	10	PLS_INT...
mycounter	0	NUMBER

PROCEDURE JOHN.TESTIT@Ora101

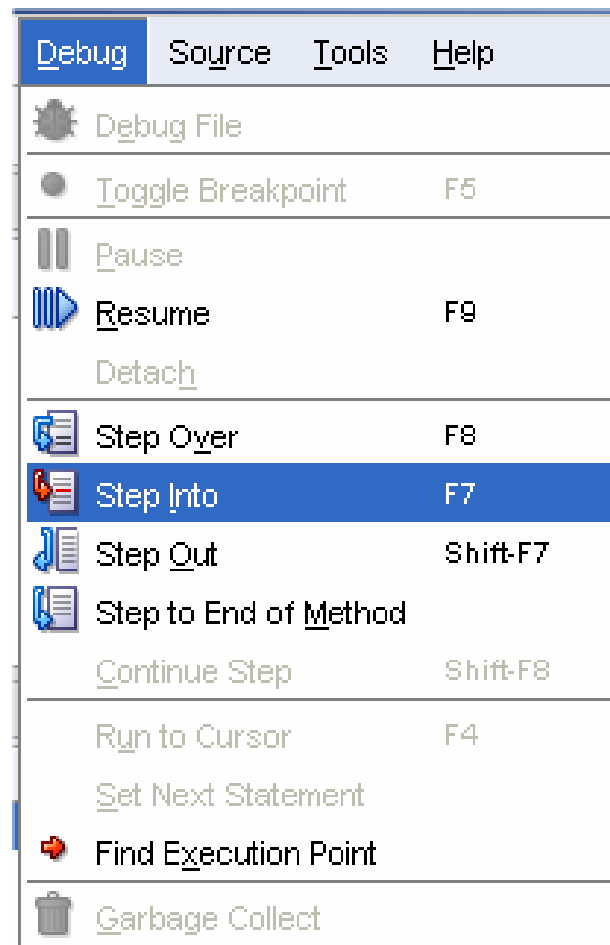
Debugging











- The debugging pane shows four important subpanels
 - Breakpoints Displays breakpoints, context menu allows new breakpoints
 - Smart Data Displays breakpoint variables (double-click value to alter)
 - Data Displays all variables (double-click value to alter)
 - Watches Displays or add "watches"



- When editing or testing PL/SQL compiled for debugging, the “Debug” menu option appears

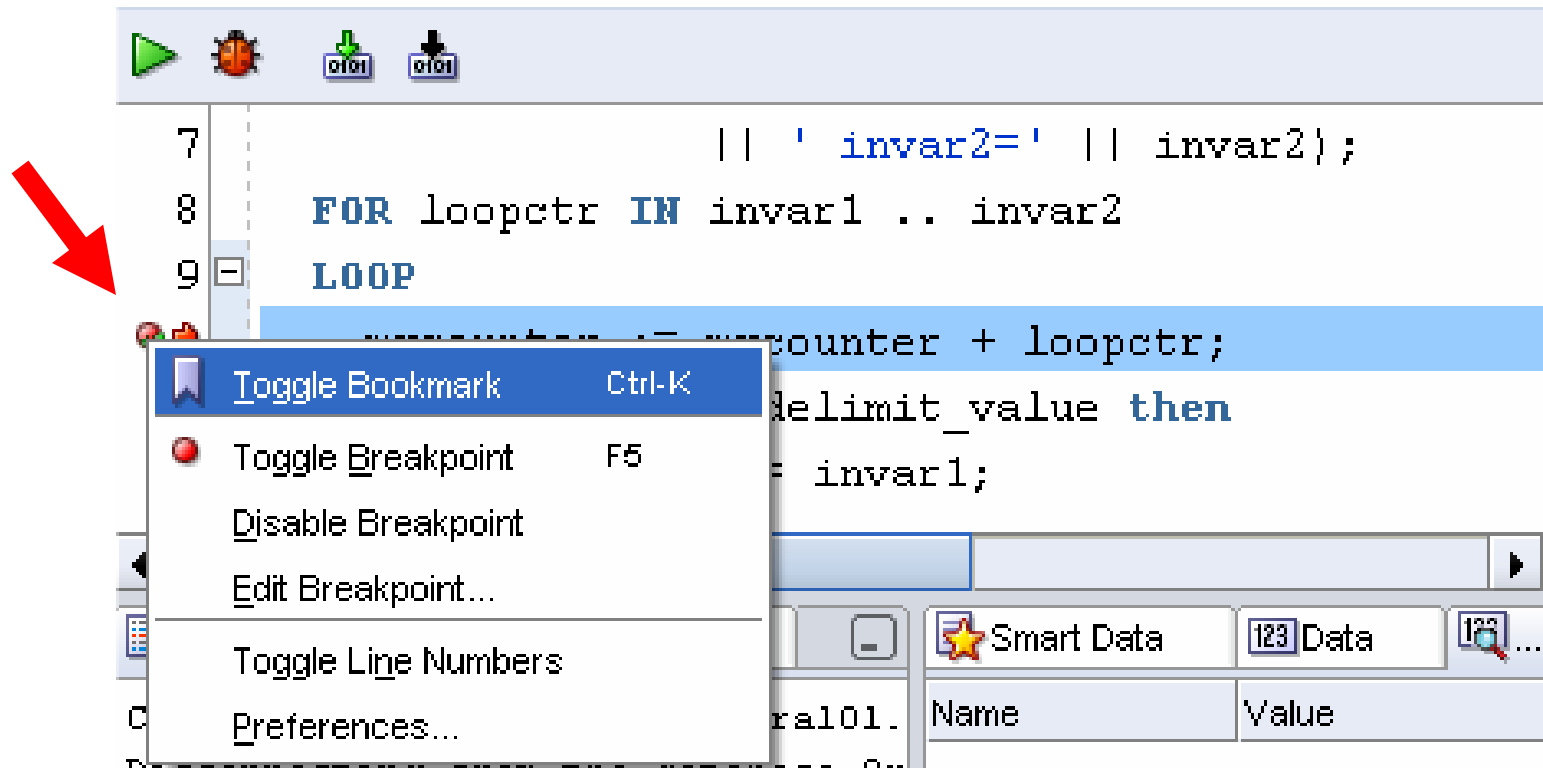




- The “Debugging” tab provides a set of icons for controlling the debug session
 -  – Find Execution Point
 -  – Resume
 -  – Step Over
 -  – Step Into
 -  – Step to End of Method
 -  – Pause
 -  – Terminate
 -  – Garbage Collection



- While debugging, breakpoints may be set/unset by clicking in the margin to the left of the PL/SQL text or by using the context menu





- Smart Data and Data panels allow viewing and/or alteration of values
- Double-click on a variable's value to display the alteration dialog

Name	Value	Type
INVAR1	10	NUMBER
INVAR2	200	NUMBER
OUTVAR	NULL	NUMBER
MYCOUNT0		NUMBER
DELIMIT_V#10		NUMBER
LOOPCTR	10	PLS_INT...

Modify Value [X]

NUMBER INVAR2

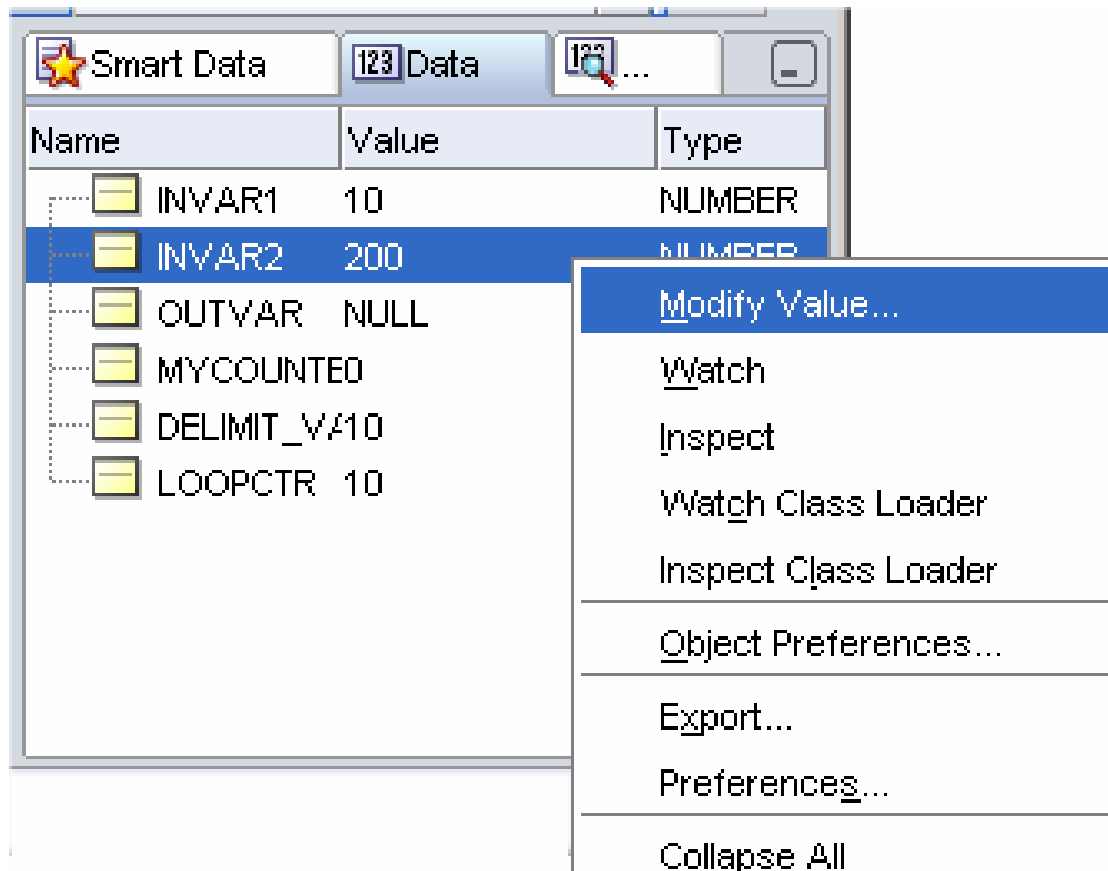
Current Value:

New Value:

Interpret New Value as Object Address



- Instead of double-clicking on a data item, right-click to display the context menu





- Use the “Explain Plan” icon or menu option to show the proposed plan for a statement

Enter SQL Statement:

```

1 select ename,dname,job
2   from many_emps emp join many_depts dept
3     on emp.deptno = dept.deptno
4  where dept.deptno in (20,40)
5     and job in ('MANAGER','OPERATOR')
6     and extract(year from hiredate) = 1981

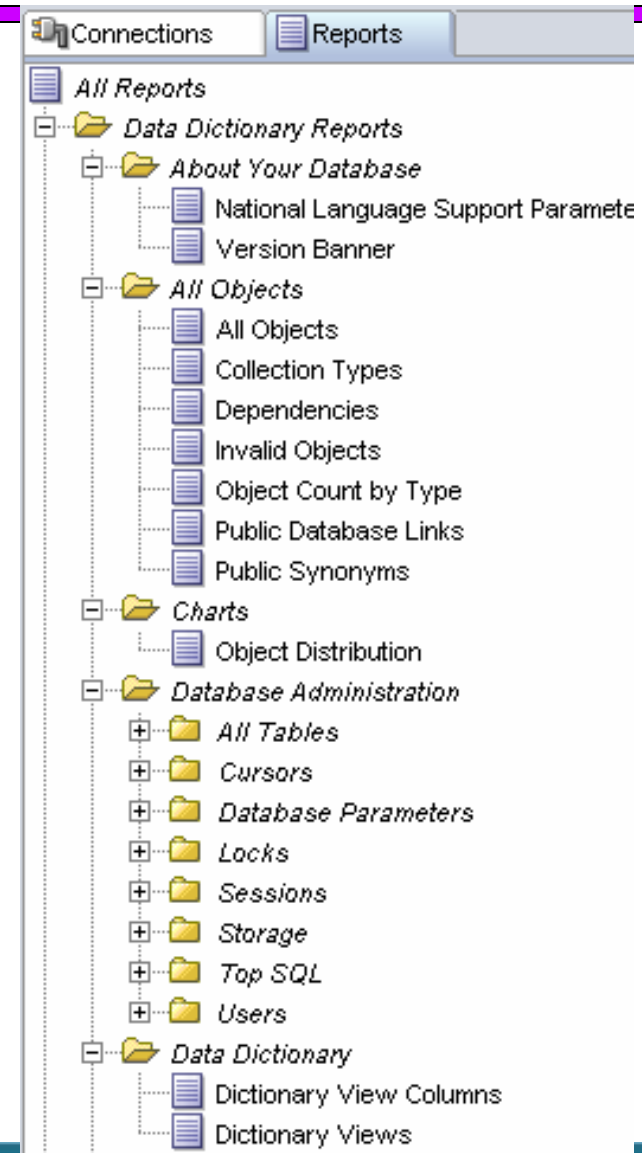
```

Results | Script Output | **Explain** | DBMS Output | OWA Output

Operation	Optimizer	Cost	Cardinality	Bytes	Partition Start	Partition Stop
SELECT STATEMENT	ALL_ROWS	91	16	672		
SORT(OBJECT BY)		91	16	672		
NESTED LOOPS		90	16	672		
TABLE ACCESS(FULL) JOHN.MANY_EMPS	ANALYZED	87	58	1566		
TABLE ACCESS(BY INDEX ROWID) JOHN.MANY_DEPTS	ANALYZED	1	1	15		
INDEX(UNIQUE SCAN) JOHN.MANYDEPT_DEPTNO	ANALYZED	0	1			



- Oracle has built many of the types of reports routinely created from the catalog and V\$ views into SQL Developer
 - Some DBA-oriented
 - Some Developer-oriented
- Before you create your own reports, look to see what's available here first!

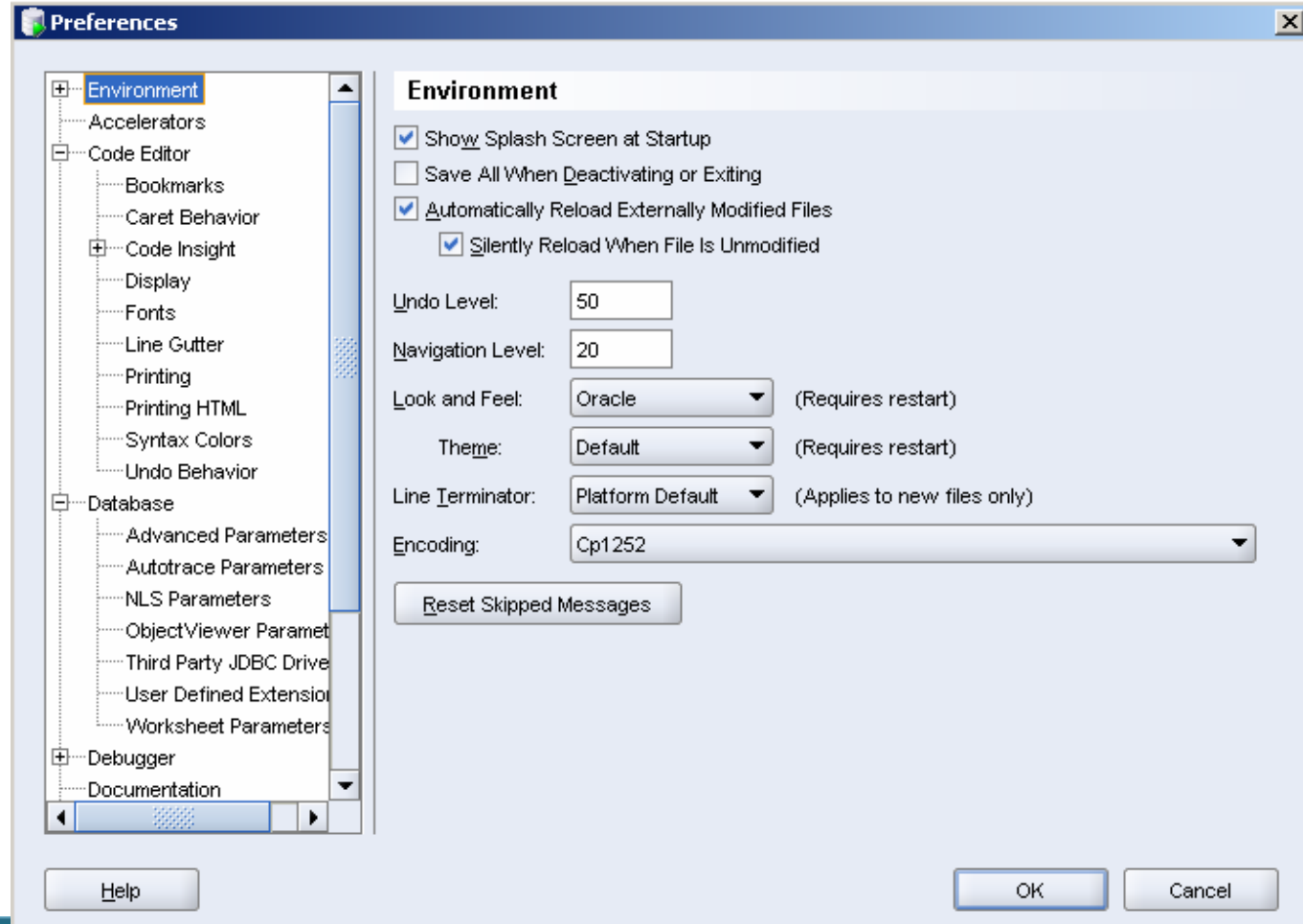




- SQL Developer provides:
 - Great context-oriented help (F1 key)
 - Via the Help menu a comprehensive discussion of SQL Developer
 - Via the Help menu excellent references for SQL and PL/SQL
 - Online documentation and “support” via the Oracle website and many blogs



- The “Tools->Preferences” menu option provides a fairly comprehensive set of tools to fine-tune SQL Developer to your needs





- Free products should worry lots, Oracle's SQL Developer:
 - Supports current SQL & PL/SQL
 - Provides unlimited free licenses
 - Licenses do not “die” periodically
 - Supports many features usually available only in “pay for” products: e.g. PL/SQL debugging, custom reporting, and more!



- The better pay-for products are probably safe for a while...
- This product needs a logo!





- Oracle's SQL Developer tool is an excellent addition to every Oracle DBA or Developer's tool belt
- If Oracle keeps improving this product and keeps it free -- it may put some other products out of business



Training Days 2008

Mark your calendar for:

February 13-14, 2008!



ODTUG Kaleidoscope

June 18 – 21, 2007

Pre-conference Hands-on Training - June 16 – 17

Hilton Daytona Beach Oceanfront Resort

Daytona, Florida

WOW-Wide Open World, Wide Open Web!



IOUG-Collaborate in Denver, April 2008

Start Planning for Next Year!



Introducing Oracle's SQL Developer

To contact the author:

John King

King Training Resources

6341 South Williams Street

Littleton, CO 80121-2627 USA

1.800.252.0652 - 1.303.798.5727

Email: john@kingtraining.com



Thanks for your attention!

Today's slides and examples are on the web:

<http://www.kingtraining.com>