THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Oracle Database 12*c*: Use XML DB

Activity Guide

D81146GC10
Edition 1.0
October 2013
D84302

ORACLE

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Disclaimer

This document contains proprietary information and is protected by copyright and other intellectual property laws. You may copy and print this document solely for your own use in an Oracle training course. The document may not be modified or altered in any way. Except where your use constitutes "fair use" under copyright law, you may not use, share, download, upload, copy, print, display, perform, reproduce, publish, license, post, transmit, or distribute this document in whole or in part without the express authorization of Oracle.

The information contained in this document is subject to change without notice. If you find any problems in the document, please report them in writing to: Oracle University, 500 Oracle Parkway, Redwood Shores, California 94065 USA. This document is not warranted to be error-free.

Restricted Rights Notice

If this documentation is delivered to the United States Government or anyone using the documentation on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS
The U.S. Government's rights to use, modify, reproduce, release, perform, display, or disclose these training materials are restricted by the terms of the applicable Oracle license agreement and/or the applicable U.S. Government contract.

Trademark Notice

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Author

Lauran K. Serhal

Technical Contributors and Reviewers

Amitabh James Hans, Anjulaponni.A.S, Asha Tarachandani, Beda Hammerschmidt, Dan Melinger, Drew Adams, Geeta Arora, Hui Chang, James Spiller, Joe Greenwald, Lakshmi Narapareddi, Mark Drake, Nancy Greenberg, Qin Yu, Ravi Chennoju, Roger Ford, S. Matt Taylor Jr., Sriram Krishnamurthy, Srividya Tata, Vikas Arora, Viktor Tchemodanov, Ying Lu, Zhen Hua Liu

This book was published using: Oracle Tutor

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Table of Contents

Practices for Lesson 1: Introduction **1-1**
Practice 1-1: Using Oracle SQL Developer 1-2
Practice 1-2: XML Additional Resources 1-13
Practices for Lesson 2: XML Basic Review **2-1**
Practice 2-1: Testing Your Knowledge 2-2
Practice 2-2: Accessing and Using Additional XML Resources 2-5
Practices for Lesson 3: Introduction to Oracle XML DB **3-1**
Practices for Lesson 3 3-2
Practices for Lesson 4: Storing XML Data in Oracle XML DB **4-1**
Practice 4-1: Storing XML Data in Oracle XML DB 4-2
Practices for Lesson 5: Using XML Schema with Oracle XML DB **5-1**
Practice 5-1: Using XML Schemas in Oracle XML DB 5-2
Practice 5-2: Storing XML Data in Oracle XML DB 5-9
Practice 5-3: Managing Changes in an XML Schema 5-12
Practices for Lesson 6: Oracle XML DB Manageability **6-1**
Practices for Lesson 6: Overview 6-2
Practice 6-1: Annotating an XML Schema by Using the DBMS_XMLSCHEMA_ANNOTATE PL/SQL Package 6-3
Practices for Lesson 7: Partitioning XMLType Tables and Columns **7-1**
Practice 7-1: Annotating an XML Schema by Using the: Partitioning XML Data 7-2
Practices for Lesson 8: Using XQuery to Retrieve XML Data in Oracle XML DB **8-1**
Practice 8-1: Retrieving XML Data in Oracle XML DB 8-2
Practice 8-2: Using XQuery: Additional Use Cases 8-14
Practices for Lesson 9: Updating XML Content Using XQuery Update **9-1**
Practice 9-1: Updating XML Content by Using XQuery Update 9-2
Practice 9-2: Inserting an Element After Another Element 9-6
Practices for Lesson 10: Searching XML Content Using XQuery Full-Text **10-1**
Practices for Lesson 10 10-2
Practices for Lesson 11: Indexing XMLType Data **11-1**
Practice 11-1: Indexing XMLType Data 11-2
Practices for Lesson 12: Generating XML Data **12-1**
Practice 12-1: Generating XML 12-2
Practice 12-2: Generating XML Using DBMS_XMLGEN 12-15
Practices for Lesson 13: Transforming and Validating XMLType Data **13-1**
Practice 13-1: Transforming and Validating XML Data 13-2
Practices for Lesson 14: Working With the Oracle XML DB Repository **14-1**
Practice 14-1: Using Oracle XML DB Repository 14-2
Practice 14-2: Working with Access Control Lists (ACLs) 14-12
Practices for Lesson 15: Using Native Oracle XML DB Web Services **15-1**
Practice 15-1: Using Oracle XML DB Web Service 15-2
Practices for Lesson 16: Exporting and Importing XML Data **16-1**
Practice 16-1: Exporting and Importing XML Data 16-2
Practices for Lesson 17: Workshop **17-1**
Practice 17-1: Workshop Workshops 17-2

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 17-2: Workshop II 17-11

Practices for Lesson 18: Case Study 18-1

Practice 18-1: Creating a New Database Connection and Setting the Autotrace Parameters 18-2
Practice 18-2: Creating an XMLType Table and Loading Data 18-8
Practice 18-3: Using XQuery to Access XML Content 18-11
Practice 18-4: Optimizing XQuery Performance with XML Indexing 18-20
Practice 18-5: Optimizing XQuery Performance by Using a Path-Subsetted XML Index 18-23
Practice 18-6: Optimizing XQuery Performance by Using a Structured XMLIndex 18-31
Practice 18-7: Updating XML Content by Using XQuery-Update 18-36
Practice 18-8: Searching XML Content by Using XQuery Full Text 18-45
Practice 18-9: Optimizing XML Storage and Processing with XML Schema 18-56
Practice 18-10: Using XQuery with Object-Relational Storage 18-66
Practice 18-11: Indexing Object-Relational Storage to Optimize XQuery Operations 18-76
Practice 18-12: Equi-Partitioning of Object-Relational Storage 18-84
Practice 18-13: Relational Access to XML Content 18-92
Practice 18-14: XML Access to Relational Content 18-109
Practice 18-15: Creating a Persistent XML View of Relational Data 18-118

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 1: Introduction

Chapter 1

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practice 1-1: Using Oracle SQL Developer

The goal of this practice is to become familiar with the SQL Developer tool to execute SQL statements.

Note

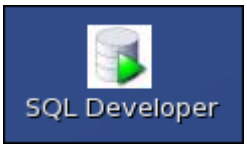
- All written practices use SQL Developer as the development environment. Although it is recommended that you use SQL Developer, you can also use the SQL*Plus or JDeveloper environments that are available in this course.
- All practices in this course and the practice solutions assume that you run the scripts by using the SQL Worksheet area in SQL Developer.

1. Create a database connection by using the information in the following table. Test the new connection. If the Status is Success, connect to the database by using the new connection.

Connection Name	`HR_Schema`
Username	`hr`
Password	`hr`
Save Password check box	Enabled (selected)
Hostname	`localhost`
Port	`1521`
SID	`orcl`

Steps:

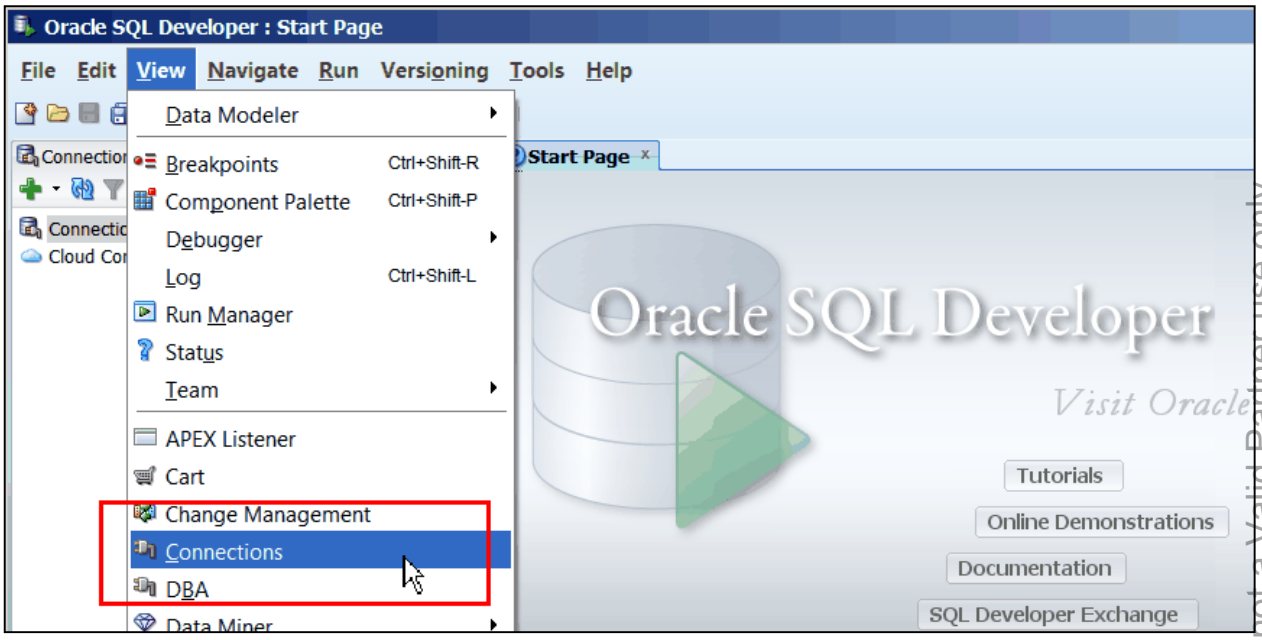
a. Open SQL Developer by double-clicking its icon on the desktop.



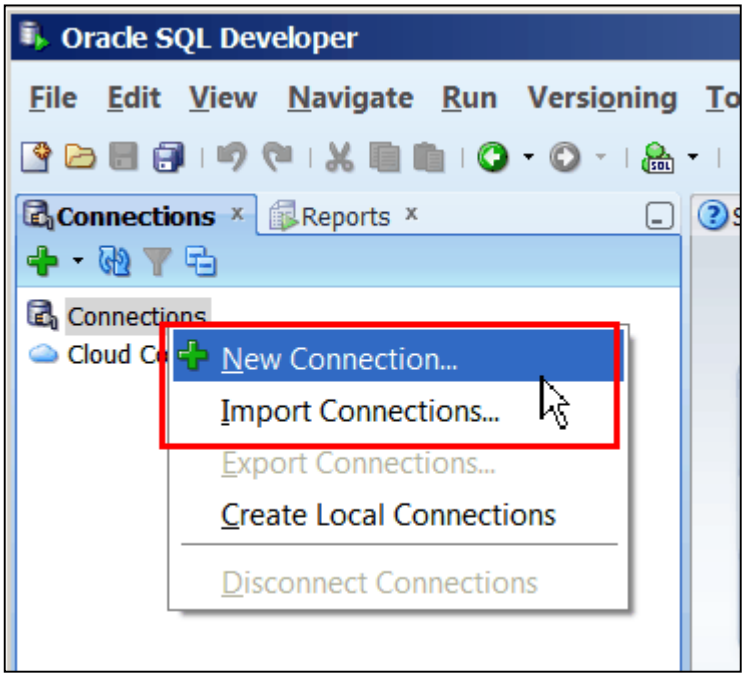
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

b. If the Connections tab is not already open, select View > Connections. The Connections window appears.



c. Right-click Connections and then select New Connection.

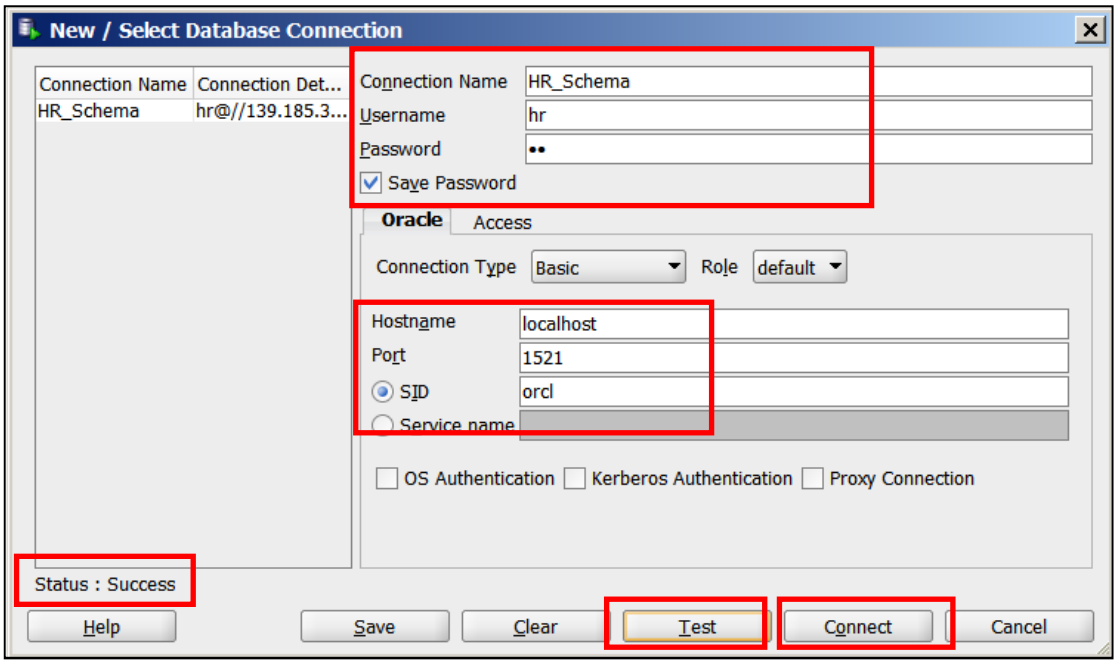


Oracle University and Error : You are not a Valid Partner use only

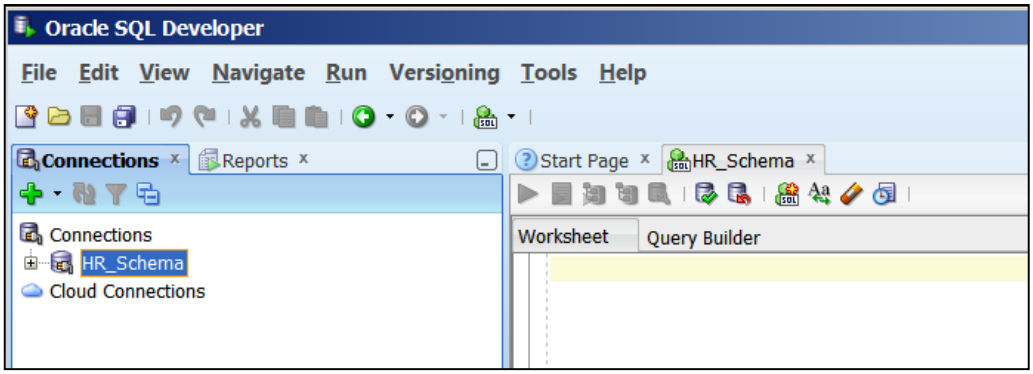
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

d. Enter the connection details (as shown in the screenshot), and then click Test to test the new connection. If the status is Success, click Connect.



e. The connection is established.



2. Create another connection by using the following information. Test the new connection. If the status is Success, connect to the database by using the new connection.

Connection Name	`OE_Schema`
Username	`oe`
Password	`oe`
Hostname	`localhost`
Save Password check box	Enabled (selected)
Port	`1521`
SID	`orcl`

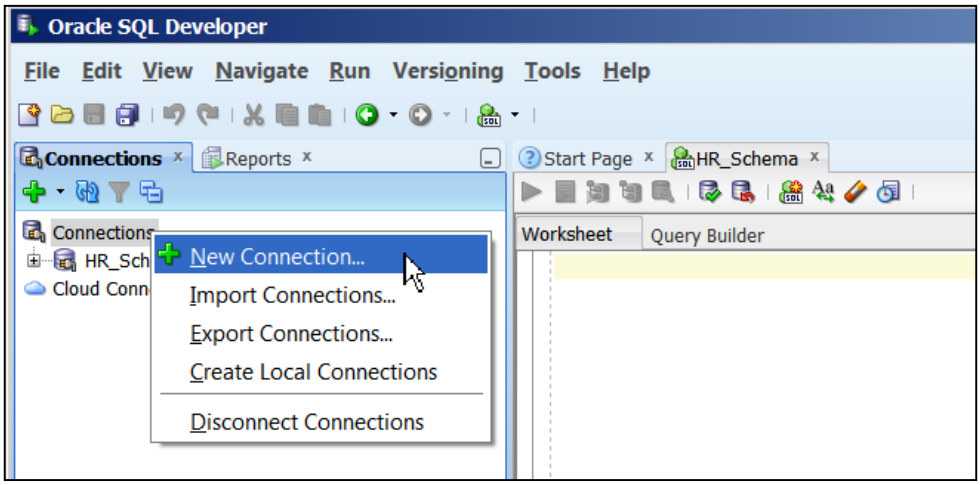
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

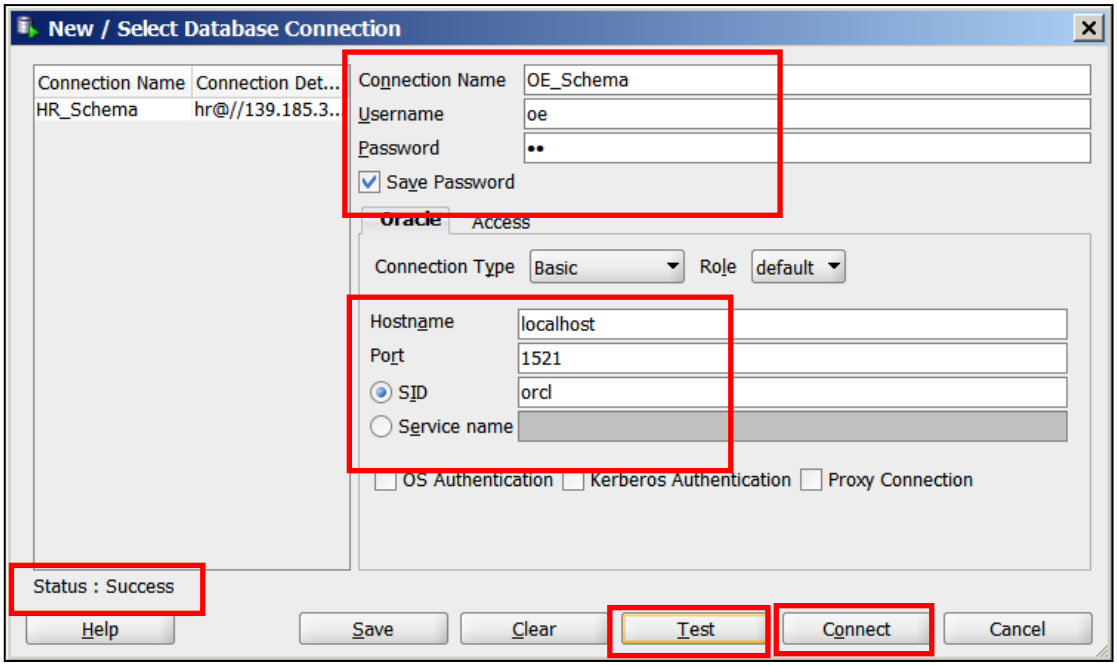
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Steps:

a. Right-click Connections in the Connections pane (View > Connections), and then select New Connection.



b. Enter the connection details, and then click Test. If the connection is successful, click Connect.

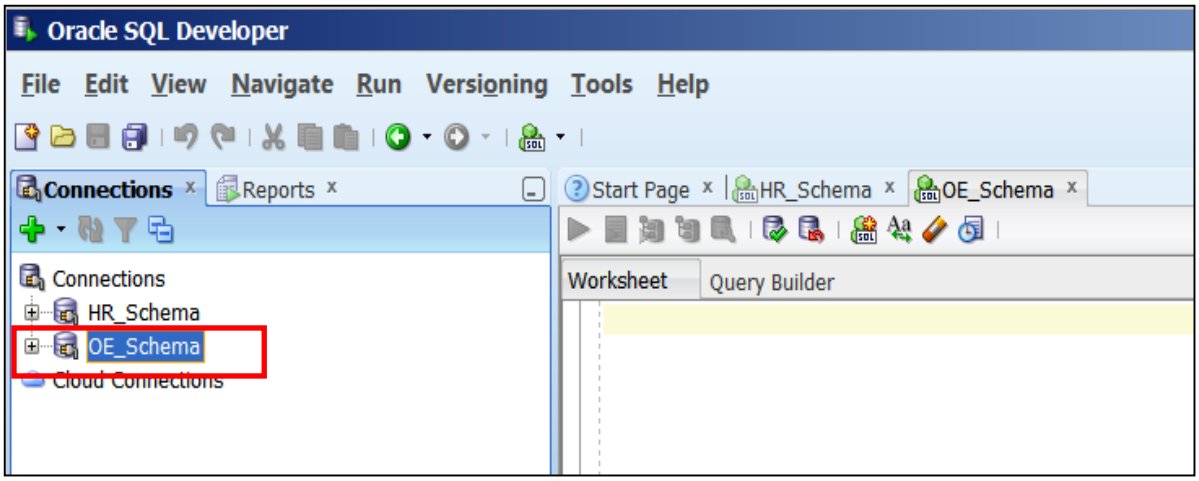


Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

c. The connection is established.



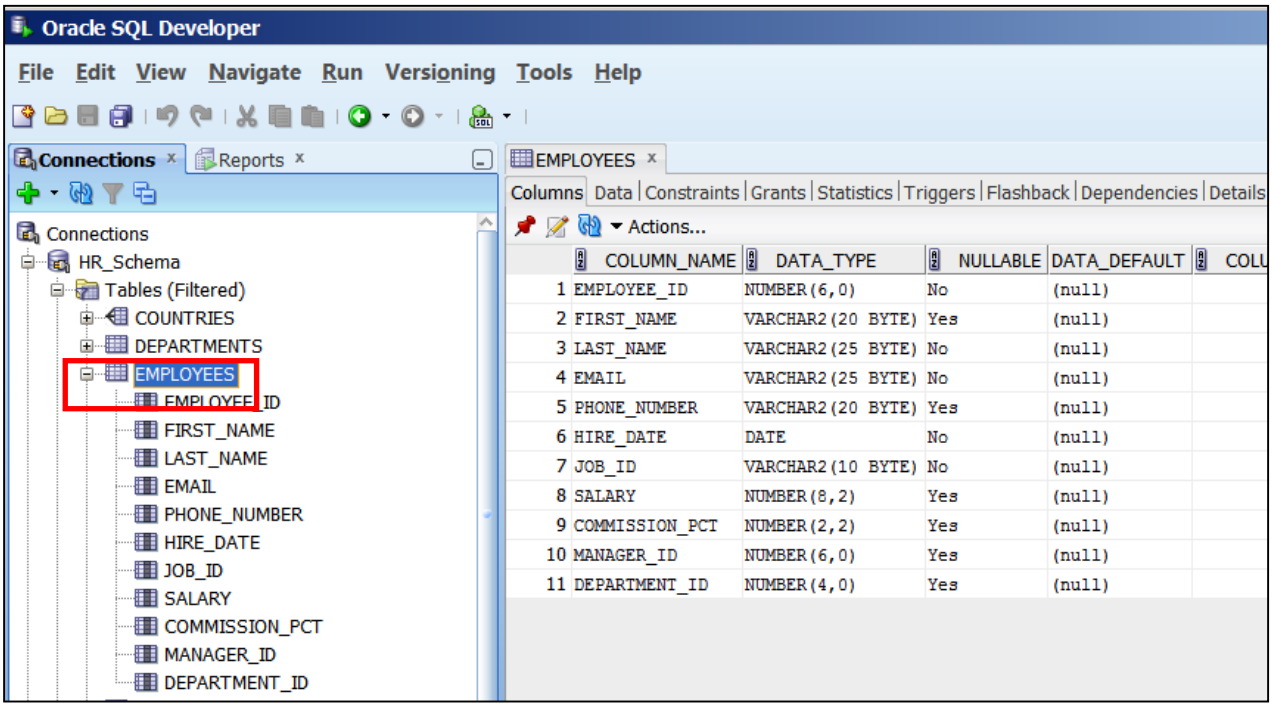
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. By using the Oracle SQL Developer graphical interface, view the table details in the `hr` and `oe` schemas, and review their structure.

   Expand the Tables node, and then click a table's name, such as `EMPLOYEES` to view its details.

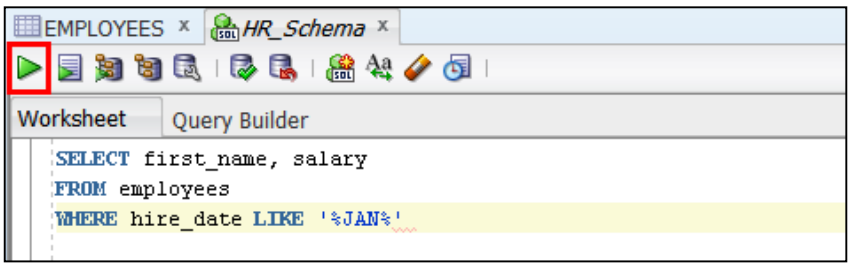


4. By using the SQL Worksheet, execute a SQL query to retrieve the first name and salary of all employees who were hired in the month of January from the `hr` schema. Save this file as `lab_01_04.sql` in the `/home/oracle/labs/lab` folder.

   Steps:

   a. Open a SQL Worksheet for the `HR` schema and enter the following query. Click the Execute Statement button (alternatively, press F9) to execute the query to see the result.

```
SELECT first_name, salary
FROM employees
WHERE hire_date LIKE '%JAN%';
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

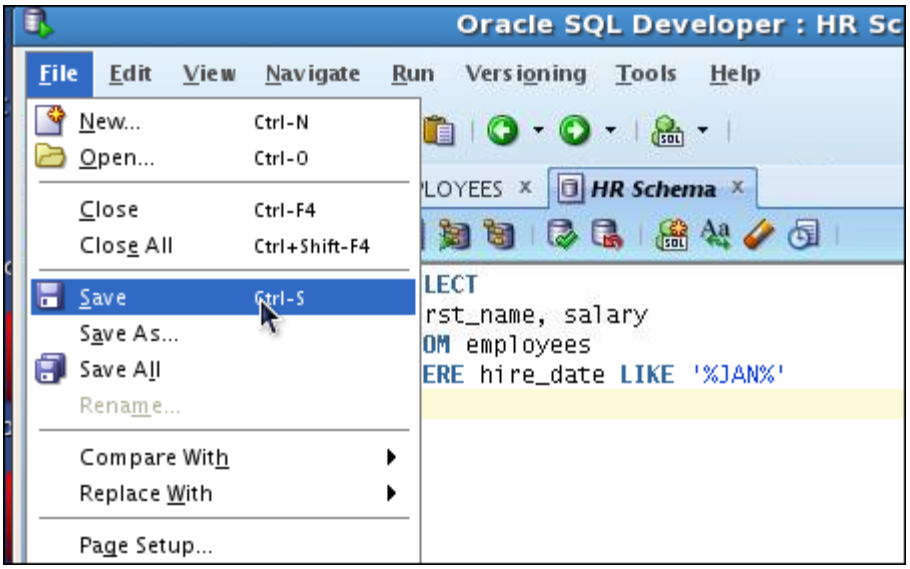
Output:

Query Result

SQL | All Rows Fetched: 14 in 3.025 seconds

	FIRST_NAME	SALARY
1	Lex	17000
2	Alexander	9000
3	James	2400
4	Curtis	3100
5	Karen	13500
6	Eleni	10500
7	Peter	10000
8	Janette	10000
9	Mattea	7200
10	Tayler	9600
11	Charles	6200
12	Winston	3200
13	Nandita	4200
14	Douglas	2600

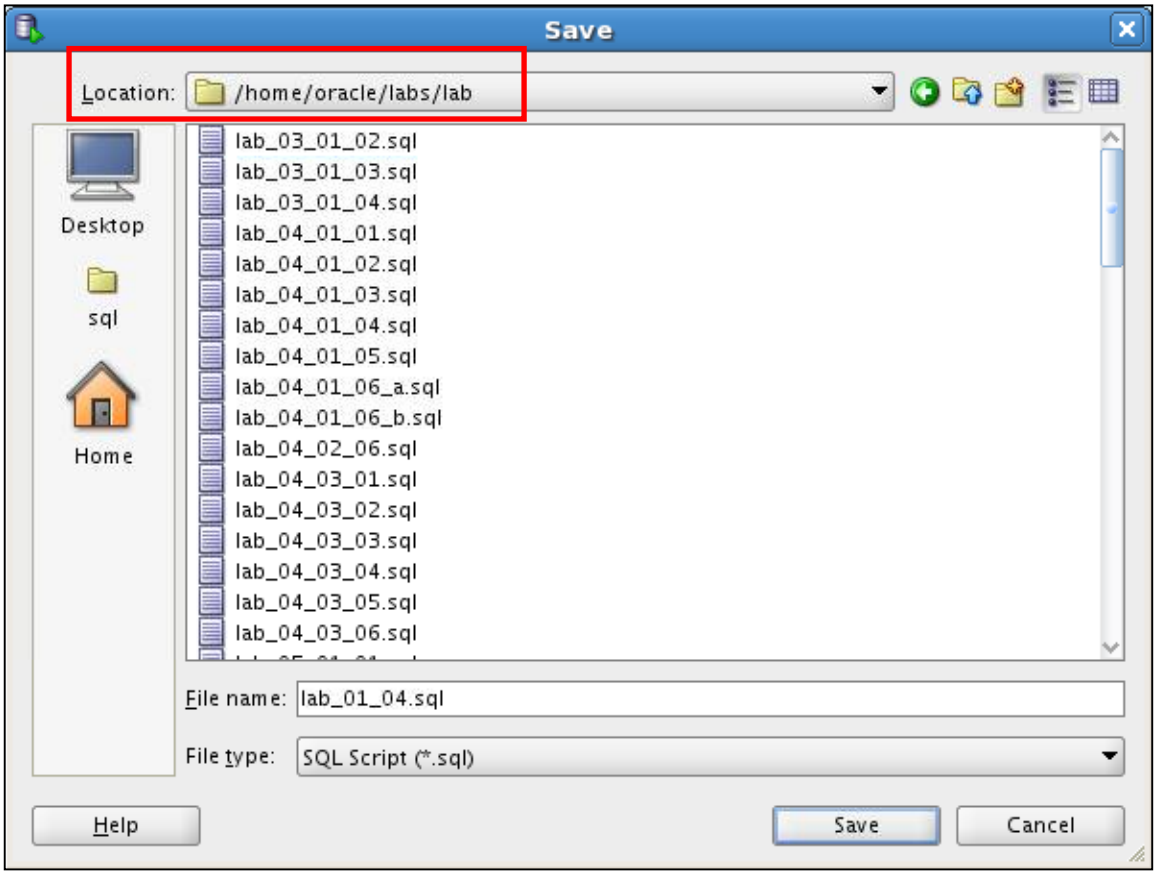
b. Click the SQL Worksheet where you entered the SQL query, and then select File > Save. In the dialog box that appears, browse to `/home/oracle/labs/lab` and save the file as `lab_01_04.sql`. Close the `lab_01_04.sql` tab after you save it.



Oracle University and Error : You are not a Valid Partner use only

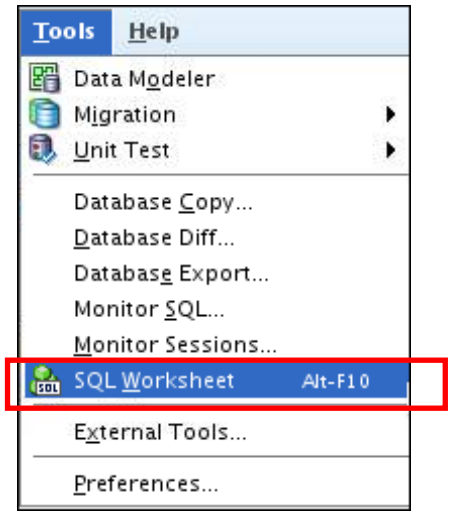
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



5. Open a new SQL Worksheet for the `hr` schema.

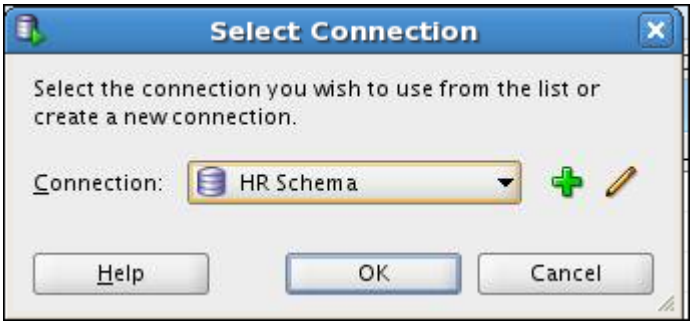
   Select Tools > SQL Worksheet to open a new worksheet. You are prompted to choose a connection. Select the `HR Schema` from the Connection drop-down list, and then click OK.



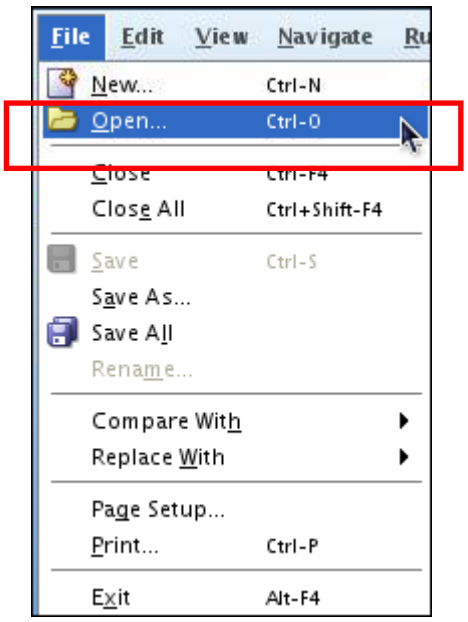
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



6. Run the `lab_01_04.sql` script saved in the `/home/oracle/labs/lab` folder.
   a. Click File > Open.

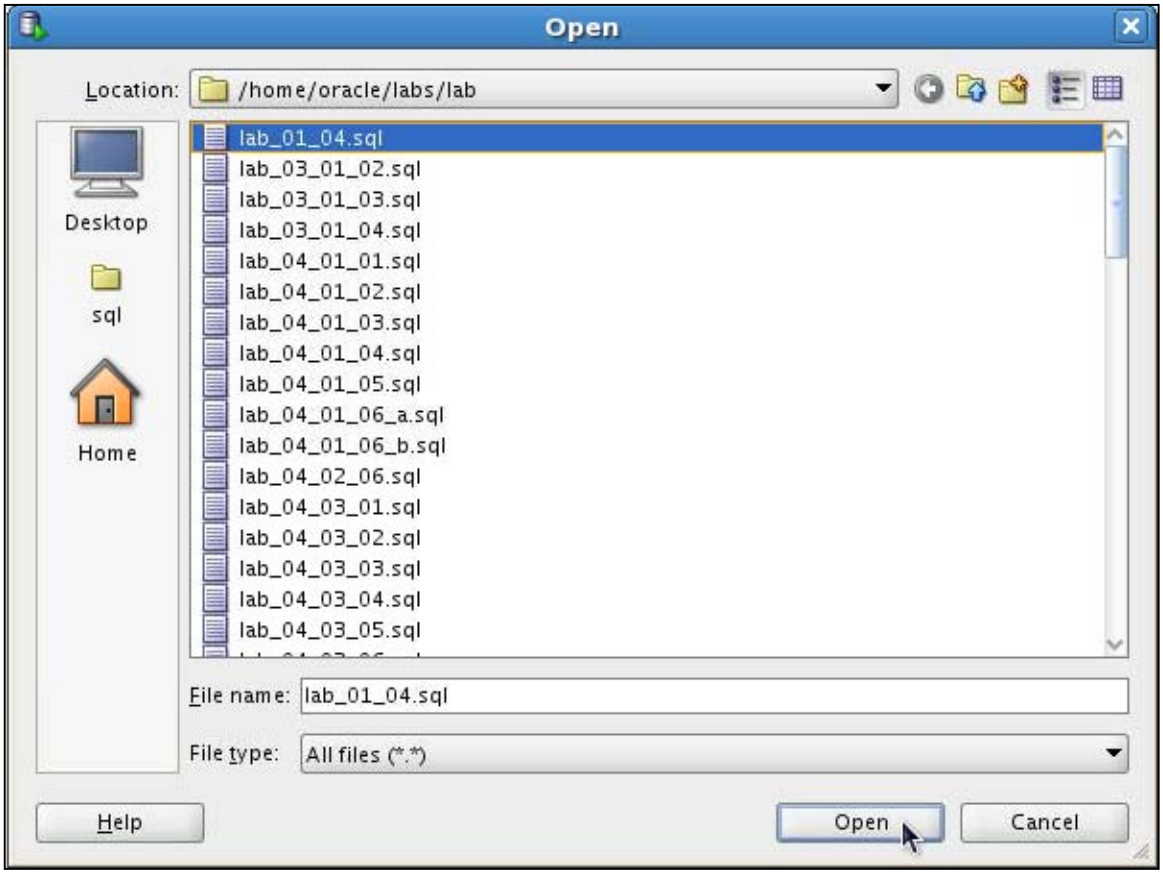


Oracle University and Error : You are not a Valid Partner use only

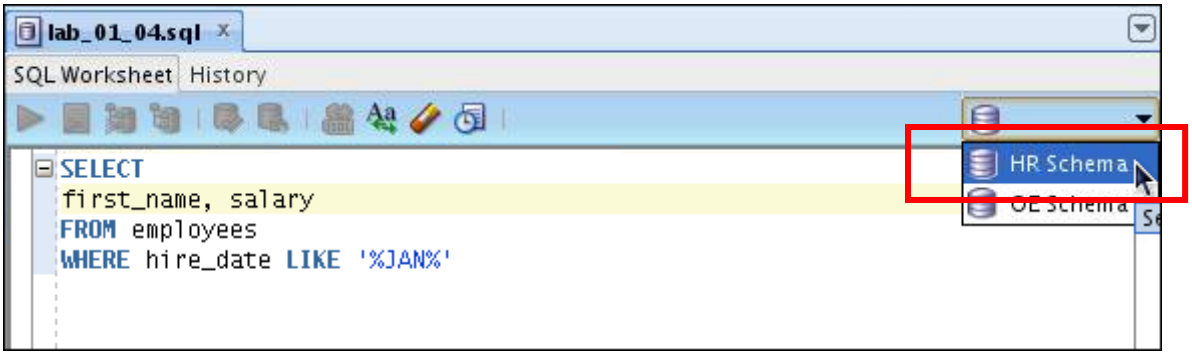
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

b. Browse to the `/home/oracle/labs/lab` folder and open the `lab_01_04.sql` file.



c. Select `HR_Schema` connection if not already selected.



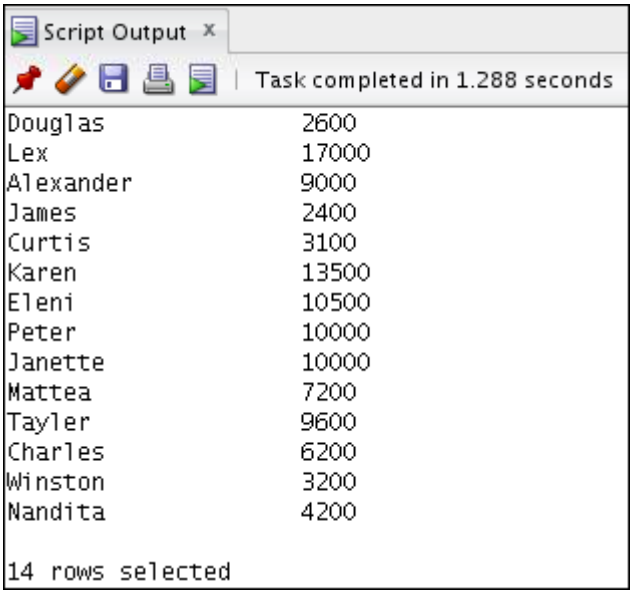
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

d. Click the Run Script button or press F5 to execute the script to see the result.

Output:



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 1-2: XML Additional Resources

Overview

In this practice, you use your web browser to access and review some of the useful XML resources and documentation, and then bookmark such resources for easier access.

Assumptions

Tasks

1. Access, review, and bookmark (on the Bookmarks toolbar) the XML DB home page at http://www.oracle.com/technetwork/database-features/xmldb/overview/index.html.

   Note: If your browser's bookmark is hidden, select View > Toolbars > Bookmarks Toolbar.

2. Access, review, and bookmark the Oracle Database documentation website at http://www.oracle.com/technology/documentation.
   a. Click the Database 12*c* Release 1 link in the Database section on the page.
   b. Click the Oracle Database, 12*c* Release 1 (12.1) link.
   c. Click the Master Book List link, and then bookmark this page.
   d. Click the XML link to view the available XML documentation references.
   e. Access and review the HTML version of the XML DB Developer's Guide. Add this guide to your Favorites list.
   f. Return to the Master Book List page and then access and review the HTML version of the Object-Relational Developer's Guide. Add this guide to your Favorites list.
   g. Access and review the Text Reference and the Text Application Developer's Guides. Add both guides to your Favorites list.
   h. Access and review the SQL*Plus User's Guide and Reference. Add this guide to your Favorites list.
   i. Click the Favorites link in the left frame of the Master Book List page to view the newly added favorites.

   Note: Your documentation Favorites are linked to your account name on the Oracle Technology Network. If you have selected some favorites but the list on this page is blank, or if you do not see the Add to Favorites links on the Master List of Books page, sign in to your OTN account or create one by using the Sign In or Register region.

3. Access and review the various products tutorials using Oracle Learning Library (OLL) at http://www.oracle.com/oll.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 2: XML Basic Review

Chapter 2

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 2-1: Testing Your Knowledge

The goal of this practice is to review some fundamental concepts of XML.

1. In the following XML data block, identify the following components:
   a. XML declaration
   b. Element
   c. Attribute
   d. Entity

```
<?xml version="1.0" encoding="UTF-8" ?>          → a

<departments>  → b

     <department id="100">  → c

          <manager>Rachael O&apos;Leary</manager>
                                  → d

     </department>
</departments>
```

2. Identify errors (if any) in the following XML documents:

```
<!-- This is a sample XML document -->
<?xml version="1.0"?>
<Rootelement>
     <1element>This is the first element</1element>
     <element2>This is the second element</element2>
</rootelement>
```

Errors:

a. A comment cannot be placed before the XML declaration statement.
b. The element name cannot start with a numerical.
c. Tag names are case-sensitive. Here, the casing for the start and end tags of the root element are not similar.

```
<?xml version="1.0" standalone="yes" encoding="UTF-8"?>
<Employee id=100>
     <name>
          <first_name>Sarah<last_name>Williams
          </first_name></last_name>
     </name>
</Employee>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Errors:

a. The `encoding` attribute of the XML declaration statement must be specified before the `standalone` attribute.

b. The value `100` of the `id` attribute must be specified within quotation marks.

c. The `first_name` and `last_name` tags must not overlap.

3. Write XPath expressions to locate the following in the given XML document: (There may be multiple ways to write an XPath expression.)

```
<?xml version="1.0" encoding=" UTF-8" ?>
<!DOCTYPE employees SYSTEM "employees.dtd">
<employees>
  <employee id='103'>
    <first_name>Alexander</first_name>
    <last_name>Hunold</last_name>
    <salary>9000<salary>
</employee>
…
  <employee>
    <first_name>Nancy</first_name>
    <last_name>Greenberg</last_name>
    <salary>12000<salary>
    </employee>
</employees>
```

a. The last employee element:

```
/employees/employee[last()]
employees/employee[last()]
//employee[last()]
```

b. The first names of the first two employees:

```
/employees/employee[position()<=first()+1]/first_name
employees/employee[position()<=first()+1]/first_name
//employee[position()<=first()+1]/first_name
```

c. The salary of the employee with employee ID 103:

```
/employees/employee[@id='103']/salary
employees/employee[@id='103']/salary
//employee[@id='103']/salary
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

d. The details of the employee with the last name Hunold:

```
/employees/employee[last_name='Hunold']
employees/employee[last_name='Hunold']
//employee[last_name='Hunold']
```

e. The comment for the root element:

```
employees/comment()
/employees/comment()
```

4. Oracle XML DB enables you to perform XML operations on relational data and SQL operations on XML data. **True** or False?
5. You can use `XMLType` views only when your data is XML. True or **False**?
6. **$** is the symbol used to start a variable name in XQuery expressions.
7. Rearrange the following in the order in which they appear in the syntax of the `FLWOR` expression syntax:

```
a. expr_containing_result
b. where
c. return
d. $identifier := expr(expr)
e. conditional_expr
f. order by
g. for
h. let
i. identifier("ascending" | "descending")
j. $identifier in expr(expr)
```

```
for
$identifier in expr(expr)
let
$identifier := expr(expr)
where
conditional_expr
order by
identifier("ascending" | "descending")
return
expr_containing_result
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practice 2-2: Accessing and Using Additional XML Resources

Overview

In this practice, you access some of the available XML resources.

Assumptions

Tasks

1. Access and review the XML Technology page at http://www.w3.org/standards/xml/.
2. Access and review the W3C XML Path Language (XPath) Version 1.0 web page at http://www.w3.org/TR/xpath/.
    a. Access the Abbreviated Syntax for XPath and review the list. Bookmark this page in the Bookmarks toolbar.
    b. Access and read the main sections of the “5 Data Model” section in the document.
3. Access and review the W3C XML Schema Part 0: Primer Second Edition web page at http://www.w3.org/TR/xmlschema-0/. Review the **2 Basic Concepts: The Purchase Order** section.
4. Access any of the other additional XML websites listed in the lesson.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 3: Introduction to Oracle XML DB

Chapter 3

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 3

There are no practices for this lesson.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 4: Storing XML Data in Oracle XML DB

Chapter 4

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practice 4-1: Storing XML Data in Oracle XML DB

The goal of this practice is to work with `XMLType`. Use SQL Developer for these tasks.

1. In SQL Developer, open a SQL Worksheet by using the `OE_Schema` database connection. Create a table to store employees' internal orders. The table must contain two columns: `employee_id` and `order_document`. The `employee_id` column must be numeric and the `order_document` column must be of `XMLType`. Name the table `internal_orders`.

```
CREATE TABLE internal_orders (
  employee_id NUMBER,
  order_document XMLTYPE);
```

```
table INTERNAL_ORDERS created.
```

2. By using `lab_04_01_02.sql`, insert some data into the table that was created in step 1 Execute the script in the `OE_Schema` schema.

```
Script Output
Task completed in 0.143 seconds
1 rows inserted
1 rows inserted
1 rows inserted
```

3. In SQL Developer, connect as `oe/oe`. Create an `XMLType` table `PRAC_PATH` and insert XML data into it by using `lab_04_01_03.sql`.

```
SET ECHO ON
create table prac_path of xmltype;
insert into prac_path values (
XMLType('<PurchaseOrder>
   <Actions>
      <Action>
         <Customer>SCOTT</Customer>
        </Action>
   </Actions>
   <Reject/>
   <Requestor>Julie P. Adams</Requestor>
   <User>ADAMS</User>
   <CostCenter>R20</CostCenter>
   <ShippingInstructions>
      <name>Julie P. Adams</name>
      <address>300 Oracle Parkway</address>
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
      <telephone>650 506 7300</telephone>
   </ShippingInstructions>
   <SpecialInstructions>Surface Mail</SpecialInstructions>
   <LineItems>
      <LineItem ItemNumber="1">
         <Description>Notorious</Description>
         <Part Id="715515012720" UnitPrice="39.95" Quantity="1"/>
      </LineItem>
      <LineItem ItemNumber="2">
         <Description>Samurai Three: Duel at Ganryu
Island</Description>
         <Part Id="037429125625" UnitPrice="29.95" Quantity="1"/>
      </LineItem>
      <LineItem ItemNumber="3">
         <Description>A Night to Remember</Description>
         <Part Id="715515009058" UnitPrice="39.95" Quantity="3"/>
      </LineItem>
      <LineItem ItemNumber="4">
         <Description>Time Bandits</Description>
         <Part Id="715515010122" UnitPrice="39.95" Quantity="3"/>
      </LineItem>
     </LineItems>
</PurchaseOrder>'));
```

```
> create table prac_path of xmltype
table PRAC_PATH created.
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
> insert into prac_path values (
XMLType('<PurchaseOrder>
   <Actions>
      <Action>
         <Customer>SCOTT</Customer>
        </Action>
   </Actions>
   <Reject/>
   <Requestor>Julie P. Adams</Requestor>
   <User>ADAMS</User>
   <CostCenter>R20</CostCenter>
   <ShippingInstructions>
      <name>Julie P. Adams</name>
      <address>300 Oracle Parkway</address>
      <telephone>650 506 7300</telephone>
   </ShippingInstructions>
   <SpecialInstructions>Surface Mail</SpecialInstructions>
   <LineItems>
      <LineItem ItemNumber="1">
         <Description>Notorious</Description>
         <Part Id="715515012720" UnitPrice="39.95" Quantity="1"/>
      </LineItem>
      <LineItem ItemNumber="2">
         <Description>Samurai Three: Duel at Ganryu Island</Description>
         <Part Id="037429125625" UnitPrice="29.95" Quantity="1"/>
      </LineItem>
      <LineItem ItemNumber="3">
         <Description>A Night to Remember</Description>
         <Part Id="715515009058" UnitPrice="39.95" Quantity="3"/>
      </LineItem>
      <LineItem ItemNumber="4">
         <Description>Time Bandits</Description>
         <Part Id="715515010122" UnitPrice="39.95" Quantity="3"/>
      </LineItem>
     </LineItems>
</PurchaseOrder>'))
1 rows inserted.
```

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Create an `XMLType` table with binary XML storage, and populate the table with data selected from the `PURCHASEORDERS` table. Run the `lab_04_01_04.sql` script in the `OE` Schema.

```
drop table po_sl_bix_table purge;
CREATE TABLE PO_SL_BIX_TABLE OF XMLType
     XMLType STORE AS BINARY XML;
/
insert into po_sl_bix_table
 select object_value from purchaseorder;
/
```

```
table PO_SL_BIX_TABLE created.
132 rows inserted.
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 5: Using XML Schema with Oracle XML DB

Chapter 5

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 5-1: Using XML Schemas in Oracle XML DB

In this practice, you use the `DBMS_XMLSCHEMA` package to manage the XML schemas in Oracle XML DB.

1. In SQL Developer, connect to the `HR_Schema`. Register the `DVDMovies.xsd` XML schema for binary XML. Open `lab_05_01_01.sql`. Replace ***<TODO>*** with the appropriate code and save the changes. Specify the schema URL as http://www.oracle.com/DVDMovies.xsd. Run `lab_05_01_01.sql` to register the `DVDMovies.xsd` schema.

   Answer:

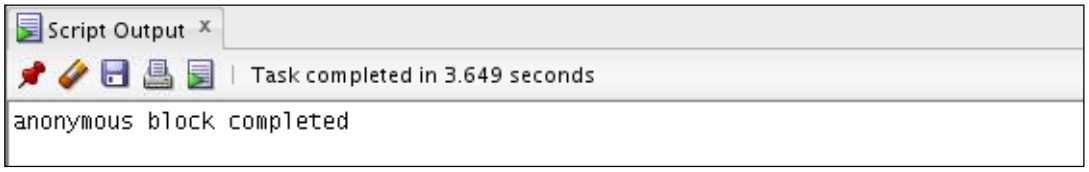
```
BEGIN
DBMS_XMLSCHEMA.registerSchema('http://www.oracle.com/DVDMovies.xs
d',
     '<schema xmlns="http://www.w3.org/2001/XMLSchema"
     targetNamespace="http://www.oracle.com/DVDMovies.xsd"
version="1.0"
      xmlns:xdb="http://xmlns.oracle.com/xdb"
     elementFormDefault="qualified">
     <element name = "DVD">
     <complexType>
     <sequence>
    <element name = "DVDId" type = "positiveInteger"/>
    <element name = "Name" type = "string"/>
    <element name = "Prod_Date" type = "date"/>
    <element name = "Price" type = "positiveInteger"/>
    <element name = "Taxes" type = "positiveInteger"/>
    <element name = "Category">
    <complexType>
    <sequence>
    <element name = "CatNo" type = "positiveInteger" />
    <element name = "CatName" type = "string"/>
    <element name = "Prod" type = "positiveInteger"/>
    </sequence>
    </complexType>
    </element>
    </sequence>
    </complexType>
    </element>
    </schema>',
    local => TRUE,
  genTypes => FALSE,
  genTables => FALSE,
  force => FALSE,
  OPTIONS => DBMS_XMLSCHEMA.REGISTER_BINARYXML);
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
    END;
   /
```



2. In SQL Developer, connect as `hr/hr`. Register the `members.xsd` schema. Open `lab_05_01_02.sql`. Replace ***<TODO>*** with the appropriate code and save the changes. Run `lab_05_01_02.sql`. Declare the `memberschema` variable as `VARCHAR2(2000)` to hold the XML schema definition.

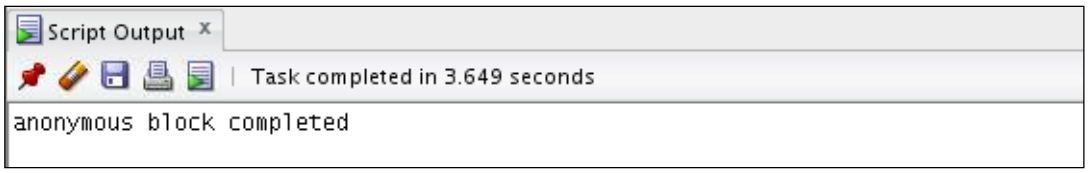
```
SET ECHO ON
--deletes the schema if already exists...
BEGIN
DBMS_XMLSCHEMA.deleteSchema(
SCHEMAURL => 'members.xsd',
DELETE_OPTION => dbms_xmlschema.DELETE_CASCADE_FORCE);
END;
/
DECLARE
memberschema VARCHAR2(2000) :=
'<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:xdb="http://xmlns.oracle.com/xdb">
<xs:complexType name="MemberADDRType" xdb:SQLType="Scheme_MADDR">
<xs:sequence>
<xs:element name="STREET" type="xs:string"/>
<xs:element name="CITY" type="xs:string"/>
<xs:element name="STATE" type="xs:string"/>
<xs:element name="ZIP" type="xs:string"/>
<xs:element name="COUNTRY" type="xs:string"/>
</xs:sequence>
</xs:complexType>
<xs:element name="SchemeMember">
<xs:complexType xdb:SQLType="Scheme_Member">
<xs:sequence>
<xs:element name="ID" type="xs:float" nillable="false"/>
<xs:element name="Member_NAME" type="xs:string"/>
<xs:element name="Member_TYPE" type="xs:string"/>
<xs:element name="Member_EMAIL" type="xs:string"/>
<xs:element name="Member_ADDR" type="MemberADDRType"/>
<xs:element name="PHONE" type="xs:string"/>
<xs:element name="JOINDATE" type="xs:date"/>
<xs:element name="DETAILS" type="xs:string"/>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>';
BEGIN
DBMS_XMLSCHEMA.RegisterSchema('members.xsd', memberschema);
END;
```



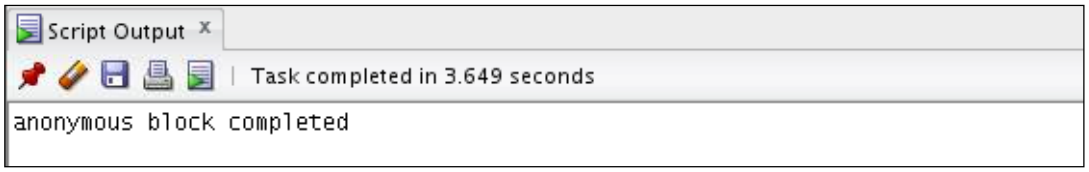
3. In SQL Developer, connect as `hr/hr`. Open `lab_05_01_03.sql`. Replace ***<TODO>*** with the appropriate code. Specify the schema URL as `Schemes.xsd`. Save the changes that you made and run `lab_05_01_03.sql` to register the `Schemes.xsd` XML schema.

```
SET ECHO ON
--deletes the schema if already exists...
DECLARE
Schemeschema VARCHAR2(2000) := '<xs:schema
xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xdb="http://xmlns.oracle.com/xdb">
    <xs:element name="Scheme">
    <xs:complexType xdb:SQLType="Sch">
    <xs:sequence>
    <xs:element name="Scheme_No" type="xs:float"
nillable="false"/>
    <xs:element name="Scheme_Id" type="xs:float"/>
    <xs:element name="Amt" type="xs:float" default="0"
nillable="false"/>

    <xs:element name="Date" type="xs:date"/>
    <xs:element name="Scheme_Type" type="xs:string"/>
    </xs:sequence>
    </xs:complexType>
    </xs:element>
    </xs:schema>';
    BEGIN
      DBMS_XMLSCHEMA.RegisterSchema('Schemes.xsd', SchemeSchema
);
END;
```

Oracle University and Error : You are not a Valid Partner use only

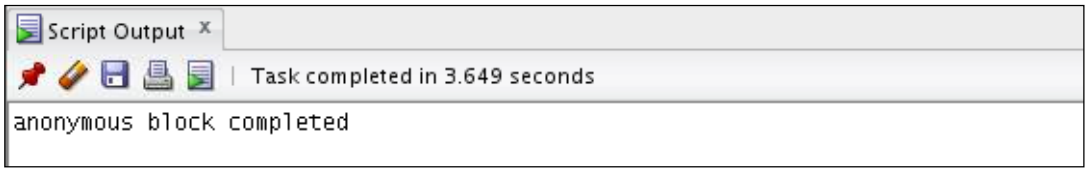
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



4. In SQL Developer, connect as `oe/oe`. Register the `prac_po_bin.xsd` XML schema for binary XML usage. Open `lab_05_01_04.sql`. Replace ***<TODO>*** with the appropriate code to register the XML schema. Specify the schema URL as http://xmlns.oracle.com/xsd/prac_po_bin.xsd. Save the changes and run `lab_05_01_04.sql`.

```
BEGIN
  DBMS_XMLSCHEMA.registerSchema(
  SCHEMAURL =>
  'http://xmlns.oracle.com/xsd/prac_po_bin.xsd',
  SCHEMADOC =>
  bfilename('XML_DIR','prac_po_bin.xsd'),
  CSID => nls_charset_id('AL32UTF8'),
  local => TRUE,
  genTypes => FALSE,
  genTables => FALSE,
  force => FALSE,
  OPTIONS => DBMS_XMLSCHEMA.REGISTER_BINARYXML
  );
  END;
  /
```



5. In SQL Developer, connect as `hr/hr`. Open `lab_05_01_05.sql`. Replace <***TODO***> with the appropriate code to register the `new_emp_bin.xsd` XML schema for binary XML. Save the changes and run `lab_05_01_05.sql`.

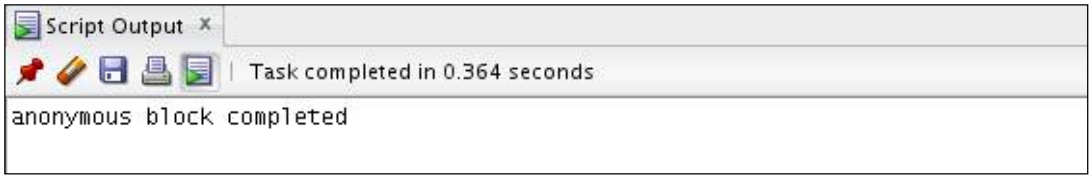
```
BEGIN
DBMS_XMLSCHEMA.registerSchema('http://www.oracle.com/new_emp_bin.
xsd',
'<schema xmlns="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://www.oracle.com/new_emp_bin.xsd"
version="1.0"
xmlns:xdb="http://xmlns.oracle.com/xdb"
elementFormDefault="qualified">
<element name = "Employee">
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
<complexType>
<sequence>
<element name = "EmployeeId" type = "positiveInteger"/>
<element name = "Name" type = "string"/>
<element name = "Job" type = "string"/>
<element name = "Manager" type = "positiveInteger"/>
<element name = "HireDate" type = "date"/>
<element name = "Salary" type = "positiveInteger"/>
<element name = "Commission" type = "positiveInteger"/>
<element name = "Dept">
<complexType>
<sequence>
<element name = "DeptNo" type = "positiveInteger" />
<element name = "DeptName" type = "string"/>
<element name = "Location" type = "positiveInteger"/>
</sequence>
</complexType>
</element>
</sequence>
</complexType>
</element>
</schema>',
local => TRUE,
  genTypes => FALSE,
  genTables => FALSE,
  force => FALSE,
  OPTIONS => DBMS_XMLSCHEMA.REGISTER_BINARYXML
  );
END;
```



6. In SQL Developer, connect as `hr/hr`.
   a. Run `lab_05_01_06_a.sql` to register `emp_reports.xsd`.

```
BEGIN
DBMS_XMLSCHEMA.registerSchema('http://www.oracle.com/emp_reports
.xsd',
'<schema xmlns="http://www.w3.org/2001/XMLSchema"
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
targetNamespace="http://www.oracle.com/emp_reports.xsd"
version="1.0"
xmlns:xdb="http://xmlns.oracle.com/xdb"
elementFormDefault="qualified">
<element name = "Employee">
<complexType>
<sequence>
<element name = "EmployeeId" type = "positiveInteger"/>
<element name = "Name" type = "string"/>
<element name = "Job" type = "string"/>
<element name = "Manager" type = "positiveInteger"/>
<element name = "HireDate" type = "date"/>
<element name = "Salary" type = "positiveInteger"/>
<element name = "Commission" type = "positiveInteger"/>
<element name = "Dept">
<complexType>
<sequence>
<element name = "DeptNo" type = "positiveInteger" />
<element name = "DeptName" type = "string"/>
<element name = "Location" type = "positiveInteger"/>
</sequence>
</complexType>
</element>
</sequence>
</complexType>
</element>
</schema>',
TRUE,
FALSE,
FALSE,
FALSE
    );
END;
/
```

```
Script Output
Task completed in 0.364 seconds
anonymous block completed
```

b. Open `lab_05_01_06_b.sql`. Replace <***TODO***> with the appropriate code to delete `emp_reports.xsd` by using the `DELETE_CASCADE_FORCE` mode.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

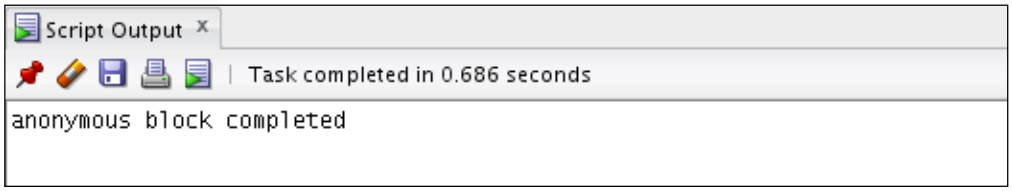
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

c. Query the data dictionary with the following command to confirm that the `emp_reports.xsd` schema exists.

```
SELECT schema_url
FROM user_xml_schemas;
```

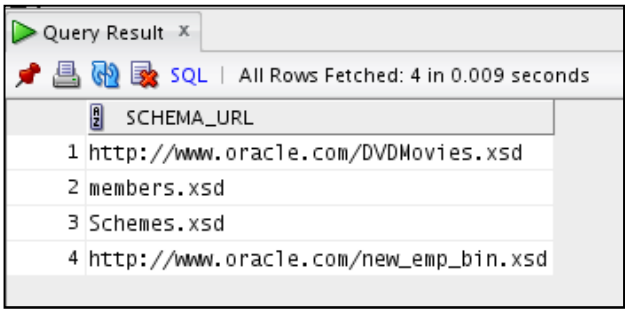
d. Save and run the script. Connect as `hr/hr`.

```
BEGIN
  DBMS_XMLSCHEMA.deleteSchema(
  SCHEMAURL => 'http://www.oracle.com/emp_reports.xsd',
  DELETE_OPTION => dbms_xmlschema.DELETE_CASCADE_FORCE);
END;
/
```



Note: Confirm the deletion by querying the data dictionary with the following command:

```
SELECT schema_url
FROM user_xml_schemas;
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 5-2: Storing XML Data in Oracle XML DB

The goal of this practice is to work with XML schemas in `XMLType` tables. Use SQL Developer for these tasks.

1. In SQL Developer, connect as `hr/hr`. Create an `XMLType` table, `DVD_tab`, where the contents of `XMLType` are constrained by the previously registered XML schema `DVDMovies.xsd` for binary XML and the contents of `XMLType` are stored by using binary XML. Note that the default storage type is binary XML so you can omit that part. This example and others include that clause for clarity purposes only.

```
CREATE TABLE DVD_tab of XMLType
XMLType STORE AS BINARY XML
XMLSCHEMA "http://www.oracle.com/DVDMovies.xsd"
ELEMENT "DVD";
```

```
table DVD_TAB created.
```

2. In SQL Developer, connect as `hr/hr`. Create the `members` table, which is of the `XMLType` type that is constrained to the `SchemeMember` element defined by the `members.xsd` schema. In the `CREATE TABLE` statement, specify the `UNIQUE` constraint `UK` on the `ID` column.

```
CREATE TABLE members of XMLType(CONSTRAINT UK
unique(xmldata."ID"))XMLSCHEMA "members.xsd" ELEMENT
"SchemeMember";
```

```
table MEMBERS created.
```

3. In SQL Developer, connect as `oe/oe`. Create the `prac_po_tab1` table, which is of the `XMLType` type that is constrained to the `PurchaseOrder` element defined by the `prac_po_bin.xsd` schema. In the `CREATE TABLE` statement, specify the `STORE AS BINARY XML` option.

```
CREATE TABLE prac_po_tab1 OF XMLType
XMLType STORE AS BINARY XML
XMLSCHEMA
"http://xmlns.oracle.com/xsd/prac_po_bin.xsd"
ELEMENT"PurchaseOrder";
```

```
table PRAC_PO_TAB1 created.
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

4. In SQL Developer, connect as `oe/oe`. Create an `XMLType` table for binary XML storage. Specify the `SECUREFILE` option. Name the table `prac_secure_tab`.

```
CREATE TABLE prac_secure_tab OF XMLType
XMLType STORE AS SECUREFILE BINARY XML;
```

```
table PRAC_SECURE_TAB created.
```

5. In SQL Developer, connect as `oe/oe`. Create the `prac_po_tab2` table, which is of the `XMLType` type that is constrained to the `PurchaseOrder` element defined by the `prac_po_bin.xsd` schema. In the `CREATE TABLE` statement, specify the `STORE AS BINARY XML` option. This table should store non-schema-based XML documents as well.

```
CREATE TABLE prac_po_tab2 OF XMLType
XMLType STORE AS BINARY XML
XMLSCHEMA
"http://xmlns.oracle.com/xsd/prac_po_bin.xsd"
ELEMENT"PurchaseOrder"
ALLOW NONSCHEMA;
```

```
table PRAC_PO_TAB2 created.
```

6. In SQL Developer, connect as `hr/hr`. Create the `schemes_tab` table of the `XMLType` type with the following requirements:
   a. `UNIQUE` constraint `UK_Key` on the `Scheme_No` column
   b. `FOREIGN KEY` constraint `VALID` on the `Scheme_Id` column with reference to the `ID` column in the `members` table that was created in question 4.

```
CREATE TABLE succeeded.
```

Use `lab_05_02_06.sql` for help. Replace <***TODO***> with the appropriate code, save the changes, and then run `lab_05_02_06.sql`.

Before editing:

```
CREATE TABLE schemes_tab OF XMLType( CONSTRAINT "UK_Key" UNIQUE
(xmldata."Scheme_No"),
CONSTRAINT "VALID" FOREIGN KEY <TODO>
REFERENCES members(xmldata."ID")) <TODO> ELEMENT "Scheme";
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

After editing:

```
CREATE TABLE schemes_tab OF XMLType( CONSTRAINT "UK_Key" UNIQUE
(xmldata."Scheme_No"),
CONSTRAINT "VALID" FOREIGN KEY (xmldata."Scheme_Id")
REFERENCES members(xmldata."ID")) XMLSCHEMA "Schemes.xsd"
ELEMENT "Scheme";
```

```
table SCHEMES_TAB created.
```

7. In SQL Developer, connect as `hr/hr`. Create `employee_bin`, an `XMLType` table that is constrained to the previously registered `new_emp_bin.xsd` XML schema for binary XML. The table should hold even non-schema-based XML documents.

```
CREATE TABLE employee_bin OF XMLType
XMLType STORE AS BINARY XML
XMLSCHEMA
"http://www.oracle.com/new_emp_bin.xsd"
ELEMENT"Employee"
ALLOW NONSCHEMA;
```

```
table EMPLOYEE_BIN created.
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 5-3: Managing Changes in an XML Schema

In this practice, you modify the `applicantDetails.xsd` XML schema. By using the `dbms_xmlschema.CopyEvolve` package's procedure, you evolve `new_applicantDetails.xsd`.

To perform all the exercises in this practice, use a terminal window to start up SQL*Plus, and then log in as `hr/hr`. Exit SQL Developer.

1. Review the XML schema definition, and then run `lab_05_03_01.sql` to register the `applicantDetails.xsd` XML schema.

   Answer:

```
--deletes the schema if already exits
...
declare
    res boolean;
    xmlschema xmltype := xmltype(
'<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xdb="http://xmlns.oracle.com/xdb">
     <xsd:complexType name="app" xdb:SQLType="app_t">
             <xsd:sequence>
                    <xsd:element name="appdetails"
type="appdetailsType"/>
                      <xsd:element name="testdetails"
type="testdetailsType"/>
                    <xsd:element name="acadetails"
type="acadetailsType"/>
             </xsd:sequence>
     </xsd:complexType>
     <xsd:complexType name="appdetailsType"
xdb:SQLType="appdetails_T">
            <xsd:sequence>
                    <xsd:element name="app_name"
type="xsd:string"/>
                    <xsd:element name="app_years"
type="xsd:decimal"/>
            </xsd:sequence>
     </xsd:complexType>
     <xsd:complexType name="testdetailsType"
xdb:SQLType="testdetails_T">
            <xsd:sequence>
                    <xsd:element name="test_date"
type="xsd:date"/>
                     <xsd:element name="test_others"
type="xsd:string"/>
            </xsd:sequence>
    </xsd:complexType>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
    <xsd:complexType name="acadetailsType"
xdb:SQLType="acadetails_T">
           <xsd:sequence>
                      <xsd:element name="years"
type="xsd:decimal"/>
                       <xsd:element name="projects"
...
```

```
PL/SQL procedure successfully completed.

SQL>
SQL>  begin
  2      dbms_xmlschema.registerSchema ('www.applicant-details.com',xdburitype('
/public/applicantDetails.xsd'),TRUE,TRUE,FALSE,TRUE);
  3      end;
  4    /

PL/SQL procedure successfully completed.

SQL> ▌
```

2. Create the `app_tab` table with the following definitions:
   a. `appId VARCHAR2(100) CONSTRAINT pk_app PRIMARY KEY`
   b. `information VARCHAR2(20)`
   c. `eval_date DATE`
   d. `applicant_part xmltype`

   Run `lab_05_03_02.sql` to create the `app_tab` table.

   Answer:

```
--drops table if already exists
...
CREATE TABLE app_tab
  (
    appId       VARCHAR2(100) CONSTRAINT pk_app PRIMARY KEY,
    information VARCHAR2(20),
    eval_date DATE,
    applicant_part xmltype
  )
  XMLTYPE COLUMN applicant_part XMLSCHEMA "www.applicant-
details.com" element
  "applicant";
```

```
Table created.

SQL> ▌
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. Insert sample data into the `app_tab` table and query for the number of rows. Run `lab_05_03_03.sql`.

Answer:

```
--Execute as HR
SET ECHO ON
insert into app_tab(appId ,information ,eval_date
,applicant_part)
   values ('C111','Preliminary','14-june-2006',
   XMLType(
  '<?xml version="1.0" encoding="UTF-8"?>
   <applicant xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:noNamespaceSchemaLocation="www.applicant-
details.com">
      <appdetails>
           <app_name>Smith</app_name>
             <app_years>6</app_years>
       </appdetails>
     <testdetails>
              <test_date>2006-06-14</test_date>
              <test_others>3</test_others>
    </testdetails>
     <acadetails>
             <years>14</years>
             <projects>two</projects>
     </acadetails>
  </applicant>'))
  /

select count(*) from app_tab;
```

Output:

```
1 row created.

SQL>
SQL>
SQL> select count(*) from app_tab;

  COUNT(*)
----------
         1

SQL>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Modify the schema definition of `applicantDetails.xsd`. Run `lab_05_03_04.sql` to add the `location` element to the schema definition.

Answer:

```
declare
   xmlschema xmltype :=
xdburitype('/public/applicantDetails.xsd').getXML();
res boolean;
begin
  select insertChildXML(xmlschema,
'/xsd:schema/xsd:complexType[@name="testdetailsType"]/xsd:sequen
ce','xsd:element', xmltype('<xsd:element
xmlns:xsd="http://www.w3.org/2001/XMLSchema" name="location"
type="xsd:string"/>'),
            'xmlns:xsd="http://www.w3.org/2001/XMLSchema"'
           )
  into xmlSchema
  from dual;

if
(dbms_xdb_repos.existsResource('/public/new_applicantDetails.xsd
'))
then
dbms_xdb_repos.deleteResource('/public/new_applicantDetails.xsd'
);
end if;

res :=
dbms_xdb_repos.createResource('/public/new_applicantDetails.xsd'
,xmlschema);
 end;
  /
```

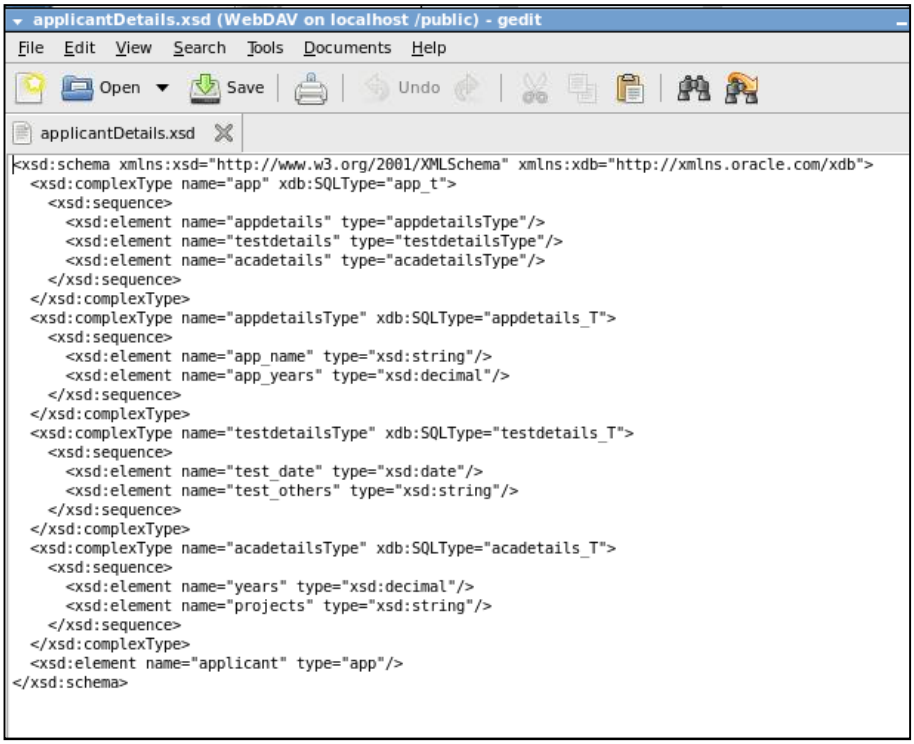
```
PL/SQL procedure successfully completed.

SQL>
SQL>
```

The following is the `applicantDetails.xsd` XML DB repository resource in the `/public` folder. This resource was created in step 1 of this practice when you registered the XML schema.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



5. Evolve the `new_applicantDetails.xsd` XML schema.

   Open the `lab_05_03_05.sql` file. Insert the missing code to evolve the `new_applicantDetails.xsd` XML schema. Save the changes and run `lab_05_03_05.sql`.

```
SET ECHO ON
begin
     <TODO>
    (
        xdb$string_list_t('www.applicant-details.com'),

XMLSequenceType(xdburitype('/public/new_applicantDetails.xsd').g
etXML()),
       null
      );
   end;
  /
```

Answer:

Replace <***TODO***> with **`dbms_xmlschema.CopyEvolve`**. Then, save the file as `lab_05_03_05.sql`.

Run `lab_05_03_05.sql`. Alternatively, you can use the `sol_05_03_05.sql` script. Be patient because running this script might take few moments.

```
begin
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

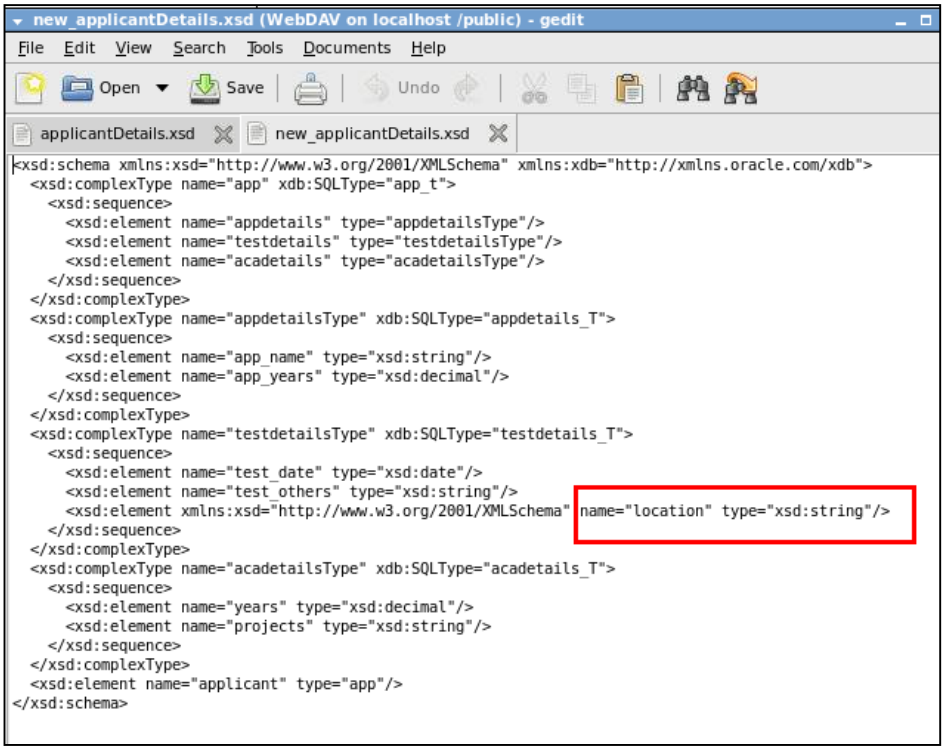
```
dbms_xmlschema.CopyEvolve
  (
     xdb$string_list_t('www.applicant-details.com'),

XMLSequenceType(xdburitype('/public/new_applicantDetails.xsd').g
etXML()),
null
);
end;
/
```

Now, query for the number of rows in the `app_tab` table.

Output:

```
PL/SQL procedure successfully completed.

SQL> select count(*) from app_tab;

  COUNT(*)
----------
         1

SQL>
```

The following is the `new_applicantDetails.xsd` XML DB repository resource in the `/public` folder. This resource was created in step 4 of this practice when you evolved the XML schema. Note the newly added `location` element.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

6. Insert another set of sample data into the `app_tab` table. Observe that the location element is included this time.
   Open the `lab_05_03_06.sql` file. Insert the value for **<*TODO*>** as **`<location>San Fransisco</location>`**. Save the changes, and then run `lab_05_03_06.sql`.

```
...
            <app_name>Reynold</app_name>
                 <app_years>5</app_years>
          </appdetails>
        <testdetails>
                 <test_date>2006-06-12</test_date>
                 <test_others>4</test_others>
                 <TODO>
       </testdetails>
        <acadetails>
                <years>10</years>
                <projects>three</projects>
        </acadetails>
     </applicant>'))
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Answer:

Replace <***TODO***> with the following and save the file as `lab_05_03_06.sql`.

```
...
            <app_name>Reynold</app_name>
                 <app_years>5</app_years>
          </appdetails>
        <testdetails>
                 <test_date>2006-06-12</test_date>
                 <test_others>4</test_others>
                 <location>San Fransisco</location>
       </testdetails>
        <acadetails>
                <years>10</years>
                <projects>three</projects>
        </acadetails>
     </applicant>'))
/
```

Run `lab_05_03_06.sql`. Alternatively, you can use the `sol_05_03_06.sql` script.

Answer:

```
1 row created.
```

7. Commit your changes. Click the Commit (F11) icon on the SQL Worksheet toolbar if you are using SQL Developerl; enter `commit;` on the command line if you are using SQL*Plus.

```
SQL> commit;

Commit complete.
```

8. Now, query the `app_tab` table for the applicant's ID and the location information. What do you observe?

Answer:

Query Result

All Rows Fetched: 2 in 0.021 seconds

	APPID	EXTRACTVALUE(APPLICANT_PART,'/APPLICANT/TESTDETAILS/LOCATION')
1	C111	(null)
2	C121	San Fransisco

With the schema evolution, you can see the existing data that is populated into the `app_tab` table along with the newly added value for the new `location` element as per the definition of `new_applicantDetails.xsd`.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 6: Oracle XML DB Manageability

Chapter 6

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practices for Lesson 6: Overview

Practices Overview

In this practice, you create a new table to store an un-annotated XML schema and then annotate the XML schema by using the PL/SQL package `DBMS_XMLSCHEMA_ANNOTATE`.

After you register the annotated XML schema, you verify the creation of the tables, types, and other objects that were created after the XML schema registration.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 6-1: Annotating an XML Schema by Using the `DBMS_XMLSCHEMA_ANNOTATE` PL/SQL Package

Overview

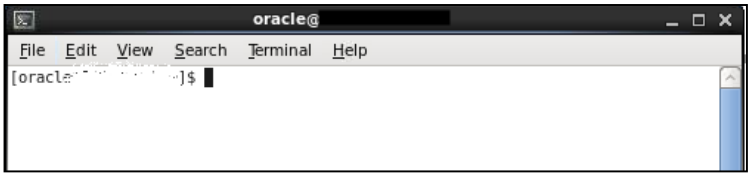
In this practice, you create a new table to store an un-annotated XML schema and then annotate the XML schema by using the PL/SQL package `DBMS_XMLSCHEMA_ANNOTATE`.

After you register the annotated XML schema, you verify the creation of the tables, types, and other objects that were created after the XML schema registration.

Assumptions

Tasks

1. Open a SQL*Plus terminal window.



2. Log in to SQL*Plus as `system/oracle`, and then create a new user `oe6/oe6` for this practice. Grant the new user the `dba` privilege just for the purpose of this practice.

```
% sqlplus system/oracle as sysdba
SQL> create user oe6 identified by oe6;
SQL> grant dba to oe6 identified by oe6;
SQL> connect oe6/oe6
```

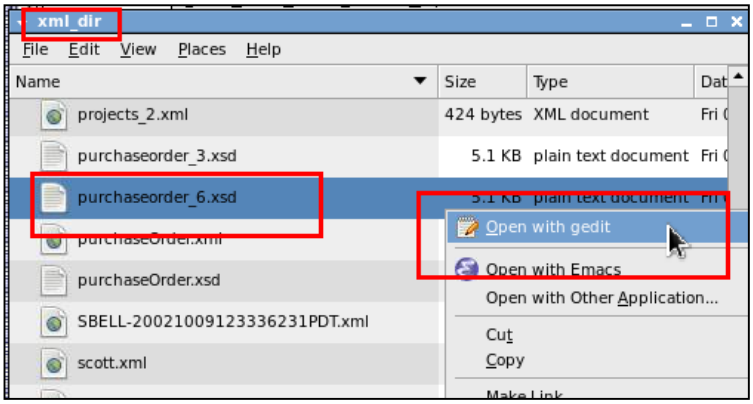
```
[oracle ~]$ sqlplus system/oracle as sysdba

SQL*Plus: Release 12.1.0.1.0 Production on Fri Aug 9 15:55:17 2013

Copyright (c) 1982, 2013, Oracle.  All rights reserved.


Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options

SQL> create user oe6 identified by oe6;

User created.

SQL> grant dba to oe6 identified by oe6;

Grant succeeded.

SQL> connect oe6/oe6
Connected.
SQL>
```

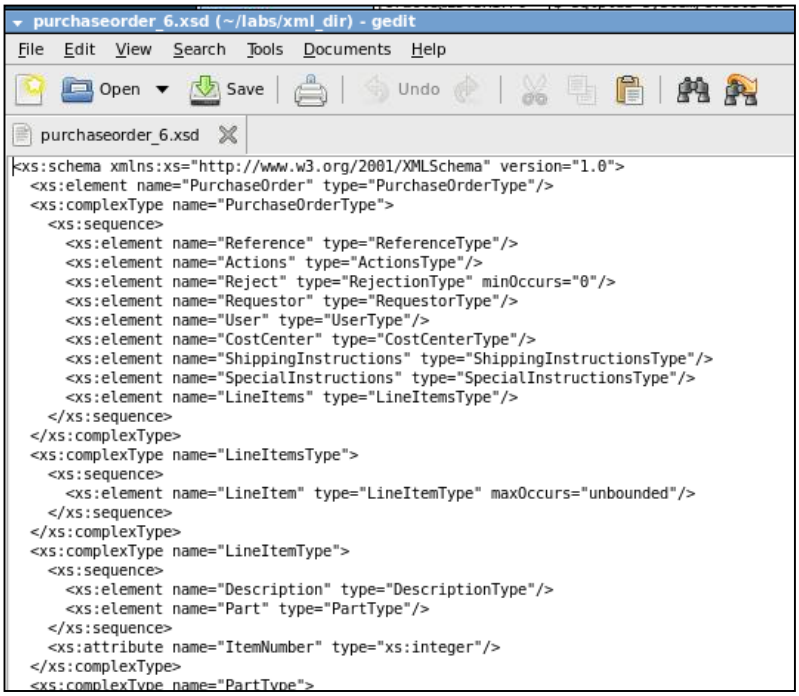
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. Access and review the un-annotated XML schema `purchaseorder_6.xsd` in the `xml_dir` folder. You can use either `gedit` or SQL Developer to view the contents of this file.




purchaseorder_6.xsd (~/labs/xml_dir) - gedit
File Edit View Search Tools Documents Help
Open
Save
Undo
purchaseorder_6.xsd
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.0">
<xs:element name="PurchaseOrder" type="PurchaseOrderType"/>
<xs:complexType name="PurchaseOrderType">
<xs:sequence>
<xs:element name="Reference" type="ReferenceType"/>
<xs:element name="Actions" type="ActionsType"/>
<xs:element name="Reject" type="RejectionType" minOccurs="0"/>
<xs:element name="Requestor" type="RequestorType"/>
<xs:element name="User" type="UserType"/>
<xs:element name="CostCenter" type="CostCenterType"/>
<xs:element name="ShippingInstructions" type="ShippingInstructionsType"/>
<xs:element name="SpecialInstructions" type="SpecialInstructionsType"/>
<xs:element name="LineItems" type="LineItemsType"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="LineItemsType">
<xs:sequence>
<xs:element name="LineItem" type="LineItemType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="LineItemType">
<xs:sequence>
<xs:element name="Description" type="DescriptionType"/>
<xs:element name="Part" type="PartType"/>
</xs:sequence>
<xs:attribute name="ItemNumber" type="xs:integer"/>
</xs:complexType>
<xs:complexType name="PartType">
</xs:schema>
</xs:schema>
</xs:schema>

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Create a new table named `annotation_tab`, which will hold the XML schema document. This table will have the following columns:

```
oracle@EDT3R17P0:~/labs/soln
File  Edit  View  Search  Terminal  Help
[oracle@EDT3R17P0 ~]$ pwd
/home/oracle
[oracle@EDT3R17P0 ~]$ cd labs/soln
[oracle@EDT3R17P0 soln]$ pwd
/home/oracle/labs/soln
[oracle@EDT3R17P0 soln]$ sqlplus oe6/oe6

SQL*Plus: Release 12.1.0.1.0 Production on Wed Sep 11 12:02:00 2013

Copyright (c) 1982, 2013, Oracle.  All rights reserved.

Last Successful login time: Wed Sep 11 2013 11:55:17 +00:00

Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing opt
ions

SQL>
```

Column Name	Column Type
ID	NUMBER
inp	XMLType
out	XMLType

```
-- Run as oe6/oe6
CREATE TABLE annotation_tab (id NUMBER, inp XMLType, out
XMLType);
```

```
Table created.
```

5. Insert the un-annotated `purchaseorder_6.xsd` XML schema from the `xml_dir` folder into the newly created `annotation_tab` `XMLType` table. Use the `XMLTYPE` constructor with `BFILE` in the insert statement. Use `nls_charset_id` and specify that the encoding for the file to be read is `UTF-8`.

```
-- Run as oe6/oe6

INSERT INTO annotation_tab
VALUES (1, xmltype(bfilename('XML_DIR', 'purchaseorder_6.xsd'),
nls_charset_id('AL32UTF8')), null);
```

```
1 row created.
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

6. Write a PL/SQL block named `practice_6_annotate.sql` to do the following:

a. Add a `defaultTable` annotation to the `PurchaseOrder` global element in the `purchaseorder_6.xsd` XML schema. Specify the name of the default table associated with this global element as `PURCHASEORDER_6_L6`.

b. Add a `SQLType` annotation to the global complex type `PurchaseOrderType`. Specify the name of the generated SQL object type as `PURCHASEORDER_T_L6`. Within the definition of this type:

1) Add a `SQLName` annotation to the `Reference` element. Specify the name of the SQL attribute corresponding to the XML element `Reference` as `REFERENCE_L6`.

2) Add a `SQLName` annotation to the `Actions` element. Specify the name of the SQL attribute corresponding to the XML element `Actions` as `ACTIONS_COLLECTION_L6`.

3) Add a `SQLName` annotation to the `USER` element. Specify the name of the SQL attribute corresponding to the XML element `USER` as `EMAIL_L6`.

4) Add a `SQLName` annotation to the `LineItems` element. Specify the name of the SQL attribute corresponding to the XML element `LineItems` as `LINEITEM_COLLECTION_L6`.

5) Add a `SQLType` annotation to the complex type `LineItemsType`. Specify the name of the generated SQL object type as `LINEITEMS_T_L6`. Within the definition of this type:

a) Add a `SQLName` annotation to the `LineItem` element. Specify the name of the generated SQL attribute corresponding to the XML element `LineItem` as `LINEITEM_VARRAY_L6`. Add a `SQLCollName` annotation that specifies that the SQL object type that manages the collection is named `LINEITEM_V_L6`.

6) Add a `SQLType` annotation to the complex type `LineItemType`. Specify the name of the generated SQL object type as `LINEITEM_T_L6`.

7) Add a `SQLType` annotation to the complex type `PartType`. Specify the name of the generated SQL object type as `PART_T_L6`. In addition, add an annotation `xdb:maintainDOM = "false"`, specifying that there is no need for Oracle XML DB to maintain DOM fidelity for elements based on this data type.

The `DISABLEMAINTAINDOM` procedure sets the DOM fidelity attribute to `FALSE`. There are two overloads. The first sets DOM fidelity attribute to `FALSE` for all complex types, and the second sets it to `FALSE` for the named complex type. This is equivalent to adding `xdb:maintainDOM="false"` on all or specified complex types respectively.

```
DECLARE schema XMLType;
BEGIN
SELECT t.inp INTO schema FROM annotation_tab t WHERE t.id =
1;
DBMS_XMLSCHEMA_ANNOTATE.setDefaultTable(schema,
'PurchaseOrder', 'PURCHASEORDER_6_L6');
DBMS_XMLSCHEMA_ANNOTATE.setSQLType(schema,
'PurchaseOrderType', 'PURCHASEORDER_T_L6');
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
DBMS_XMLSCHEMA_ANNOTATE.setSQLName(schema, 'complexType',
'PurchaseOrderType', 'element', 'Reference',
'REFERENCE_L6');
DBMS_XMLSCHEMA_ANNOTATE.setSQLName(schema, 'complexType',
'PurchaseOrderType', 'element', 'Actions',
'ACTIONS_COLLECTION_L6');
DBMS_XMLSCHEMA_ANNOTATE.setSQLName(schema, 'complexType',
'PurchaseOrderType', 'element', 'User', 'EMAIL_L6');
DBMS_XMLSCHEMA_ANNOTATE.setSQLName(schema, 'complexType',
'PurchaseOrderType', 'element', 'LineItems',
'LINEITEM_COLLECTION_L6');
DBMS_XMLSCHEMA_ANNOTATE.setSQLType(schema, 'LineItemsType',
'LINEITEMS_T_L6');
DBMS_XMLSCHEMA_ANNOTATE.setSQLCollType(schema,
'complexType', 'LineItemsType', 'LineItem',
'LINEITEM_C_L6');
DBMS_XMLSCHEMA_ANNOTATE.setSQLName(schema, 'complexType',
'LineItemsType', 'element', 'LineItem',
'LINEITEM_VARRAY_L6');
DBMS_XMLSCHEMA_ANNOTATE.setSQLType(schema, 'LineItemType',
'LINEITEM_T_L6');
DBMS_XMLSCHEMA_ANNOTATE.setSQLType(schema, 'PartType',
'PART_T');
DBMS_XMLSCHEMA_ANNOTATE.disableMaintainDom(schema,
'PartType');
UPDATE annotation_tab t SET t.out = schema WHERE t.id = 1;
END;
/
```

```
PL/SQL procedure successfully completed.

SQL>
```

7. Query the `inp` column of the `annotation_tab` and review the output. What does the `inp` column contain? Use the SQL*Plus settings `PAGESIZE` and `LONG` are used to ensure that the entire document is printed correctly, without line breaks.

```
-- Run as oe6/oe6

SET LONG 20000
SET PAGESIZE 100

SELECT inp
FROM annotation_tab;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

The `inp` column contains the un-annotated XML schema.

```
SQL> SET LONG 20000
SQL> SET PAGESIZE 100
SQL> SELECT inp
  2  FROM annotation_tab;

INP
--------------------------------------------------------------------------------
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.0">
  <xs:element name="PurchaseOrder" type="PurchaseOrderType"/>
  <xs:complexType name="PurchaseOrderType">
    <xs:sequence>
      <xs:element name="Reference" type="ReferenceType"/>
      <xs:element name="Actions" type="ActionsType"/>
      <xs:element name="Reject" type="RejectionType" minOccurs="0"/>
      <xs:element name="Requestor" type="RequestorType"/>
      <xs:element name="User" type="UserType"/>
      <xs:element name="CostCenter" type="CostCenterType"/>
      <xs:element name="ShippingInstructions" type="ShippingInstructionsType"/>
      <xs:element name="SpecialInstructions" type="SpecialInstructionsType"/>
      <xs:element name="LineItems" type="LineItemsType"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="LineItemsType">
    <xs:sequence>
      <xs:element name="LineItem" type="LineItemType" maxOccurs="unbounded"/>
    </xs:sequence>
```

8. Query the `out` column of the `annotation_tab` and review the output. What does the `out` column contain? The annotations are not yet performed until you register the XML schema.

```
 -- Run as oe6/oe6

SELECT out
FROM annotation_tab;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
OUT
--------------------------------------------------------------------------------
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.0" xmlns:xdb="
http://xmlns.oracle.com/xdb">
  <xs:element name="PurchaseOrder" type="PurchaseOrderType" xdb:defaultTable="PU
RCHASEORDER_6_L6"/>
  <xs:complexType name="PurchaseOrderType" xdb:SQLType="PURCHASEORDER_T_L6">
    <xs:sequence>
      <xs:element name="Reference" type="ReferenceType" xdb:SQLName="REFERENCE_L
6"/>
      <xs:element name="Actions" type="ActionsType" xdb:SQLName="ACTIONS_COLLECT
ION_L6"/>
      <xs:element name="Reject" type="RejectionType" minOccurs="0"/>
      <xs:element name="Requestor" type="RequestorType"/>
      <xs:element name="User" type="UserType" xdb:SQLName="EMAIL_L6"/>
      <xs:element name="CostCenter" type="CostCenterType"/>
      <xs:element name="ShippingInstructions" type="ShippingInstructionsType"/>
      <xs:element name="SpecialInstructions" type="SpecialInstructionsType"/>
      <xs:element name="LineItems" type="LineItemsType" xdb:SQLName="LINEITEM_CO
LLECTION_L6"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="LineItemsType" xdb:SQLType="LINEITEMS_T_L6">
    <xs:sequence>
      <xs:element name="LineItem" type="LineItemType" maxOccurs="unbounded" xdb:
SQLCollType="LINEITEM_C_L6" xdb:SQLName="LINEITEM_VARRAY_L6"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="LineItemType" xdb:SQLType="LINEITEM_T_L6">
    <xs:sequence>
      <xs:element name="Description" type="DescriptionType"/>
      <xs:element name="Part" type="PartType"/>
    </xs:sequence>
    <xs:attribute name="ItemNumber" type="xs:integer"/>
  </xs:complexType>
  <xs:complexType name="PartType" xdb:SQLType="PART_T" xdb:maintainDOM="false">
    <xs:attribute name="Id">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="10"/>
          <xs:maxLength value="14"/>
```

The `out` column contains an annotated version of the XML schema. The actual annotation objects are not yet generated. The objects are generated when you register the XML schema.

9. Write a PL/SQL block named `practice_6_register_schema.sql` to register the `purchaseorder_6.xsd` XML schema by using the following information. Declare an `XMLType` variable to hold to the XML schema.

Parameter	Setting
SCHEMAURL	`http://localhost:8080/source/schemas/poSource/xsd/purchaseorder_6.xsd`
SCHEMADOC	`schema`
Local	TRUE
GENTYPES	TRUE
GENTABLES	TRUE

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
-- Delete the XML schema if already registered. The delete_cascade_force deletes dependant tables and types.

BEGIN
  DBMS_XMLSCHEMA.deleteSchema(
    'http://localhost:8080/source/schemas/poSource/xsd/purchaseorder_6.xsd',
    DBMS_XMLSCHEMA.DELETE_CASCADE_FORCE );
END;
/

-- Registeres the XML schmema.

DECLARE
schema XMLType;
BEGIN
SELECT t.out INTO schema FROM annotation_tab t WHERE t.id = 1;
DBMS_XMLSCHEMA.registerSchema(
SCHEMAURL => 'http://localhost:8080/source/schemas/poSource/xsd/purchaseorder_6.xsd',
SCHEMADOC => schema,
LOCAL => TRUE,
GENTYPES => TRUE,
GENTABLES => TRUE);
END;
/
```

```
PL/SQL procedure successfully completed.

SQL>
```

10. Verify the creation of the tables, types, and other objects that were created after the schema registration.
    a. Describe the `USER_XML_TABLES` table. Select the table name, XML schema, and the element names from the `USER_XML_TABLES` table.

```
-- Run as oe6/oe6
DESCRIBE user_xml_tables
/
SELECT *
FROM USER_XML_TABLES;
```

```
Name                                      Null?    Type
----------------------------------------- -------- ----------------------------
TABLE_NAME                                         VARCHAR2(128)
XMLSCHEMA                                          VARCHAR2(700)
SCHEMA_OWNER                                       VARCHAR2(128)
ELEMENT_NAME                                       VARCHAR2(2000)
STORAGE_TYPE                                       VARCHAR2(17)
ANYSCHEMA                                          VARCHAR2(3)
NONSCHEMA                                          VARCHAR2(3)
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
TABLE_NAME
--------------------------------------------------------------------------------
XMLSCHEMA
--------------------------------------------------------------------------------
SCHEMA_OWNER
--------------------------------------------------------------------------------
ELEMENT_NAME
--------------------------------------------------------------------------------
STORAGE_TYPE      ANY NON

PURCHASEORDER_6_L6
http://localhost:8080/source/schemas/poSource/xsd/purchaseorder_6.xsd
OE6
PurchaseOrder
OBJECT-RELATIONAL

SQL> ▌
```

b. Verify the creation of the tables, types, and other objects that were created after the schema registration. Notice how the system-generated annotations have numbers as part of their type names such as `RejectionType782_T`, which is also case-sensitive. The following is a description of some of the generated table and types.

`DESC PURCHASEORDER_6_L6`

```
SQL> desc PURCHASEORDER_6_L6
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
TABLE of SYS.XMLTYPE(XMLSchema "http://localhost:8080/source/schemas/poSource/xsd/purchaseorder_6.xsd" Element "PurchaseOrder") STORAGE Object-rela
tional TYPE "PURCHASEORDER_T_L6"

SQL> ▌
```

`DESC PURCHASEORDER_T_L6`

```
SQL> DESC PURCHASEORDER_T_L6
 PURCHASEORDER_T_L6 is NOT FINAL
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 SYS_XDBPD$                                         XDB.XDB$RAW_LIST_T
 REFERENCE_L6                                       VARCHAR2(30 CHAR)
 ACTIONS_COLLECTION_L6                              ActionsType783_T
 Reject                                             RejectionType782_T
 Requestor                                          VARCHAR2(128 CHAR)
 EMAIL_L6                                           VARCHAR2(10 CHAR)
 CostCenter                                         VARCHAR2(4 CHAR)
 ShippingInstructions                               ShippingInstruction781_T
 SpecialInstructions                                VARCHAR2(2048 CHAR)
 LINEITEM_COLLECTION_L6                             LINEITEMS_T_L6
```

`DESC "RejectionType782_T"`

```
SQL> desc "RejectionType782_T"
 "RejectionType782_T" is NOT FINAL
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 SYS_XDBPD$                                         XDB.XDB$RAW_LIST_T
 User                                               VARCHAR2(10 CHAR)
 Date                                               DATE
 Comments                                           VARCHAR2(2048 CHAR)
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Note: The names of system-generated annotations such as `RejectionType782_T` will be different than the names of system annotations in your results.

```
DESC PART_T
```

```
SQL> DESC PART_T
 PART_T is NOT FINAL
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 Id                                                 VARCHAR2(14 CHAR)
 Quantity                                           NUMBER(14,2)
 UnitPrice                                          NUMBER(12,4)
```

11. Issue the following command to see the changes between the `inp` and `out` columns in the `annotation_tab` table. This shows the difference between the un-annotated and annotated XML schemas:

```
-- Run as oe6/oe6

SET LONG 20000
SET PAGESIZE 100

SELECT XMLDIFF(inp, out)
FROM annotation_tab;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
SQL> SELECT XMLDIFF(inp, out)
FROM annotation_tab;
  2
XMLDIFF(INP,OUT)
--------------------------------------------------------------------------------
<xd:xdiff xsi:schemaLocation="http://xmlns.oracle.com/xdb/xdiff.xsd http://xmlns
.oracle.com/xdb/xdiff.xsd" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:xdb
="http://xmlns.oracle.com/xdb" xmlns:xd="http://xmlns.oracle.com/xdb/xdiff.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <?oracle-xmldiff operations-in-docorder="true" output-model="snapshot" diff-al
gorithm="global"?>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:ele
ment[1]" xd:attr-local="defaultTable" xd:attr-nsuri="http://xmlns.oracle.com/xdb
">
    <xd:content>PURCHASEORDER_6_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[1]" xd:attr-local="SQLType" xd:attr-nsuri="http://xmlns.oracle.com/xdb"
>
    <xd:content>PURCHASEORDER_T_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[1]/xs:sequence[1]/xs:element[1]" xd:attr-local="SQLName" xd:attr-nsuri=
"http://xmlns.oracle.com/xdb">
    <xd:content>REFERENCE_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[1]/xs:sequence[1]/xs:element[2]" xd:attr-local="SQLName" xd:attr-nsuri=
"http://xmlns.oracle.com/xdb">
    <xd:content>ACTIONS_COLLECTION_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[1]/xs:sequence[1]/xs:element[5]" xd:attr-local="SQLName" xd:attr-nsuri=
"http://xmlns.oracle.com/xdb">
    <xd:content>EMAIL_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[1]/xs:sequence[1]/xs:element[9]" xd:attr-local="SQLName" xd:attr-nsuri=
"http://xmlns.oracle.com/xdb">
    <xd:content>LINEITEM_COLLECTION_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[2]" xd:attr-local="SQLType" xd:attr-nsuri="http://xmlns.oracle.com/xdb"
>
    <xd:content>LINEITEMS_T_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[2]/xs:sequence[1]/xs:element[1]" xd:attr-local="SQLName" xd:attr-nsuri=
"http://xmlns.oracle.com/xdb">
    <xd:content>LINEITEM_VARRAY_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[2]/xs:sequence[1]/xs:element[1]" xd:attr-local="SQLCollType" xd:attr-ns
uri="http://xmlns.oracle.com/xdb">
    <xd:content>LINEITEM_C_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[3]" xd:attr-local="SQLType" xd:attr-nsuri="http://xmlns.oracle.com/xdb"
>
    <xd:content>LINEITEM_T_L6</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[4]" xd:attr-local="SQLType" xd:attr-nsuri="http://xmlns.oracle.com/xdb"
>
    <xd:content>PART_T</xd:content>
  </xd:append-node>
  <xd:append-node xd:node-type="attribute" xd:parent-xpath="/xs:schema[1]/xs:com
plexType[4]" xd:attr-local="maintainDOM" xd:attr-nsuri="http://xmlns.oracle.com/
```

. . .

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

12. The following example, steps 12, 13, and 14, shows how SQL object types are created automatically when XML schema `test.xsd` is registered with Oracle XML DB. Note how elements without annotations are assigned system annotations.

    Note: Do not run this example. Steps 12, 13, and 14 are meant for informational purposes only.

    1) The `test.xsd` is a copy of the `purchaseorder_6.xsd` that you have used earlier, and is named here `purchaseorder_3`.
    2) The code used to register the schema is as follows:

```
begin
DBMS_XMLSCHEMA.registerSchema(
SCHEMAURL =>
'http://localhost:8080/source/schemas/poSource/xsd/
purchaseorder_3.xsd',
SCHEMADOC => bfilename('XML_DIR',' purchaseorder_3'),
LOCAL => TRUE,
GENTYPES => TRUE,
GENTABLES => TRUE);
end;
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
SQL> set long 20000
SQL> set pagesize 200
SQL> select schema from user_xml_schemas
  2  ;

SCHEMA
--------------------------------------------------------------------------------
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.0" xmlns:xdb="
http://xmlns.oracle.com/xdb" xdb:storeVarrayAsTable="true" xdb:flags="2105639" x
db:schemaURL="http://localhost:8080/source/schemas/poSource/xsd/purchaseorder_3.
xsd" xdb:schemaOwner="OE6" xdb:numProps="26">
  <xs:element name="PurchaseOrder" type="PurchaseOrderType" xdb:defaultTable="PU
RCHASEORDER" xdb:propNumber="4226" xdb:global="true" xdb:SQLName="PurchaseOrder"
 xdb:SQLType="PURCHASEORDER_T" xdb:SQLSchema="OE6" xdb:memType="258" xdb:default
TableSchema="OE6"/>
  <xs:complexType name="PurchaseOrderType" xdb:SQLType="PURCHASEORDER_T" xdb:SQL
Schema="OE6">
    <xs:sequence>
      <xs:element name="Reference" type="ReferenceType" xdb:SQLName="REFERENCE"
xdb:propNumber="4227" xdb:global="false" xdb:SQLType="VARCHAR2" xdb:memType="1"
xdb:MemInline="true" xdb:SQLInline="true" xdb:JavaInline="true"/>
      <xs:element name="Actions" type="ActionsType" xdb:SQLName="ACTIONS_COLLECT
ION" xdb:propNumber="4228" xdb:global="false" xdb:SQLType="ActionsType703_T" xdb
:SQLSchema="OE6" xdb:memType="258" xdb:MemInline="false" xdb:SQLInline="true" xd
b:JavaInline="false"/>
      <xs:element name="Reject" type="RejectionType" minOccurs="0" xdb:propNumbe
r="4229" xdb:global="false" xdb:SQLName="Reject" xdb:SQLType="RejectionType702_T
" xdb:SQLSchema="OE6" xdb:memType="258" xdb:MemInline="false" xdb:SQLInline="tru
e" xdb:JavaInline="false"/>
      <xs:element name="Requestor" type="RequestorType" xdb:propNumber="4230" xd
b:global="false" xdb:SQLName="Requestor" xdb:SQLType="VARCHAR2" xdb:memType="1"
xdb:MemInline="true" xdb:SQLInline="true" xdb:JavaInline="true"/>
      <xs:element name="User" type="UserType" xdb:SQLName="EMAIL" xdb:propNumber
="4231" xdb:global="false" xdb:SQLType="VARCHAR2" xdb:memType="1" xdb:MemInline=
"true" xdb:SQLInline="true" xdb:JavaInline="true"/>
      <xs:element name="CostCenter" type="CostCenterType" xdb:propNumber="4232"
xdb:global="false" xdb:SQLName="CostCenter" xdb:SQLType="VARCHAR2" xdb:memType="
1" xdb:MemInline="true" xdb:SQLInline="true" xdb:JavaInline="true"/>
```

```
SQL>
SQL> desc purchaseorder_t
 purchaseorder_t is NOT FINAL
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 SYS_XDBPD$                                         XDB.XDB$RAW_LIST_T
 REFERENCE                                          VARCHAR2(30 CHAR)
 ACTIONS_COLLECTION                                 ActionsType703_T
 Reject                                             RejectionType702_T
 Requestor                                          VARCHAR2(128 CHAR)
 EMAIL                                              VARCHAR2(10 CHAR)
 CostCenter                                         VARCHAR2(4 CHAR)
 ShippingInstructions                               ShippingInstruction701_T
 SpecialInstructions                                VARCHAR2(2048 CHAR)
 LINEITEM_COLLECTION                                LINEITEMS_T
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

13. Describe the `user_xml_schemas` data dictionary view and display the values of the schema and `schema_url` columns. When you register an XML schema, the schema information is captured in the `user_xml_schemas` data dictionary view.

```
DESCRIBE user_xml_schemas

SELECT schema, schema_url
FROM user_xml_schemas;
```

```
SQL> desc user_xml_schemas
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 SCHEMA_URL                                         VARCHAR2(700)
 LOCAL                                              VARCHAR2(3)
 SCHEMA                                             SYS.XMLTYPE
 INT_OBJNAME                                        VARCHAR2(4000)
 QUAL_SCHEMA_URL                                    VARCHAR2(865)
 HIER_TYPE                                          VARCHAR2(11)
 BINARY                                             VARCHAR2(3)
 SCHEMA_ID                                          RAW(16)
 HIDDEN                                             VARCHAR2(3)
```

```
SQL> SET LONG 20000
SQL> SET PAGESIZE 100
SQL> SELECT schema_url
  2  FROM user_xml_schemas;

SCHEMA_URL
--------------------------------------------------------------------------------
http://localhost:8080/source/schemas/poSource/xsd/purchaseorder_3.xsd
http://localhost:8080/source/schemas/poSource/xsd/test.xsd

SQL> ▌
```

In the above example, there are two XML schemas.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
SQL> SELECT schema
  2  FROM user_xml_schemas;

SCHEMA
--------------------------------------------------------------------------------
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" version="1.0" xmlns:xdb="
http://xmlns.oracle.com/xdb" xdb:storeVarrayAsTable="true" xdb:flags="2105639" x
db:schemaURL="http://localhost:8080/source/schemas/poSource/xsd/purchaseorder_3.
xsd" xdb:schemaOwner="OE6" xdb:numProps="26">
  <xs:element name="PurchaseOrder" type="PurchaseOrderType" xdb:defaultTable="PU
RCHASEORDER" xdb:propNumber="4226" xdb:global="true" xdb:SQLName="PurchaseOrder"
 xdb:SQLType="PURCHASEORDER_T" xdb:SQLSchema="OE6" xdb:memType="258" xdb:default
TableSchema="OE6"/>
  <xs:complexType name="PurchaseOrderType" xdb:SQLType="PURCHASEORDER_T" xdb:SQL
Schema="OE6">
    <xs:sequence>
      <xs:element name="Reference" type="ReferenceType" xdb:SQLName="REFERENCE"
xdb:propNumber="4227" xdb:global="false" xdb:SQLType="VARCHAR2" xdb:memType="1"
xdb:MemInline="true" xdb:SQLInline="true" xdb:JavaInline="true"/>
      <xs:element name="Actions" type="ActionsType" xdb:SQLName="ACTIONS_COLLECT
ION" xdb:propNumber="4228" xdb:global="false" xdb:SQLType="ActionsType703_T" xdb
:SQLSchema="OE6" xdb:memType="258" xdb:MemInline="false" xdb:SQLInline="true" xd
b:JavaInline="false"/>
      <xs:element name="Reject" type="RejectionType" minOccurs="0" xdb:propNumbe
r="4229" xdb:global="false" xdb:SQLName="Reject" xdb:SQLType="RejectionType702_T
" xdb:SQLSchema="OE6" xdb:memType="258" xdb:MemInline="false" xdb:SQLInline="tru
e" xdb:JavaInline="false"/>
      <xs:element name="Requestor" type="RequestorType" xdb:propNumber="4230" xd
b:global="false" xdb:SQLName="Requestor" xdb:SQLType="VARCHAR2" xdb:memType="1"
xdb:MemInline="true" xdb:SQLInline="true" xdb:JavaInline="true"/>
      <xs:element name="User" type="UserType" xdb:SQLName="EMAIL" xdb:propNumber
="4231" xdb:global="false" xdb:SQLType="VARCHAR2" xdb:memType="1" xdb:MemInline=
"true" xdb:SQLInline="true" xdb:JavaInline="true"/>
      <xs:element name="CostCenter" type="CostCenterType" xdb:propNumber="4232"
xdb:global="false" xdb:SQLName="CostCenter" xdb:SQLType="VARCHAR2" xdb:memType="
1" xdb:MemInline="true" xdb:SQLInline="true" xdb:JavaInline="true"/>
      <xs:element name="ShippingInstructions" type="ShippingInstructionsType" xd
b:propNumber="4233" xdb:global="false" xdb:SQLName="ShippingInstructions" xdb:SQ
LType="ShippingInstruction701_T" xdb:SQLSchema="OE6" xdb:memType="258" xdb:MemIn
line="false" xdb:SQLInline="true" xdb:JavaInline="false"/>
      <xs:element name="SpecialInstructions" type="SpecialInstructionsType" xdb:
```

. . .

Note the system-generated information in the above screenshot such as `MemInline` and `propNumber`. Also note the system-generated annotations such as `ActionsType703_T`, which are generated at the time you register your XML schema.

14. Commit your changes.

```
COMMIT;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 7: Partitioning XMLType Tables and Columns

Chapter 7

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practice 7-1: Annotating an XML Schema by Using the: Partitioning XML Data

The goal of this practice is to partition XML data.

1. In SQL Developer, open a SQL Worksheet by using the `OE` schema connection. Register the `ipo.xsd` schema by running `lab_07_01_01.sql`.

```
BEGIN
-- Delete the schema, if it already exists.
DBMS_XMLSCHEMA.deleteSchema('http://www.example.com/schemas/ipo.x
sd',4);
END;
/
BEGIN
-- Register the schema
DBMS_XMLSCHEMA.registerSchema(
'http://www.example.com/schemas/ipo.xsd',
'<schema targetNamespace="http://www.example.com/IPO"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:ipo="http://www.example.com/IPO">
<!-- annotation>
<documentation xml:lang="en">
International Purchase order schema for Example.com
Copyright 2000 Example.com. All rights reserved.
</documentation>
</annotation -->
<element name="purchaseOrder" type="ipo:PurchaseOrderType"/>
<element name="comment" type="string"/>
<complexType name="PurchaseOrderType">
<sequence>
<element name="shipTo" type="ipo:Address"/>
<element name="billTo" type="ipo:Address"/>
<element ref="ipo:comment" minOccurs="0"/>
<element name="ID" type="ipo:IDType"/>
<element name="items" type="ipo:Items"/>
</sequence>
</complexType>
<simpleType name="IDType">
    <restriction base="string">
      <minLength value="18"/>
      <maxLength value="30"/>
    </restriction>
</simpleType>
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
<complexType name="Items">
<sequence>
<element name="item" minOccurs="0" maxOccurs="unbounded">
<complexType>
<sequence>
<element name="productName" type="string"/>
<element name="quantity">
<simpleType>
<restriction base="positiveInteger">
<maxExclusive value="100"/>
</restriction>
</simpleType>
</element>
<element name="USPrice" type="decimal"/>
<element ref="ipo:comment" minOccurs="0"/>
<element name="shipDate" type="date" minOccurs="0"/>
</sequence>
<attribute name="partNum" type="ipo:SKU" use="required"/>
</complexType>
</element>
</sequence>
</complexType>
<complexType name="Address">
<sequence>
<element name="name" type="string"/>
<element name="street" type="string"/>
<element name="city" type="string"/>
<element name="state" type="string"/>
<element name="country" type="string"/>
<element name="zip" type="string"/>
</sequence>
</complexType>
<simpleType name="SKU">
<restriction base="string">
<pattern value="[0-9]{3}-[A-Z]{2}"/>
</restriction>
</simpleType>
</schema>',
TRUE,
TRUE,
FALSE);
END;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
anonymous block completed
```

2. Open `lab_07_01_02.sql`.

```
 CREATE TABLE part_purchaseorder OF XMLType
XMLSCHEMA "http://www.example.com/schemas/ipo.xsd"
ELEMENT "purchaseOrder"
VARRAY "XMLDATA"."items"."item" STORE AS TABLE my_item_table
((PRIMARY KEY (NESTED_TABLE_ID, SYS_NC_ARRAY_INDEX$)))
<TODO 1>
<TODO 2>
VARRAY "XMLDATA"."items"."item" STORE AS TABLE
my_item_p1(STORAGE (MINEXTENTS 13)),
<TODO 3>
VARRAY "XMLDATA"."items"."item" STORE AS TABLE
my_item_p2(STORAGE (MINEXTENTS 13)));.
```

   - Replace **<TODO 1>** to partition the base table `part_purchaseorder` by range of the `ID` element.
   - Replace **<TODO 2>** to create a partition `my_p1` for the ID values less than 100 and **<TODO 3>** to create a partition `my_p2` for the `ID` values less than `MAXVALUE`. (Execute in the `OE` schema). Save the changes and run `lab_07_01_02.sql`. Alternatively run `sol_07_01_02.sql`.

Answer:

```
CREATE TABLE part_purchaseorder OF XMLType
XMLSCHEMA "http://www.example.com/schemas/ipo.xsd"
ELEMENT "purchaseOrder"
VARRAY "XMLDATA"."items"."item" STORE AS TABLE my_item_table
((PRIMARY KEY (NESTED_TABLE_ID, SYS_NC_ARRAY_INDEX$)))
PARTITION BY RANGE (XMLDATA.ID)
(PARTITION my_p1 VALUES LESS THAN (100)
VARRAY "XMLDATA"."items"."item" STORE AS TABLE my_item_p1
(STORAGE (MINEXTENTS 13)),
PARTITION my_p2 VALUES LESS THAN (MAXVALUE)
VARRAY "XMLDATA"."items"."item" STORE AS TABLE my_item_p2
(STORAGE (MINEXTENTS 13)));
```

```
table PART_PURCHASEORDER created.
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. Verify that a nested table `my_item_table` has been created on the column `items.item`, of the parent table `part_purchaseorder`.

   Run `sol_07_01_03.sql`.

```
select table_name, parent_table_name, parent_table_column
  from USER_NESTED_TABLES;
```

Query Result

SQL | All Rows Fetched: 6 in 0.668 seconds

	TABLE_NAME	PARENT_TABLE_NAME	PARENT_TABLE_COLUMN
1	MY_ITEM_TABLE	PART_PURCHASEORDER	"XMLDATA"."items"."item"
2	SYS_NT4i/WR3scGgPgQ4IjuYuL0A==	purchaseOrder699_TAB	"XMLDATA"."items"."item"
3	SUBCATEGORY_REF_LIST_NESTEDTAB	CATEGORIES_TAB	TREAT(SYS_NC_ROWINFO$ AS "OE"."
4	PRODUCT_REF_LIST_NESTEDTAB	CATEGORIES_TAB	TREAT(SYS_NC_ROWINFO$ AS "OE"."
5	LINEITEM_TABLE	PURCHASEORDER	"XMLDATA"."LINEITEMS"."LINEITEM
6	ACTION_TABLE	PURCHASEORDER	"XMLDATA"."ACTIONS"."ACTION"

4. Verify that the child OCT `my_item_table` has been equi-partitioned with respect to the base table `part_purchaseorder`.

   Run `sol_07_01_04.sql`.

```
select table_name, partition_name
  from user_tab_partitions
```

Query Result

SQL | All Rows Fetched: 4 in 0.074 seconds

	TABLE_NAME	PARTITION_NAME
1	MY_ITEM_TABLE	MY_ITEM_P1
2	MY_ITEM_TABLE	MY_ITEM_P2
3	PART_PURCHASEORDER	MY_P1
4	PART_PURCHASEORDER	MY_P2

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 8: Using XQuery to Retrieve XML Data in Oracle XML DB

Chapter 8

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 8-1: Retrieving XML Data in Oracle XML DB

In this practice, you query the `XMLType` columns. You also use the SQL/XML functions to retrieve XML data from Oracle XML DB.

1. In SQL Developer, connect to the `HR_Schema`. By using `XMLTable`, write an XQuery statement that returns `EMPLOYEE_ID`, `FIRST_NAME`, and `LAST_NAME` from the `EMPLOYEES` table, of all employees who have `EMPLOYEE_ID` less than 110. Your query should display the output in the following style:

```
COLUMN_VALUE
-------------
<EMPLOYEE id="100"><EMPLOYEE_NAME>Steven_King</EMPLOYEE_NAME>
</EMPLOYEE><EMPLOYEE id="101">
<EMPLOYEE_NAME>Neena_Kochhar</EMPLOYEE_NAME></EMPLOYEE>
. . .
10 Rows Selected
```

Note: Click the **Run Script** icon or press **F5** in SQL Developer.

Run `sol_08_01_01.sql`.

```
-- the set long only works in sqlplus.
SET LONG 5000
Select * FROM XMLTABLE('
for $i in fn:collection("oradb:/HR/EMPLOYEES")/ROW
where $i/EMPLOYEE_ID/text() < 110
return
    <EMPLOYEE id="{$i/EMPLOYEE_ID}">
         <EMPLOYEE_NAME>{$i/FIRST_NAME/text()}
                                _{$i/LAST_NAME/text()}
         </EMPLOYEE_NAME>
    </EMPLOYEE>' );
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Output:

```
Script Output
Task completed in 0.252 seconds
line 3: SQLPLUS Command Skipped: SET LONG 5000
COLUMN_VALUE
--------------------------------------------------------------------------------
<EMPLOYEE id="100"><EMPLOYEE_NAME>Steven_King</EMPLOYEE_NAME></EMPLOYEE>
<EMPLOYEE id="101"><EMPLOYEE_NAME>Neena_Kochhar</EMPLOYEE_NAME></EMPLOYEE>
<EMPLOYEE id="102"><EMPLOYEE_NAME>Lex_De Haan</EMPLOYEE_NAME></EMPLOYEE>
<EMPLOYEE id="103"><EMPLOYEE_NAME>Alexander_Hunold</EMPLOYEE_NAME></EMPLOYEE>
<EMPLOYEE id="104"><EMPLOYEE_NAME>Bruce_Ernst</EMPLOYEE_NAME></EMPLOYEE>
<EMPLOYEE id="105"><EMPLOYEE_NAME>David_Austin</EMPLOYEE_NAME></EMPLOYEE>
<EMPLOYEE id="106"><EMPLOYEE_NAME>Valli_Pataballa</EMPLOYEE_NAME></EMPLOYEE>
<EMPLOYEE id="107"><EMPLOYEE_NAME>Diana_Lorentz</EMPLOYEE_NAME></EMPLOYEE>
<EMPLOYEE id="108"><EMPLOYEE_NAME>Nancy_Greenberg</EMPLOYEE_NAME></EMPLOYEE>
<EMPLOYEE id="109"><EMPLOYEE_NAME>Daniel_Faviet</EMPLOYEE_NAME></EMPLOYEE>

10 rows selected
```

2. In SQL Developer, connect to the `HR_Schema`. Modify the preceding query to use the `XMLQUERY` function. Observe the difference in the number of rows selected. You can use `lab_08_01_02.sql` for help.

   Note: Click the **Run Script** icon or press **F5** in SQL Developer.

   Run `sol_08_01_02.sql`.

```
Select XMLQuery('
for $i in fn:collection("oradb:/HR/EMPLOYEES")/ROW
where $i/EMPLOYEE_ID/text()< 110
return
     <EMPLOYEE id="{$i/EMPLOYEE_ID}">
          <EMPLOYEE_NAME>
               {$i/FIRST_NAME/text()}
     _{$i/LAST_NAME/text()}
          </EMPLOYEE_NAME>
     </EMPLOYEE>'
Returning content ) as Result FROM Dual;
```

Output:

```
Script Output
Task completed in 0.044 seconds
RESULT
--------------------------------------------------------------------------------
<EMPLOYEE id="100"><EMPLOYEE_NAME>Steven_King</EMPLOYEE_NAME></EMPLOYEE><EMPLOY
```

. . .

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. In SQL*Plus, connect as `oe` with the password `oe`. By using different path expressions with the `XMLQuery()` function, display the items listed in a) through e) from the `PRAC_PATH` table. You may use `lab_08_01_03.sql` for help. Replace ***<TODO>*** with suitable path expressions.

```
set long 5000
SELECT XMLQUery(
'for $i in fn:collection <TODO>
return $i'
RETURNING CONTENT) as PurchaseOrder
FROM DUAL;
```

a. The entire `PurchaseOrder` element.

Run `sol_08_01_03_a.sql`.

```
SELECT XMLQUery(
'for $i in fn:collection("oradb:/OE/PRAC_PATH")
return $i'
RETURNING CONTENT) as PurchaseOrder
FROM DUAL;
```

Output:

```
PURCHASEORDER
-------------------------------------------------------------------
<PurchaseOrder>
  <Actions>
    <Action>
      <Customer>SCOTT</Customer>
    </Action>
  </Actions>
  <Reject/>
  <Requestor>Julie P. Adams</Requestor>
  <User>ADAMS</User>
  <CostCenter>R20</CostCenter>
  <ShippingInstructions>
    <name>Julie P. Adams</name>
    <address>300 Oracle Parkway</address>
    <telephone>650 506 7300</telephone>
  </ShippingInstructions>
  <SpecialInstructions>Surface Mail</SpecialInstructions>
  <LineItems>
    <LineItem ItemNumber="1">
      <Description>Notorious</Description>
      <Part Id="715515012720" UnitPrice="39.95" Quantity="1"/>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
    </LineItem>
    <LineItem ItemNumber="2">
      <Description>Samurai Three: Duel at Ganryu
Island</Description>
      <Part Id="037429125625" UnitPrice="29.95" Quantity="1"/>
    </LineItem>
    <LineItem ItemNumber="3">
      <Description>A Night to Remember</Description>
      <Part Id="715515009058" UnitPrice="39.95" Quantity="3"/>
    </LineItem>
    <LineItem ItemNumber="4">
      <Description>Time Bandits</Description>
      <Part Id="715515010122" UnitPrice="39.95" Quantity="3"/>
    </LineItem>
  </LineItems>
</PurchaseOrder>
```

b. The `SHIPPINGINSTRUCTIONS` element.

Run `sol_08_01_03_b.sql`.

```
SELECT XMLQUery(
'for $i in
fn:collection("oradb:/OE/PRAC_PATH")//ShippingInstructions
return $i'
RETURNING CONTENT) as SHIPPINGINSTRUCTIONS
FROM DUAL;
```

Output:

```
SHIPPINGINSTRUCTIONS
--------------------------------------------------------------------
<ShippingInstructions>
  <name>Julie P. Adams</name>
  <address>300 Oracle Parkway</address>
  <telephone>650 506 7300</telephone>
</ShippingInstructions>
```

c. The second `LineItem` element

Run `sol_08_01_03_c.sql`.

```
SELECT XMLQUery(
'for $i in
fn:collection("oradb:/OE/PRAC_PATH")//LineItems/LineItem[2]
return $i'
RETURNING CONTENT) as PurchaseOrder
FROM DUAL;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Output:

```
PURCHASEORDER
--------------------------------------------------------------------------------
<LineItem ItemNumber="2"><Description>Samurai Three: Duel at Ganryu Island</Desc
ription><Part Id="037429125625" UnitPrice="29.95" Quantity="1"/></LineItem>
```

d. The `LineItem` element having `ItemNumber="1"` or `"4"`

Run `sol_08_01_03_d.sql`.

```
SELECT XMLQUery(
'for $i in
fn:collection("oradb:/OE/PRAC_PATH")//LineItems/LineItem[@ItemNu
mber="1" or @ItemNumber="4"]
return $i'
RETURNING CONTENT) as PurchaseOrder
FROM DUAL;
```

Output:

```
PURCHASEORDER
--------------------------------------------------------------------------------
<LineItem ItemNumber="1">
  <Description>Notorious</Description>
  <Part Id="715515012720" UnitPrice="39.95" Quantity="1"/>
</LineItem>
<LineItem ItemNumber="4">
  <Description>Time Bandits</Description>
  <Part Id="715515010122" UnitPrice="39.95" Quantity="3"/>
</LineItem>
```

e. The `LineItem` element having `Description ="A Night to Remember"`

Run `sol_08_01_03_e.sql`.

```
SELECT XMLQUery(
'for $i in
fn:collection("oradb:/OE/PRAC_PATH")//LineItems/LineItem[Descrip
tion ="A Night to Remember"]
return $i'
RETURNING CONTENT) as PurchaseOrder
FROM DUAL;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Output:

```
PURCHASEORDER
--------------------------------------------------------------------------
<LineItem ItemNumber="3">
  <Description>A Night to Remember</Description>
  <Part Id="715515009058" UnitPrice="39.95" Quantity="3"/>
</LineItem>
```

4. In SQL Developer, connect to the `HR_Schema`. Use `XMLQuery` to query the `EMPLOYEES` table and display the `FIRST_NAME` of all employees working in the department with `DEPARTMENT_ID` = `60` and earning a salary greater than `5,000`.

   You may use `lab_08_01_04.sql` for help. Replace <***TODO1***> with the name of the table and database schema, <***TODO2***> with the conditions, and <***TODO3***> with the `FIRST_NAME` of selected employees. After you make the changes, save the file, and then run `lab_08_01_04.sql`. **Alternatively, you can run `sol_08_01_04.sql`.**

   Note: Use **Run Script** or **F5** in SQL Developer.

Before editing `lab_08_01_04.sql`:

```
SET LONG 5000
SELECT XMLQuery(
    'for $i in fn:collection <TODO1>
     where <TODO2>
     return <TODO3> returning content) AS HR
  FROM dual;
```

After editing `lab_08_01_04.sql`:

```
SELECT XMLQuery(
    'for $i in fn:collection("oradb:/HR/EMPLOYEES")
     where $i/ROW/DEPARTMENT_ID = 60 and
           $i/ROW/SALARY > 5000
     return $i/ROW/FIRST_NAME' returning content) AS HR
  FROM dual;
```

Output:

```
Script Output
Task completed in 0.031 seconds
HR
--------------------------------------------------------------------------
<FIRST_NAME>Alexander</FIRST_NAME><FIRST_NAME>Bruce</FIRST_NAME>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

5. In SQL Developer, connect to the `HR_Schema`. Use `XMLQuery` to query the `EMPLOYEES` table and display the `FIRST_NAME` of all employees working in the department with `DEPTNO` = `20`, along with the `FIRST_NAME` of their managers.

   Hint: `Employee_Id` of the managers of an employee is stored in the `MGR` field. You can use a self join to obtain the desired result.

   You may use `lab_08_01_05.sql` for help. Replace <***TODO***> with a statement that matches `MANAGER_ID` of each employee in the outer `FLWOR` with `EMPLOYEE_ID` of the inner `FLWOR`. After you make the changes, save the file, and then run it. **Alternatively, you can run `sol_08_01_05.sql`.**

   Note: Use **Run Script** or **F5** in SQL Developer.

   Before editing `lab_08_01_05.sql`:

```
SELECT XMLQUERY('
   for $i in fn:collection("oradb:/HR/EMPLOYEES")/ROW
where $i/DEPARTMENT_ID = 20
return <EMPLOYEE NAME ="{ $i/FIRST_NAME  }">
     {
       for $j in fn:collection("oradb:/HR/EMPLOYEES")/ROW
       where <TODO>
       return <MANAGER>{$j/FIRST_NAME/text()} </MANAGER>
     }
  </EMPLOYEE>'
returning content) as XML FROM dual
```

   After editing `lab_08_01_05.sql`:

```
SELECT XMLQUERY('
   for $i in fn:collection("oradb:/HR/EMPLOYEES")/ROW
where $i/DEPARTMENT_ID = 20
return <EMPLOYEE NAME ="{ $i/FIRST_NAME  }">
     {
       for $j in fn:collection("oradb:/HR/EMPLOYEES")/ROW
       where $j/EMPLOYEE_ID eq $i/MANAGER_ID
       return <MANAGER>{$j/FIRST_NAME/text()} </MANAGER>
     }
  </EMPLOYEE>'
returning content) as XML FROM dual
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Output:

```
Script Output
Task completed in 0.058 seconds
XML
--------------------------------------------------------------------------------
<EMPLOYEE NAME="Michael"><MANAGER>Steven</MANAGER></EMPLOYEE><EMPLOYEE NAME="Pat"><MANAGER>Michael</MANAGER></EMPLOYEE>
```

6. In SQL Developer, connect to the `OE_Schema`. Use `XMLTable` to decompose the XML data that is stored in the `LineItem` element within the `PURCHASEORDER` table into an `XMLType` collection element. The `XMLTable` function should create a virtual table named `LineItem` with the following columns:
   a. `lineitem` containing the `NUMBER` data from the `ItemNumber` attribute
   b. `description` containing the `VARCHAR2` data from the `Description` element
   c. `partid` containing the `NUMBER` data from the `Id` attribute under the `Part` element
   d. `unitprice` containing the `NUMBER` data from the `UnitPrice` attribute under the `Part` element
   e. `quantity` containing the `NUMBER` data from the `Quantity` attribute under the `Part` element

   You may use `lab_08_01_06.sql` for help. Replace ***<TODO>*** with the data type followed by the path for the columns. After you make the changes, save the file, and then run it.

   Note: Use **Run Script** or **F5** in SQL Developer.

Before editing `lab_08_01_06.sql`:

```
SELECT lines.lineitem, lines.description, lines.partid,
               lines.unitprice, lines.quantity
FROM purchaseorder, XMLTable('
     for $i in /PurchaseOrder/LineItems/LineItem
     return $i'
     PASSING OBJECT_VALUE
     COLUMNS
          lineitem <TODO>,
          description <TODO>,
          partid <TODO>,
          unitprice <TODO>
          quantity <TODO>;
```

After editing `lab_08_01_06.sql`:

```
SELECT lines.lineitem, lines.description, lines.partid,
               lines.unitprice, lines.quantity
FROM purchaseorder, XMLTable('
     for $i in /PurchaseOrder/LineItems/LineItem
     return $i'
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
PASSING OBJECT_VALUE
COLUMNS
      lineitem NUMBER PATH '@ItemNumber',
      description VARCHAR2(30) PATH 'Description',
      partid NUMBER PATH 'Part/@Id',
      unitprice NUMBER PATH 'Part/@UnitPrice',
      quantity NUMBER PATH 'Part/@Quantity') lines;
```

Output:

Script Output

Task completed in 2.222 seconds

```
 11 The Horse's Mouth                 3.7E+10      29.95          1
 12 Carnival of Souls                 7.2E+11      39.95          3
 13 Playtime                          3.7E+10      29.95          4
 14 Andrei Rublev                     7.2E+11      39.95          1
 15 The Horse's Mouth                 3.7E+10      29.95          3
 16 The Element of Crime              3.7E+10      39.95          3
 17 Lord of the Flies                 3.7E+10      39.95          2
 18 How to Get Ahead in Advertisin    7.2E+11      29.95          4
 19 Samurai 1: Musashi Miyamoto       3.7E+10      29.95          4
 20 Closely Watched Trains            3.7E+10      29.95          4
 21 Variety Lights                    3.7E+10      29.95          1
  1 Testament of Orpheus              3.7E+10          0          1
  2 Rififi                            3.7E+10      29.95          2
  3 Le Trou                           3.7E+10      29.95          3
  4 M                                 3.7E+10      29.95          4
  5 The Unbearable Lightness Of Be    3.7E+10      29.95          3
  6 High and Low                      3.7E+10      39.95          1
  7 The Passion of Joan of Arc        3.7E+10      39.95          1
  8 The Cranes Are Flying             3.7E+10      29.95          2
  9 Vagabond                          3.7E+10      29.95          1
 10 The Scarlet Empress               7.2E+11      29.95          1
 11 The Scarlet Empress               7.2E+11      29.95          4
 12 Nights of Cabiria                 3.7E+10      39.95          2
 13 Under the Roofs of Paris          3.7E+10      29.95          3
 14 Wild Strawberries                 3.7E+10      39.95          3
 15 Notorious                         7.2E+11      39.95          4
 16 Black Orpheus                     3.7E+10      29.95          2
 17 Rushmore                          7.2E+11      39.95          2
 18 The 400 Blows                     3.7E+10      39.95          1
 19 Closely Watched Trains            3.7E+10      29.95          3
 20 Sanjuro                           3.7E+10      29.95          1

2,232 rows selected
```

7. In SQL Developer, connect to the `OE_Schema` connection. Query the `internal_orders` table to select the number of purchase orders for employee `100`.

   Note: Click the **Run Statement** icon or press **(Ctrl + Enter)** in SQL Developer.

   Run `sol_08_01_07.sql`.

```
SELECT count(*)
FROM internal_orders
WHERE employee_id = 100;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
Script Output
Task completed in 0.031 seconds
COUNT(*)
----------------------
2
```

8. In SQL*Plus, connect as `oe/oe`. Write a query on the `internal_orders` table to retrieve the purchase order for the employee "`King`".

   Note: Click the **Run Statement** icon or press **(Ctrl + Enter)** in SQL Developer.

   Run `sol_08_01_08.sql`.

```
SET LONG 20000
/
SELECT *
FROM internal_orders
WHERE order_document LIKE '%King%';
```

```
EMPLOYEE_ID
-----------
ORDER_DOCUMENT
--------------------------------------------------------------------------------
        100
<PurchaseOrder xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNames
paceSchemaLocation="http://www.oracle.com/xdb/po.xsd">
  <Reference>King-20011127121040988PST</Reference>
  <Actions>
    <Action>
      <User>SCOTT</User>
      <Date xsi:nil="true"/>
    </Action>
  </Actions>
  <Reject/>
  <Requestor>Steven King</Requestor>
  <User>King</User>
  <CostCenter>R20</CostCenter>
  <ShippingInstructions>
    <name>Steven King</name>
    <address>300 Oracle Parkway, Redwood Shores, CA 94065</address>
    <telephone>650 506 7300</telephone>
  </ShippingInstructions>
  <SpecialInstructions>Ground</SpecialInstructions>
  <LineItems>
    <LineItem ItemNumber="1">
      <Description>The Ruling Class</Description>
      <Part Id="715515012423" UnitPrice="39.95" Quantity="2"/>
    </LineItem>
    <LineItem ItemNumber="2">
      <Description>Diabolique</Description>
      <Part Id="037429135020" UnitPrice="29.95" Quantity="3"/>
    </LineItem>
    <LineItem ItemNumber="3">
      <Description>8 1/2</Description>
      <Part Id="037429135624" UnitPrice="39.95" Quantity="4"/>
    </LineItem>
  </LineItems>
</PurchaseOrder>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED


```
       100
<PurchaseOrder xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNameS
paceSchemaLocation="http://www.oracle.com/xdb/po.xsd">
  <Reference>King-200111271210412 8PST</Reference>
  <Actions>
    <Action>
      <User>HR</User>
      <Date/>
    </Action>
  </Actions>
  <Reject/>
  <Requestor>Steven King</Requestor>
  <User>King</User>
  <CostCenter>R20</CostCenter>
  <ShippingInstructions>
    <name>Steven King</name>
    <address>300 Oracle Parkway
Redwood Shores
CA
94065
USA</address>
    <telephone>650 506 7300</telephone>
  </ShippingInstructions>
  <SpecialInstructions>Hand Carry</SpecialInstructions>
  <LineItems>
    <LineItem ItemNumber="1">
      <Description>The Life of Brian</Description>
      <Part Id="715515010320" UnitPrice="39.95" Quantity="2"/>
    </LineItem>
    <LineItem ItemNumber="2">
      <Description>Hamlet</Description>
      <Part Id="037429128428" UnitPrice="29.95" Quantity="2"/>
    </LineItem>
    <LineItem ItemNumber="3">
      <Description>Salesman</Description>
      <Part Id="037429158920" UnitPrice="39.95" Quantity="4"/>
    </LineItem>
    <LineItem ItemNumber="4">
      <Description>Nights of Cabiria</Description>
      <Part Id="037429138427" UnitPrice="39.95" Quantity="1"/>
    </LineItem>
    <LineItem ItemNumber="5">
      <Description>Unbearable Lightness Of Being</Description>
      <Part Id="037429140222" UnitPrice="29.95" Quantity="2"/>
    </LineItem>
  </LineItems>
</PurchaseOrder>
```


9. In SQL Developer, connect to the `OE_Schema`. Use `XMLEXists` and write a query on the `PURCHASEORDER` table. Your query should check whether the XML document contains a root element `PurchaseOrder` that contains a `LineItems` element, the `LineItem` element of which contains a `Part` element with an `Id` attribute.

   Note: Click the **Run Statement** icon or press **(Ctrl + Enter)** in SQL Developer.

   Run `sol_08_01_09.sql`.

```
SELECT count(*)
FROM purchaseorder
WHERE XMLExists('$p/PurchaseOrder/LineItems/LineItem/Part/@Id'
PASSING OBJECT_VALUE AS "p");
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

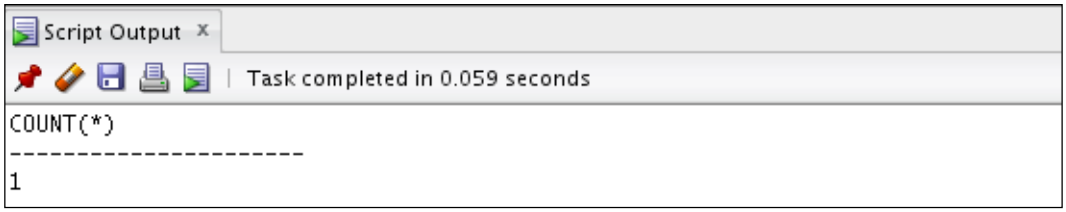
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

10. In SQL Developer, connect to the `OE_Schema`. Use `XMLEXists` and write a query on the `PURCHASEORDER` table. Your query should check whether the XML document contains a root element `PurchaseOrder` that contains a `LineItems` element, the third `LineItem` of which contains a `Part` element with the `Id` attribute value `71551500958`.

    Note: Click the **Run Statement** icon or press **(Ctrl + Enter)** in SQL Developer.

    Run `sol_08_01_10.sql`.

```
SELECT count(*)
FROM purchaseorder
WHERE XMLExists(
'$p/PurchaseOrder/LineItems/LineItem[3]/Part[@Id="715515009058"]
'
PASSING OBJECT_VALUE AS "p");
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 8-2: Using XQuery: Additional Use Cases

In this practice, you use XQuery to query XML data from various resources. You also use the SQL*Plus `XQUERY` command to query XML data.

1. In SQL Developer, open a SQL Worksheet by using the `OE_Schema` connection. By using `XQuery`, retrieve the data for the user `EABEL` from the `PURCHASEORDER` table, and then transform it to be displayed in the following style:

```
<Order_Total>
<Reference>EABEL-20021009123336251PDT</Reference>
<Total>Total of amount for this order</Total>
</Order_Total>
```

You may use `lab_08_02_01.sql` for help. Replace ***<TODO>*** by a call to the `fn:sum` method. Save the changes and run `lab_08_02_01.sql`. **Alternatively, you can run `sol_08_02_01.sql`.**

Before editing `lab_08_02_01.sql`:

```
SELECT ttab.COLUMN_VALUE AS Order_Total FROM purchaseorder,
XMLTable(
'for $i in /PurchaseOrder
  where $i/User = "EABEL"
  return  <Order_Total> {$i/Reference}
            <Total>
                {<TODO>}
            </Total>
            </Order_Total>'
  PASSING OBJECT_VALUE) ttab;
```

Note: Highlight the statement, and then click the **Run Statement** icon or press **(Ctrl + Enter)** in SQL Developer.

After editing `lab_08_02_01.sql`:

```
SELECT ttab.COLUMN_VALUE AS Order_Total FROM purchaseorder,
XMLTable(
'for $i in /PurchaseOrder
  where $i/User = "EABEL"
  return  <Order_Total> {$i/Reference}
            <Total>
                {fn:sum(for $j in $i/LineItems
                                    /LineItem/Part
                      return ($j/@Quantity*$j/@UnitPrice))
                }
            </Total>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
          </Order_Total>'
  PASSING OBJECT_VALUE) ttab;
```

Output:

Query Result

SQL | All Rows Fetched: 4 in 0.112 seconds

	ORDER_TOTAL
1	<Order_Total><Reference>EABEL-20021009123336251PDT</Reference><Total>289.6</Total></Order_Total>
2	<Order_Total><Reference>EABEL-20021009123338324PDT</Reference><Total>1328.05</Total></Order_Total>
3	<Order_Total><Reference>EABEL-20021009123335791PDT</Reference><Total>2067.15</Total></Order_Total>
4	<Order_Total><Reference>EABEL-20021009123336382PDT</Reference><Total>928.92</Total></Order_Total>

2. Modify the query in the preceding step to use `XMLQuery`. Execute the query. What do you observe?

   Note: Click the **Run Statement** icon or press **(Ctrl + Enter)** in SQL Developer.

   Run `sol_08_02_02.sql`.

```
SELECT XMLQuery(
'for $i in /PurchaseOrder
  where $i/User = "EABEL"
  return  <Order_Total> {$i/Reference}
            <Total>
                {fn:sum(for $j in $i/LineItems
                                    /LineItem/Part
                      return ($j/@Quantity*$j/@UnitPrice))
                }
            </Total>
          </Order_Total>'
  PASSING OBJECT_VALUE RETURNING CONTENT) FROM purchaseorder;
```

Output:

Query Result

SQL | All Rows Fetched: 132 in 0.15 seconds

	XMLQUERY('FOR$IIN/PURCHASEORDERWHERE$I/USER="EABEL"RETURN<ORDER_TOTAL>{$I/REFERENCE}<TOTAL>{FN:SUM(FO
13	(null)
14	(null)
15	(null)
16	<Order_Total><Reference>EABEL-20021009123336251PDT</Reference><Total>289.6</Total></Order_Total>
17	(null)
18	(null)

...

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
Query Result
SQL | All Rows Fetched: 132 in 0.15 seconds
    XMLQUERY('FOR$IIN/PURCHASEORDERWHERE$I/USER="EABEL"RETURN<ORDER_TOTAL>{$I/REFERENCE}<TOTAL>{FN:SUM(FO
37  (null)
38  (null)
39  <Order_Total><Reference>EABEL-20021009123338324PDT</Reference><Total>1328.05</Total></Order_Total>
40  (null)
41  (null)
```

...

```
Query Result
SQL | All Rows Fetched: 132 in 0.15 seconds
    XMLQUERY('FOR$IIN/PURCHASEORDERWHERE$I/USER="EABEL"RETURN<ORDER_TOTAL>{$I/REFERENCE}<TOTAL>{FN:SUM(FOR$
49  (null)
50  <Order_Total><Reference>EABEL-20021009123335791PDT</Reference><Total>2067.15</Total></Order_Total>
51  (null)
52  (null)
53  (null)
```

...

```
Query Result
SQL | All Rows Fetched: 132 in 0.15 seconds
    XMLQUERY('FOR$IIN/PURCHASEORDERWHERE$I/USER="EABEL"RETURN<ORDER_TOTAL>{$I/REFERENCE}<TOTAL>{FN:SUM(FOR
76  (null)
77  <Order_Total><Reference>EABEL-20021009123336382PDT</Reference><Total>928.92</Total></Order_Total>
78  (null)
79  (null)
80  (null)
81  (null)
```

...

The empty sequences returned by the `XQuery` expression are included in the result set of the query for a total of 132 rows. This means that the output contains not only the `Order_Total` elements for the orders requested by `Eabel` but also empty lines for all the other orders stored in the `PURCHASEORDER` table.

3. In SQL Developer, connect to the `HR_Schema` connection. Use the SQL*Plus `XQUERY` command to query the `DEPARTMENT_ID` column of the `EMPLOYEES` table and display the details of all employees working in department 30.

   Note: Use SQL*Plus or click the **Run Script** icon in SQL Developer.

   Run `sol_08_02_03.sql`.

   Answer:

```
XQUERY for $i in fn:collection("oradb:/HR/EMPLOYEES")
where $i/ROW/DEPARTMENT_ID = 30
return $i
 /
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Output:

```
Script Output
Task completed in 0.046 seconds
COLUMN_VALUE
--------------------------------------------------------------------------------
<ROW><EMPLOYEE_ID>114</EMPLOYEE_ID><FIRST_NAME>Den</FIRST_NAME><LAST_NAME>Raphaely</LAST_NAME><EMAIL>DRAPH
<ROW><EMPLOYEE_ID>115</EMPLOYEE_ID><FIRST_NAME>Alexander</FIRST_NAME><LAST_NAME>Khoo</LAST_NAME><EMAIL>AKH
<ROW><EMPLOYEE_ID>116</EMPLOYEE_ID><FIRST_NAME>Shelli</FIRST_NAME><LAST_NAME>Baida</LAST_NAME><EMAIL>SBAID
<ROW><EMPLOYEE_ID>117</EMPLOYEE_ID><FIRST_NAME>Sigal</FIRST_NAME><LAST_NAME>Tobias</LAST_NAME><EMAIL>STOBI
<ROW><EMPLOYEE_ID>118</EMPLOYEE_ID><FIRST_NAME>Guy</FIRST_NAME><LAST_NAME>Himuro</LAST_NAME><EMAIL>GHIMURO
<ROW><EMPLOYEE_ID>119</EMPLOYEE_ID><FIRST_NAME>Karen</FIRST_NAME><LAST_NAME>Colmenares</LAST_NAME><EMAIL>K

6 rows selected
```

4. In SQL Developer, connect to the `HR_Schema` connection. Modify the PL/SQL program unit code snippet in `lab_08_02_04.sql`. In <***TODO1***>, buffer a query that uses `XMLQuery` to query the `EMPLOYEES` table and retrieve the last names of employees who earn a salary greater than 2000, in the `sql_stmt` variable. You should then use dynamic SQL to execute the query and store the query result in the `XMLType` variable result in <***TODO2***>. Save the file, and then run it. **Alternatively, you can run `sol_08_02_04.sql`.**

Before editing `lab_08_02_04.sql`:

```
DECLARE
sql_stmt VARCHAR2(2000);
  result XMLType;
  doc DBMS_XMLDOM.DOMDocument;
  ndoc DBMS_XMLDOM.DOMNode;
  buf VARCHAR2(20000);
BEGIN
  sql_stmt :=
        <TODO1>;
EXECUTE IMMEDIATE <TODO2> ;
doc := DBMS_XMLDOM.newDOMDocument(result);
ndoc := DBMS_XMLDOM.makeNode(doc);
DBMS_XMLDOM.writeToBuffer(ndoc, buf);
DBMS_OUTPUT.put_line(buf);
END;
/
```

After editing `lab_08_02_04.sql`:

```
DECLARE
  sql_stmt VARCHAR2(2000);
  result XMLType;
  doc DBMS_XMLDOM.DOMDocument;
  ndoc DBMS_XMLDOM.DOMNode;
  buf VARCHAR2(20000);
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
BEGIN
  sql_stmt :=
        'SELECT XMLQuery(
         ''for $i in fn:collection("oradb:/HR/EMPLOYEES") ' ||
            'where $i/ROW/SALARY >2000' ||
                  'return $i/ROW/LAST_NAME'''||
        'RETURNING CONTENT) FROM DUAL';
  EXECUTE IMMEDIATE sql_stmt INTO result ;
  doc := DBMS_XMLDOM.newDOMDocument(result);
  ndoc := DBMS_XMLDOM.makeNode(doc);
  DBMS_XMLDOM.writeToBuffer(ndoc, buf);
  DBMS_OUTPUT.put_line(buf);
END;
/
```

Output:

```
Script Output
Task completed in 0.13 seconds
anonymous block completed
<LAST_NAME>OConnell</LAST_NAME>
<LAST_NAME>Grant</LAST_NAME>
<LAST_NAME>Whalen</LAST_NAME>
<LAST_NAME>Hartstein</LAST_NAME>
<LAST_NAME>Fay</LAST_NAME>
<LAST_NAME>Mavris</LAST_NAME>
<LAST_NAME>Baer</LAST_NAME>
<LAST_NAME>Higgins</LAST_NAME>
<LAST_NAME>Gietz</LAST_NAME>
<LAST_NAME>King</LAST_NAME>
<LAST_NAME>Kochhar</LAST_NAME>
<LAST_NAME>De Haan</LAST_NAME>
<LAST_NAME>Hunold</LAST_NAME>
<LAST_NAME>Ernst</LAST_NAME>
<LAST_NAME>Austin</LAST_NAME>
<LAST_NAME>Pataballa</LAST_NAME>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 9: Updating XML Content Using XQuery Update

Chapter 9

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 9-1: Updating XML Content by Using XQuery Update

Overview

In this practice you update XML content using XQuery update. You insert a `Date` element as the last child of an `Action` element.

Assumptions

Tasks

1. Use `oe/oe` to create an `XMLType` table named `purchaseorder_9` with binary XML storage, and populate the table with data selected from the `PURCHASEORDERS` table.

```
CREATE TABLE purchaseorder_9 OF XMLType;
INSERT INTO purchaseorder_9
SELECT OBJECT_VALUE FROM purchaseorder;
```

```
Table created.


132 rows created.
```

2. Describe the newly created `purchaseorder_9` table.

```
DESCRIBE purchaseorder_9
```

```
SQL> DESCRIBE purchaseorder_9
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
TABLE of SYS.XMLTYPE STORAGE BINARY
SQL>
```

3. Count the rows in the `purchaseorder_9` table to make sure it contains the same number of rows as in the `PURCHASEORDERS` table.

```
SELECT COUNT(*)
FROM purchaseorder_9;
```

```
SQL> SELECT COUNT(*)
  2  FROM purchaseorder_9;

  COUNT(*)
----------
       132

SQL>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Display the contents of the `purchaseorder_9` table.

```
SET LONG 20000
SET PAGESIZE 100

SELECT OBJECT_VALUE
FROM purchaseorder_9;
```

. . .

```
OBJECT_VALUE
--------------------------------------------------------------------------------
    <LineItem ItemNumber="16">
      <Description>Blood for Dracula</Description>
      <Part Id="715515009522" UnitPrice="39.95" Quantity="3"/>
    </LineItem>
  </LineItems>
</PurchaseOrder>

<PurchaseOrder xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNames
paceSchemaLocation="http://localhost:8080/source/schemas/poSource/xsd/purchaseOr
der.xsd">
  <Reference>SMCCAIN-20021009123335470PDT</Reference>
  <Actions>
    <Action>
      <User>SVOLLMAN</User>
    </Action>
  </Actions>
  <Reject/>
  <Requestor>Samuel B. McCain</Requestor>
  <User>SMCCAIN</User>
  <CostCenter>A10</CostCenter>
  <ShippingInstructions>
    <name>Samuel B. McCain</name>
    <address>800 Bridge Parkway
Redwood Shores
CA
```

. . .

```
      <Part Id="715515009126" UnitPrice="39.95" Quantity="3"/>
    </LineItem>
    <LineItem ItemNumber="15">
      <Description>The Horse&apos;s Mouth</Description>
      <Part Id="37429168721" UnitPrice="29.95" Quantity="4"/>
    </LineItem>
    <LineItem ItemNumber="16">
      <Description>Pygmalion</Description>
      <Part Id="37429151822" UnitPrice="29.95" Quantity="4"/>
    </LineItem>
  </LineItems>
</PurchaseOrder>


132 rows selected.
```

5. Run the following query to review the first child of the `<Action>` node. In this example, there is only one child of the `<Action>` node.

```
SELECT XMLQuery('$p/PurchaseOrder/Actions/Action[1]' PASSING
po.OBJECT_VALUE AS "p"
RETURNING CONTENT) action
FROM purchaseorder_9 po
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
WHERE XMLExists('$p/PurchaseOrder[Reference="TFOX-
2002100912333681PDT"]' PASSING po.OBJECT_VALUE AS "p");
```

You can also use the `last()` XPath function to display the last node element for the `<Action>` node as follows:

```
SELECT XMLQuery('$p/PurchaseOrder/Actions/Action[last()]' PASSING
po.OBJECT_VALUE AS "p"
RETURNING CONTENT) action
FROM purchaseorder_9 po
WHERE XMLExists('$p/PurchaseOrder[Reference="TFOX-
2002100912333681PDT"]'PASSING po.OBJECT_VALUE AS "p");
```

```
ACTION
--------------------------------------------------------
<Action>
  <User>GCAMBRAU</User>
</Action>
```

6. Next, insert the following `Date` element as the last child of an `Action` element where the `Reference="TFOX-2002100912333681PDT"`:

   `<Date>2002-12-31</Date>`

```
UPDATE purchaseorder_9
SET OBJECT_VALUE =
XMLQuery('copy $i := $p1 modify
 (for $j in $i/PurchaseOrder/Actions/Action[1]
  return insert nodes $p2 as last into $j)
  return $i' PASSING OBJECT_VALUE AS "p1",
  XMLType('<Date>2002-11-04</Date>') AS "p2"
  RETURNING CONTENT)
WHERE XMLExists('$p/PurchaseOrder[Reference="TFOX-
2002100912333681PDT"]' PASSING OBJECT_VALUE AS "p");
```

```
1 row updated.

SQL>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

7. Confirm the successful update operation by using the first `SELECT` statement from step 5.

```
SELECT XMLQuery('$p/PurchaseOrder/Actions/Action[1]' PASSING
po.OBJECT_VALUE AS "p"
RETURNING CONTENT) action
FROM purchaseorder_9 po
WHERE XMLExists('$p/PurchaseOrder[Reference="TFOX-
2002100912333681PDT"]' PASSING po.OBJECT_VALUE AS "p");
```

```
ACTION
--------------------------------------------------------------------------
<Action>
  <User>GCAMBRAU</User>
  <Date>2002-11-04</Date>
</Action>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 9-2: Inserting an Element After Another Element

Overview

In this practice, you insert a new element after another existing element.

Assumptions

Tasks

1. Using `oe/oe`, write a query to display all of the `LineItem` children of `LineItems` for `Reference="LSMITH-20021009123338234PDT"`. Use the `purchaseorder_9` that you created in the previous practice.

```
SELECT XMLQuery('$p/PurchaseOrder/LineItems/LineItem'
PASSING po.OBJECT_VALUE AS "p" RETURNING CONTENT)
FROM purchaseorder_9 po
WHERE XMLExists('$p/PurchaseOrder[Reference="LSMITH-
20021009123338234PDT"]' PASSING po.OBJECT_VALUE AS "p");
```

```
XMLQUERY('$P/PURCHASEORDER/LINEITEMS/LINEITEM'PASSINGPO.OBJECT_VALUEAS"P"RETURNI
--------------------------------------------------------------------------------
<LineItem ItemNumber="1">
  <Description>The Naked Kiss</Description>
  <Part Id="37429125823" UnitPrice="39.95" Quantity="1"/>
</LineItem>
<LineItem ItemNumber="2">
  <Description>Hearts and Minds</Description>
  <Part Id="37429166321" UnitPrice="39.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="3">
  <Description>Tokyo Drifter</Description>
  <Part Id="37429136126" UnitPrice="29.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="4">
  <Description>Traffic</Description>
  <Part Id="696306038924" UnitPrice="39.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="5">
  <Description>Salo</Description>
  <Part Id="715515009225" UnitPrice="29.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="6">
  <Description>For All Mankind</Description>
  <Part Id="37429139523" UnitPrice="39.95" Quantity="4"/>
</LineItem>
<LineItem ItemNumber="7">
  <Description>The Night Porter</Description>
  <Part Id="37429145227" UnitPrice="29.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="8">
  <Description>Haxan: Witchcraft Through the Ages</Description>
  <Part Id="37429161722" UnitPrice="39.95" Quantity="1"/>
</LineItem>
<LineItem ItemNumber="9">
  <Description>Carnival of Souls</Description>
  <Part Id="715515010221" UnitPrice="39.95" Quantity="3"/>
</LineItem>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
<LineItem ItemNumber="10">
  <Description>Cries and Whispers</Description>
  <Part Id="37429156322" UnitPrice="29.95" Quantity="3"/>
</LineItem>
<LineItem ItemNumber="11">
  <Description>Sanjuro</Description>
  <Part Id="37429141526" UnitPrice="29.95" Quantity="1"/>
</LineItem>
<LineItem ItemNumber="12">
  <Description>Andrei Rublev</Description>
  <Part Id="715515009928" UnitPrice="39.95" Quantity="4"/>
</LineItem>
<LineItem ItemNumber="13">
  <Description>The Magic Flute</Description>
  <Part Id="37429147528" UnitPrice="29.95" Quantity="4"/>
</LineItem>
<LineItem ItemNumber="14">
  <Description>Sid &amp; Nancy</Description>
  <Part Id="715515009232" UnitPrice="39.95" Quantity="4"/>
</LineItem>
```

2. Update the query in step 1 to display only the last element. You can use the `last()` XPath function as a predicate to display only the last `ItemNumber`.

```
SELECT XMLQuery('$p/PurchaseOrder/LineItems/LineItem[last()]'
PASSING po.OBJECT_VALUE AS "p" RETURNING CONTENT)
FROM purchaseorder_9 po
WHERE XMLExists('$p/PurchaseOrder[Reference="LSMITH-
20021009123338234PDT"]'
PASSING po.OBJECT_VALUE AS "p");
```

```
XMLQUERY('$P/PURCHASEORDER/LINEITEMS/LINEITEM[LAST()]'PASSINGPO.OBJECT_VALUEAS"P
--------------------------------------------------------------------------------
<LineItem ItemNumber="14">
  <Description>Sid &amp; Nancy</Description>
  <Part Id="715515009232" UnitPrice="39.95" Quantity="4"/>
</LineItem>
```

3. Using an update statement, add the following element *after* the last element:

```
<LineItem ItemNumber="15">
  <Description>The Last Samurai</Description>
  <Part Id="715515009232" UnitPrice="22.46" Quantity="4"/>
</LineItem>
```

```
UPDATE purchaseorder_9
SET OBJECT_VALUE =
XMLQuery('copy $i := $p1 modify
(for $j in $i/PurchaseOrder/LineItems/LineItem[last()]
return insert node $p2 after $j)
return $i'
PASSING OBJECT_VALUE AS "p1",
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
XMLType('<LineItem ItemNumber="15">
  <Description>The Last Samurai</Description>
  <Part Id="715515009232" UnitPrice="22.46" Quantity="4"/>
</LineItem>') AS "p2"
RETURNING CONTENT)
WHERE XMLExists('$p/PurchaseOrder[Reference="LSMITH-
20021009123338234PDT"]' PASSING OBJECT_VALUE AS "p");
```

```
SQL> @sol_09_02_03

1 row updated.
```

4. Confirm the addition of `LineItem 15`.

```
SELECT XMLQuery('$p/PurchaseOrder/LineItems/LineItem'
PASSING po.OBJECT_VALUE AS "p" RETURNING CONTENT)
FROM purchaseorder_9 po
WHERE XMLExists('$p/PurchaseOrder[Reference="LSMITH-
20021009123338234PDT"]'
PASSING po.OBJECT_VALUE AS "p");
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
XMLQUERY('$P/PURCHASEORDER/LINEITEMS/LINEITEM'PASSINGPO.OBJECT_VALUEAS"P"RETURNI
--------------------------------------------------------------------------------
<LineItem ItemNumber="1">
  <Description>The Naked Kiss</Description>
  <Part Id="37429125823" UnitPrice="39.95" Quantity="1"/>
</LineItem>
<LineItem ItemNumber="2">
  <Description>Hearts and Minds</Description>
  <Part Id="37429166321" UnitPrice="39.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="3">
  <Description>Tokyo Drifter</Description>
  <Part Id="37429136126" UnitPrice="29.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="4">
  <Description>Traffic</Description>
  <Part Id="696306038924" UnitPrice="39.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="5">
  <Description>Salo</Description>
  <Part Id="715515009225" UnitPrice="29.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="6">
  <Description>For All Mankind</Description>
  <Part Id="37429139523" UnitPrice="39.95" Quantity="4"/>
</LineItem>
<LineItem ItemNumber="7">
  <Description>The Night Porter</Description>
  <Part Id="37429145227" UnitPrice="29.95" Quantity="2"/>
</LineItem>
<LineItem ItemNumber="8">
  <Description>Haxan: Witchcraft Through the Ages</Description>
  <Part Id="37429161722" UnitPrice="39.95" Quantity="1"/>
</LineItem>
<LineItem ItemNumber="9">
  <Description>Carnival of Souls</Description>
  <Part Id="715515010221" UnitPrice="39.95" Quantity="3"/>
</LineItem>
<LineItem ItemNumber="10">
  <Description>Cries and Whispers</Description>
  <Part Id="37429156322" UnitPrice="29.95" Quantity="3"/>
</LineItem>
```

```
  <Part Id="37429141526" UnitPrice="29.95" Quantity="1"/>
</LineItem>
<LineItem ItemNumber="12">
  <Description>Andrei Rublev</Description>
  <Part Id="715515009928" UnitPrice="39.95" Quantity="4"/>
</LineItem>
<LineItem ItemNumber="13">
  <Description>The Magic Flute</Description>
  <Part Id="37429147528" UnitPrice="29.95" Quantity="4"/>
</LineItem>
<LineItem ItemNumber="14">
  <Description>Sid &amp; Nancy</Description>
  <Part Id="715515009232" UnitPrice="39.95" Quantity="4"/>
</LineItem>
<LineItem ItemNumber="15">
  <Description>The Last Samurai</Description>
  <Part Id="715515009232" UnitPrice="22.46" Quantity="4"/>
</LineItem>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 10: Searching XML Content Using XQuery Full-Text

Chapter 10

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 10

There are no practices for this lesson.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 11: Indexing XMLType Data

Chapter 11

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

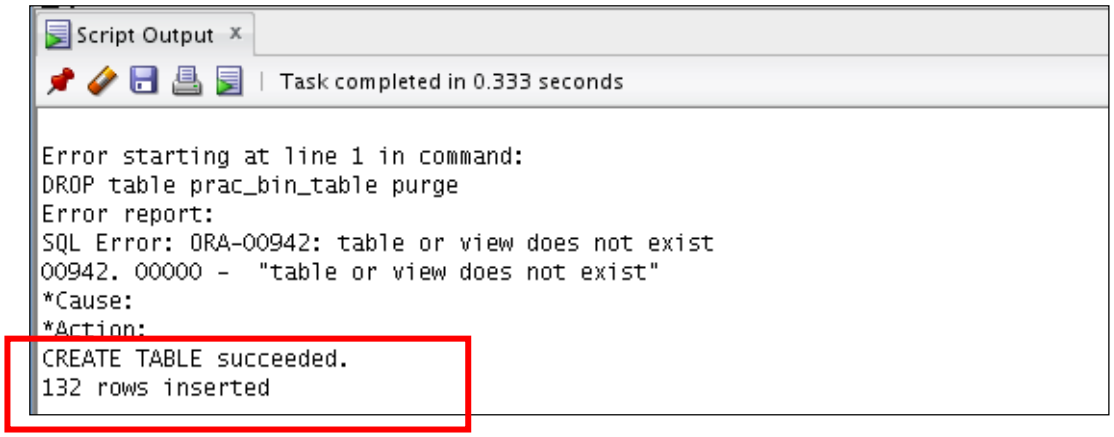
Oracle University and Error : You are not a Valid Partner use only

Practice 11-1: Indexing `XMLType` Data

In this practice, you create an `XMLIndex` index on a binary XML table and review the execution plan of a query. In the last step, you query the data dictionary views to obtain the name of the `XMLIndex` index and the system-generated name of the path table.

1. In SQL Developer, connect to the `OE_Schema` connection if you are not already connected. Run `lab_11_01.sql` to create an `XMLType` table with binary XML storage. This script also populates the table with data selected from the `PURCHASEORDER` table. Name the table `prac_bin_table`.

```
DROP table prac_bin_table purge;
/
CREATE TABLE prac_bin_table OF XMLType
XMLType STORE AS SECUREFILE BINARY XML
/
INSERT INTO prac_bin_table
SELECT object_value from purchaseorder
/
```

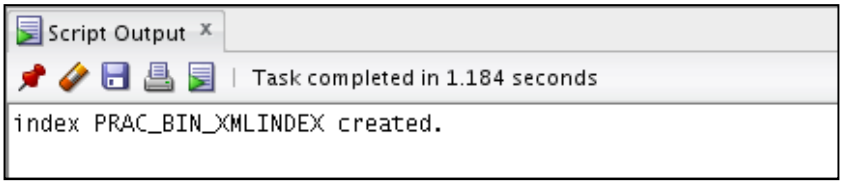


a. Open `lab_11_01_a.sql`. Replace <***TODO***> with the appropriate code to create the `prac_bin_xmlindex` `XMLIndex` on the `prac_bin_table` table. Run the script.

```
create index prac_bin_xmlindex on prac_bin_table(object_value)
indextype is xdb.xmlindex
parameters ('PATH TABLE prac_bin_xmlindex_path_table
PATH ID INDEX prac_bin_path_id_ix
ORDER KEY INDEX prac_bin_order_key_ix
VALUE INDEX prac_bin_value_ix')
/
```

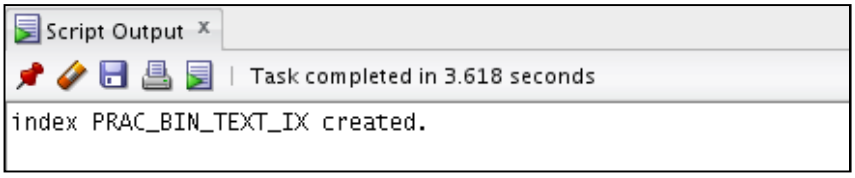
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

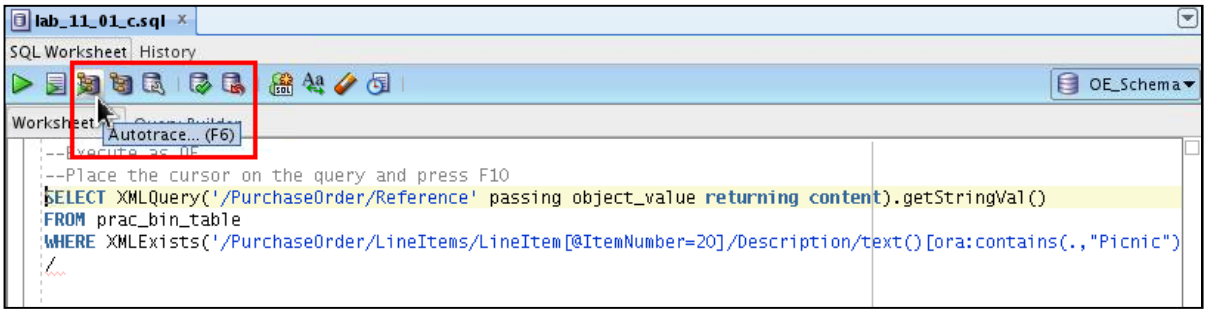


b. Create a secondary text index on the `VALUE` column of the path table. The secondary text index can be called `prac_bin_text_ix`. Use `lab_11_01_b.sql` for help. Run the script.

```
CREATE INDEX prac_bin_text_ix ON prac_bin_xmlindex_path_table
(VALUE) INDEXTYPE IS CTXSYS.CONTEXT
parameters ('transactional')
/
```



c. In SQL Developer, open `lab_11_01_c.sql`, place the cursor anywhere on the `SELECT` statement, and then click the **Autotrace** icon.



On the Autotrace output tab, you see the Explain Plan of the query. Observe the usage of the `XMLIndex` index.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Autotrace

SQL | 1.856 seconds

OPERATION	OBJECT_NAME	COST	LAST_C
SELECT STATEMENT		13	
SORT (GROUP BY)			18
TABLE ACCESS (BY INDEX ROWID BATCH	PRAC_BIN_XMLINDEX_PATH_TABLE	2	6
Filter Predicates			
SYS_XMLI_LOC_ISNODE(SYS_P1:			
INDEX (RANGE SCAN)	PRAC_BIN_PATH_ID_IX	1	4
Access Predicates			
AND			
SYS_P11.PATHID=HEXTC			
SYS_P11.RID=:B1			
NESTED LOOPS		11	487
VIEW	VW_SQ_1	10	485
HASH (UNIQUE)			485
NESTED LOOPS		10	485
NESTED LOOPS		8	456
TABLE ACCESS (BY INDEX RC	PRAC_BIN_XMLINDEX_PATH_TABLE	2	29
Filter Predicates			
AND			
SYS_P4.PATHID=			
TO_BINARY_DOL			
SYS_XMLI_LOC_I:			
INDEX (RANGE SCAN)	PRAC_BIN_ORDER_KEY_IX	1	16
Access Predicates			
AND			
SYS_P4.RID=:			
SYS_P2.ORDE			
SYS_P4.ORDE			
Filter Predicates			
SYS_ORDERKEY_I			
TABLE ACCESS (BY USER ROWID)	PRAC_BIN_TABLE	1	2

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

2. Describe the `USER_INDEXES` data dictionary view.

```
DESCRIBE user_indexes
```

Script Output

Task completed in 0.549 seconds

```
desc user_indexes
Name                    Null     Type
----------------------- -------- --------------
INDEX_NAME              NOT NULL VARCHAR2(128)
INDEX_TYPE                       VARCHAR2(27)
TABLE_OWNER             NOT NULL VARCHAR2(128)
TABLE_NAME              NOT NULL VARCHAR2(128)
TABLE_TYPE                       VARCHAR2(11)
UNIQUENESS                       VARCHAR2(9)
COMPRESSION                      VARCHAR2(13)
PREFIX_LENGTH                    NUMBER
TABLESPACE_NAME                  VARCHAR2(30)
INI_TRANS                        NUMBER
MAX_TRANS                        NUMBER
INITIAL_EXTENT                   NUMBER
NEXT_EXTENT                      NUMBER
MIN_EXTENTS                      NUMBER
MAX_EXTENTS                      NUMBER
PCT_INCREASE                     NUMBER
PCT_THRESHOLD                    NUMBER
INCLUDE_COLUMN                   NUMBER
FREELISTS                        NUMBER
FREELIST_GROUPS                  NUMBER
PCT_FREE                         NUMBER
LOGGING                          VARCHAR2(3)
BLEVEL                           NUMBER
LEAF_BLOCKS                      NUMBER
DISTINCT_KEYS                    NUMBER
AVG_LEAF_BLOCKS_PER_KEY          NUMBER
AVG_DATA_BLOCKS_PER_KEY          NUMBER
CLUSTERING_FACTOR                NUMBER
STATUS                           VARCHAR2(8)
NUM_ROWS                         NUMBER
SAMPLE_SIZE                      NUMBER
LAST_ANALYZED                    DATE
```

3. Query the `USER_INDEXES` data dictionary view to obtain the name of the `XMLIndex` index on the `prac_bin_table` table.

   Run `sol_11_01_03.sql`.

```
SELECT INDEX_NAME FROM USER_INDEXES
WHERE TABLE_NAME = 'PRAC_BIN_TABLE' AND ITYP_NAME = 'XMLINDEX';
```

Oracle University and Error : You are not a Valid Partner use only

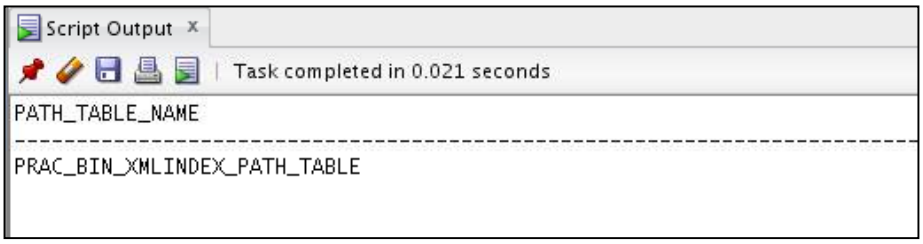
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Write a query to determine the name of the path table. Use the `USER_XML_INDEXES` data dictionary view.

   Run sol_11_01_04.sql.

```
SELECT PATH_TABLE_NAME FROM USER_XML_INDEXES
WHERE TABLE_NAME = 'PRAC_BIN_TABLE';
```



Note: The output may not be the same when you query to determine the system-generated name of a path table.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 12: Generating XML Data

Chapter 12

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

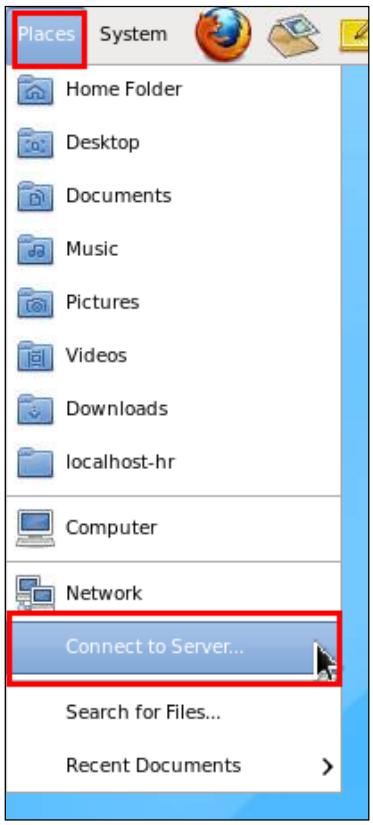
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 12-1: Generating XML

The goal of this practice is to gain hands-on experience of the functions learned in this lesson.

Before performing this practice, you need to map a Web Folder to Oracle XML DB Repository. Perform the following:

1. On your Linux desktop, select **Places > Connect to Server**.



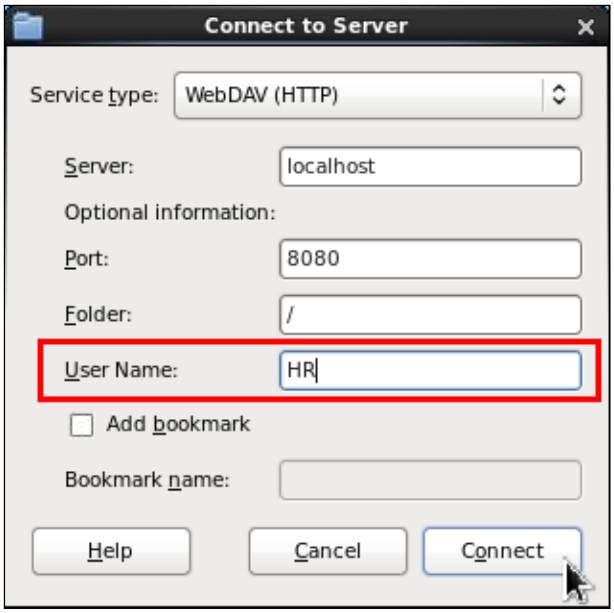
2. In the **Connect to Server** window, enter the following details, and then click **Connect**.
   a. Service type: Webdav (HTTP)
   b. Server: localhost
   c. Port: 8080
   d. Folder: /
   e. User Name: **`HR`**

   Note: Make sure that the username is entered in uppercase, `HR`, because that is how the username is stored in the database.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



3. You are prompted for a password for the XDB realm but for the `HR` username, which is `hr` in lowercase. Enter the password as `hr` (lowercase), select the "Remember password until you logout" option, and then click **Connect**.

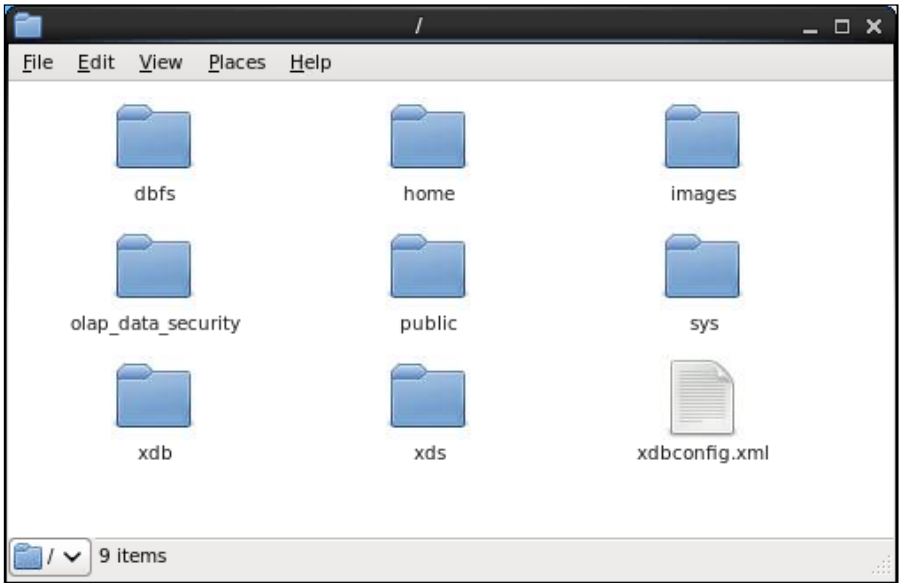
Note: You learned how to map a Web Folder in the *Oracle 11g: XML Fundamentals* prerequisite course. This topic is reviewed again in this course in the lesson titled "Oracle XML DB Repository."

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

The home web folder is displayed.



4. In SQL Developer, open a SQL Worksheet for the `hr` schema. Open `lab_12_01_04.sql`. The program creates the `departments.xml` resource in `/public/employees` in Oracle XML DB Repository. The program stores the results of the query in `departments.xml`.
   a. Replace <***TODO***> with the appropriate code to query `DEPARTMENT_ID`, `DEPARTMENT_NAME`, and `LOCATION_ID` from the `DEPARTMENTS` table for all departments with `DEPARTMENT_ID` < `150`. Save the changes and run `lab_12_01_04.sql` using the `HR_Schema` database connection. **Alternatively, you can run `sol_12_01_04.sql`.**

      Note: In Oracle Database 12*c*, you use the `DBMS_XDB_REPOS` package instead of the `DBMS_XDB` package. The subprograms and constants of PL/SQL package `DBMS_XDB` have been divided among the following packages: `DBMS_XDB_ADMIN`, `DBMS_XDB_CONFIG`, and `DBMS_XDB_REPOS`.

Answer:

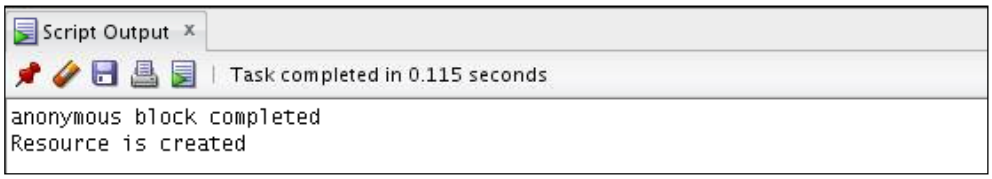
```
-- Use hr/hr
SET SERVEROUTPUT ON
DECLARE
  XMLdoc XMLType;
res boolean;
BEGIN
if not (DBMS_XDB_REPOS.EXISTSRESOURCE('/public/employees')) then
    res:= DBMS_XDB_REPOS.CREATEFOLDER('/public/employees');
end if;
  SELECT XMLQuery(
'for $j in 1
  return (
  <DEPARTMENTS> {
   for $i in fn:collection("oradb:/HR/DEPARTMENTS")/ROW
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
   where $i/DEPARTMENT_ID < 150
   return (<DEPARTMENT>
            <DEPTNO>{xs:string($i/DEPARTMENT_ID)}</DEPTNO>
            <DNAME>{xs:string($i/DEPARTMENT_NAME)}</DNAME>
            <LOC>{xs:integer($i/LOCATION_ID)}</LOC>
           </DEPARTMENT>)} </DEPARTMENTS>)'
  RETURNING CONTENT) INTO XMLdoc FROM DUAL;
-- Delete the department.xml resource if it already exits.
    if
(dbms_xdb_repos.existsResource('/public/employees/departments.xm
l')) then

dbms_xdb_repos.deleteResource('/public/employees/departments.xml
');
    end if;
IF(DBMS_XDB_REPOS.CREATERESOURCE('/public/employees/departments.
xml', XMLdoc)) THEN
      DBMS_OUTPUT.PUT_LINE('Resource is created');
  ELSE
      DBMS_OUTPUT.PUT_LINE('Cannot create resource');
  END IF;
  COMMIT;
END;
/
```



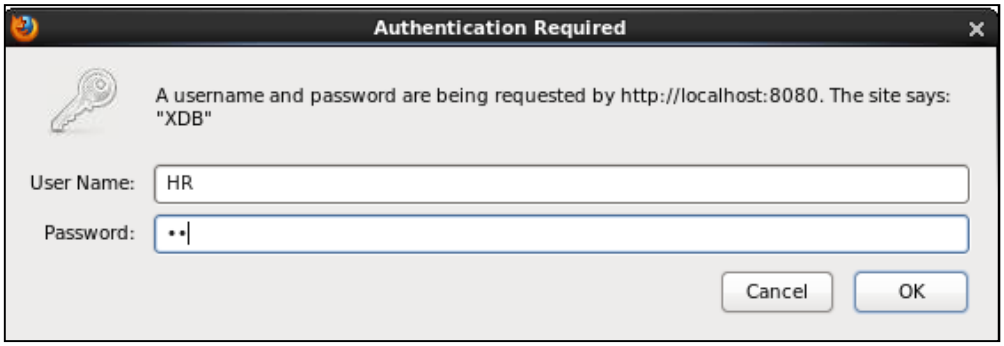
b. By using gedit or Firefox and the Web Folder that you created, open and verify the contents of `departments.xml`. **Note:** If you are prompted for a User Name and Password, enter **`HR`** for the User Name and `hr` for the password. Browse to `/public/employees` and open `departments.xml` by using the following URL, and then double-click the `departments.xml` link if you are using Firefox.

http://localhost:8080/public/employees

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

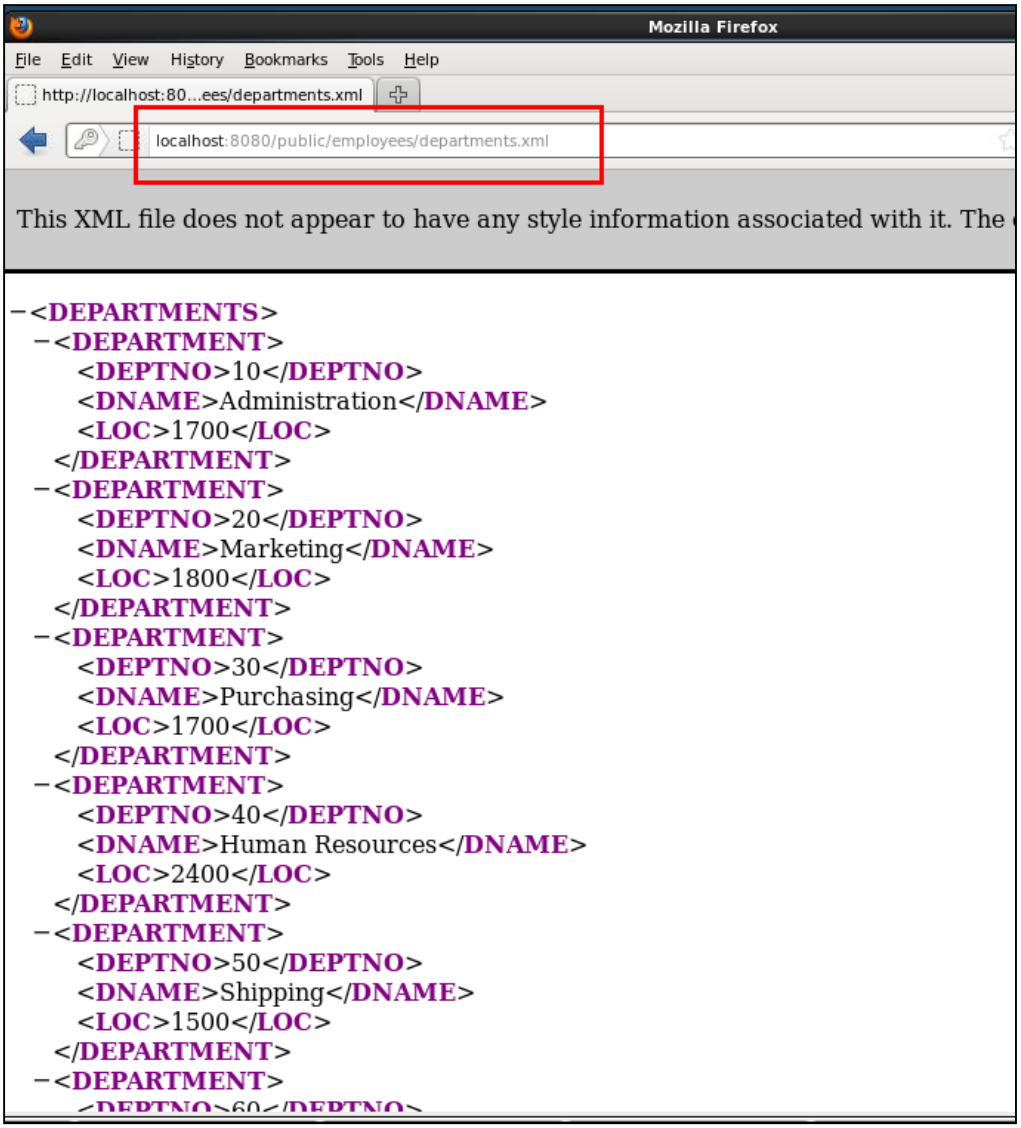


The following content should be displayed:

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



5. By using the `XMLELEMENT()` and `XMLATTRIBUTES()` SQL/XML functions, generate the following XML data from the `EMPLOYEES` table (`hr` schema) for all employees belonging to department `20`. Save your SQL statement in a text file named `lab_12_01_05.sql`. Run `lab_12_01_05.sql`. Alternatively, you can run `sol_12_01_05.sql`.

```
<Employee ID="">
<Name></Name>
<Department></Department>
<Pay></Pay>
</Employee>
```

Answer:

```
select
XMLELEMENT("Employee", XMLATTRIBUTES(employee_id id),
XMLELEMENT("Name", first_name),
```

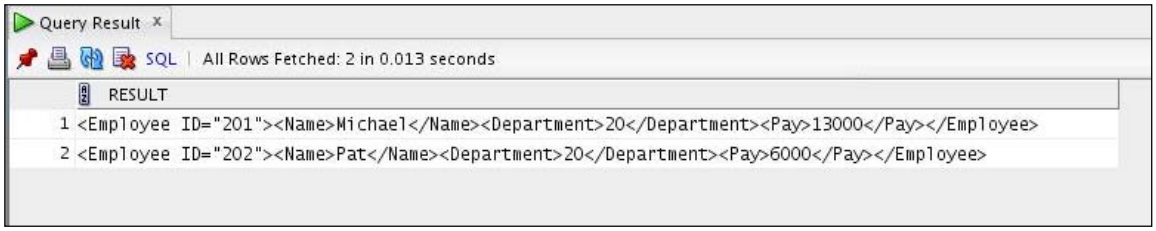
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
XMLELEMENT("Department", department_id),
XMLELEMENT("Pay", salary))
as Result
from employees
where department_id=20;
```

Output:

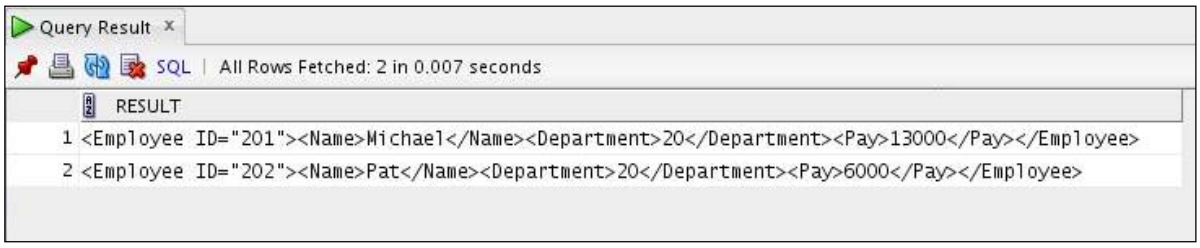


6. Modify your query in **`lab_12_01_05.sql`** and retrieve the same output by using the `XMLFOREST()` function to generate child elements. Save your SQL statement in a text file named **`lab_12_01_06.sql`**. Run **`lab_12_01_06.sql`**. Alternatively, you can run **`sol_12_01_06.sql`**.

Answer:

```
select
XMLELEMENT("Employee",XMLATTRIBUTES(employee_id id),
XMLFOREST(first_name "Name", department_id "Department", salary
"Pay"))
as Result
from employees
where department_id=20;
```

Output:



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

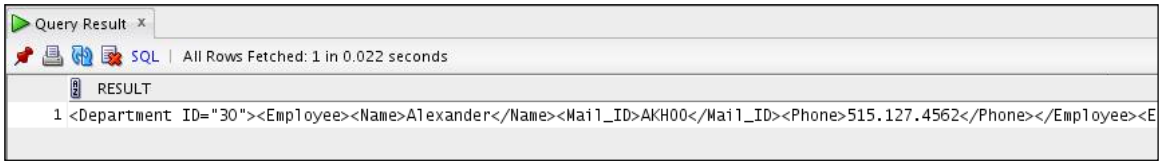
7. Generate the name, email ID, and phone number for each employee working in department 30. Name the elements and attributes as follows. Save your SQL statement in a text file named `lab_12_01_07.sql`, and then run the script. Alternatively, you can run `sol_12_01_07.sql`.

```
<Department ID="30">
<Employee>
  <Name></Name>
  <Mail_ID></Mail_ID>
  <Phone></Phone>
</Employee>
  …
</Department>
```

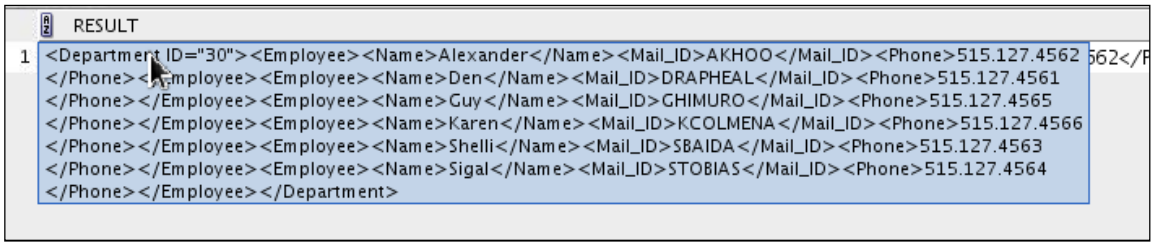
Hint: Use the `XMLAGG()` function.

Answer:

```
select
XMLELEMENT("Department",XMLATTRIBUTES('30' "ID"),
XMLAGG(XMLELEMENT("Employee",
XMLFOREST(first_name "Name",email "Mail_ID",phone_number "Phone"))
order by first_name))
as result
from employees
where department_id=30;
```



Tip: If you move your mouse cursor over the long output row, you can see the entire output in a pop-up window as follows:



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

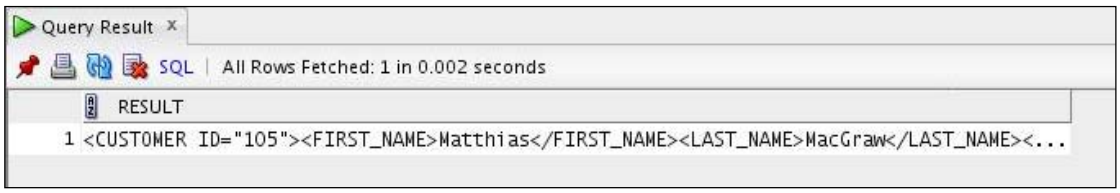
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

8. From the **oe** schema, generate the first and last name, and the order details for the customer with `id=105` as shown. Save your SQL statement in a text file named `lab_12_01_08.sql`, and then run the script. Alternatively, you can run `sol_12_01_08.sql`.

```
<customer id="105">
     <first_name></first_name>
     <last_name></last_name>
          <orders>
               <cust_order id="">
                    <order_date></order_date>
                    <order_mode></order_mode>
               <cust_order>
          …
          </orders>
<customer>
```

Answer:

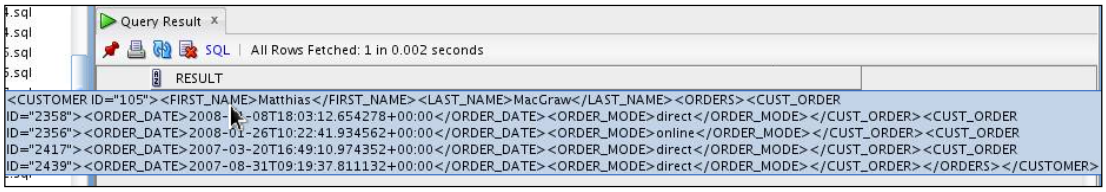
```
select
XMLELEMENT(Customer, XMLATTRIBUTES(customer_id id),
XMLFOREST(cust_first_name "FIRST_NAME", cust_last_name
"LAST_NAME"),
XMLELEMENT(orders,
(select XMLAGG(XMLELEMENT(Cust_order,
XMLATTRIBUTES(order_id id),
XMLFOREST(order_date, order_mode)))
from orders o
where o.customer_id= c.customer_id)))
as result
from customers c
where customer_id=105;
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



9. Which of the following statements are true?
   a. The `XMLPI()` function returns a value even if the second argument (`value_expr`) is omitted. **TRUE**
   b. The `XMLROOT()` function is used to generate the root element for an XML document. **FALSE**
   c. In the `XMLSERIALIZE()` and `XMLPARSE()` functions, if you specify `DOCUMENT` before the first argument, the resulting `XMLType` instance is not validated against the rules of a well-formed document. **FALSE**
10. Generate XML data from the `EMPLOYEES` table (**`hr`** schema) in the following format for each employee whose first name starts with the letter A. Save your SQL statement in a text file named `lab_12_01_10.sql`, and then run the script. Alternatively, you can run `sol_12_01_10.sql`.

```
<column name="First_Name"></column>
<column name="Last_Name"></column>
<column name="Joining_Date"></column>
<column name="Department_Name"></column>
```

Answer:

```
select
XMLCOLATTVAL(e.first_name "First_Name",
e.last_name "Last_Name", e.hire_date "Joining_Date",
d.department_name "Department_Name")
as result
from employees e, departments d
where
e.DEPARTMENT_ID=d.department_id and e.first_name like 'A%';
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Output:

Query Result

SQL | All Rows Fetched: 10 in 0.009 seconds

	RESULT
1	<column name = "First_Name">Amit</column><column name = "Last_Name">Banda</...
2	<column name = "First_Name">Alexis</column><column name = "Last_Name">Bull<...
3	<column name = "First_Name">Anthony</column><column name = "Last_Name">Cabr...
4	<column name = "First_Name">Alberto</column><column name = "Last_Name">Erra...
5	<column name = "First_Name">Adam</column><column name = "Last_Name">Fripp</...
6	<column name = "First_Name">Alexander</column><column name = "Last_Name">Hu...
7	<column name = "First_Name">Alyssa</column><column name = "Last_Name">Hutto...
8	<column name = "First_Name">Alexander</column><column name = "Last_Name">Kh...
9	<column name = "First_Name">Allan</column><column name = "Last_Name">McEwen...
10	<column name = "First_Name">Alana</column><column name = "Last_Name">Walsh<...

11. Use the **hr** schema to create an XML document for each employee who joined in the year 2008. The document should have the employee's ID, hire date, and salary. Name the root element `EMPLOYEE`. Save your SQL statement in a text file named `lab_08_01_11.sql`, and then run the script. Alternatively, you can run `sol_08_01_11.sql`.

Answer:

```
select
SYS_XMLGEN(XMLFOREST (employee_id,hire_date,salary),
XMLFORMAT.CREATEFORMAT('EMPLOYEE'))
as result
from employees
where hire_date like '%08'
```

Query Result

SQL | All Rows Fetched: 11 in 0.01 seconds

	RESULT
1	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>128</EMPLOYEE_ID><HIRE_DATE>20...
2	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>136</EMPLOYEE_ID><HIRE_DATE>20...
3	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>149</EMPLOYEE_ID><HIRE_DATE>20...
4	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>164</EMPLOYEE_ID><HIRE_DATE>20...
5	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>165</EMPLOYEE_ID><HIRE_DATE>20...
6	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>166</EMPLOYEE_ID><HIRE_DATE>20...
7	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>167</EMPLOYEE_ID><HIRE_DATE>20...
8	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>173</EMPLOYEE_ID><HIRE_DATE>20...
9	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>179</EMPLOYEE_ID><HIRE_DATE>20...
10	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>183</EMPLOYEE_ID><HIRE_DATE>20...
11	<?xml version="1.0"?><EMPLOYEE><EMPLOYEE_ID>199</EMPLOYEE_ID><HIRE_DATE>20...

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

12. Using the `hr` schema, modify `lab_12_01_11.sql` to retrieve the same details of all employees in a single XML document. Name the root element `EMPLOYEES`. Your output should be in the following format. Save your SQL statement in a text file named `lab_12_01_12.sql`, and then run the script. Alternatively, you can run `sol_12_01_12.sql`.

```
<?xml version='1.0'?>
      <EMPLOYEES>
            <EMPLOYEE>
                  <EMPLOYEE_ID></EMPLOYEE_ID>
                  <HIRE_DATE></HIRE_DATE>
                  <SALARY></SALARY>
            </EMPLOYEE>
            …
      </EMPLOYEES>
```

Answer:

```
select
SYS_XMLAGG(
SYS_XMLGEN(XMLFOREST( employee_id,hire_date,salary ),
XMLFORMAT.CREATEFORMAT('EMPLOYEE')),
XMLFORMAT.CREATEFORMAT('EMPLOYEES'))
as result
from employees
where hire_date like '%08'
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Output:

```
RESULT
-----------------------------------------
<?xml version="1.0"?>
<EMPLOYEES>
<EMPLOYEE>
  <EMPLOYEE_ID>128</EMPLOYEE_ID>
  <HIRE_DATE>2008-03-08</HIRE_DATE>
  <SALARY>2200</SALARY>
</EMPLOYEE>
<EMPLOYEE>
  <EMPLOYEE_ID>136</EMPLOYEE_ID>
  <HIRE_DATE>2008-02-06</HIRE_DATE>
  <SALARY>2200</SALARY>
</EMPLOYEE>
<EMPLOYEE>
  <EMPLOYEE_ID>149</EMPLOYEE_ID>
  <HIRE_DATE>2008-01-29</HIRE_DATE>
  <SALARY>10500</SALARY>
</EMPLOYEE>
<EMPLOYEE>
  <EMPLOYEE_ID>164</EMPLOYEE_ID>
  <HIRE_DATE>2008-01-24</HIRE_DATE>
  <SALARY>7200</SALARY>
</EMPLOYEE>
<EMPLOYEE>
  <EMPLOYEE_ID>165</EMPLOYEE_ID>
  <HIRE_DATE>2008-02-23</HIRE_DATE>
  <SALARY>6800</SALARY>
</EMPLOYEE>
<EMPLOYEE>
  <EMPLOYEE_ID>166</EMPLOYEE_ID>
  <HIRE_DATE>2008-03-24</HIRE_DATE>
  <SALARY>6400</SALARY>
</EMPLOYEE>
<EMPLOYEE>
```

```
  <EMPLOYEE_ID>167</EMPLOYEE_ID>
  <HIRE_DATE>2008-04-21</HIRE_DATE>
  <SALARY>6200</SALARY>
</EMPLOYEE>
<EMPLOYEE>
  <EMPLOYEE_ID>173</EMPLOYEE_ID>
  <HIRE_DATE>2008-04-21</HIRE_DATE>
  <SALARY>6100</SALARY>
</EMPLOYEE>
<EMPLOYEE>
  <EMPLOYEE_ID>179</EMPLOYEE_ID>
  <HIRE_DATE>2008-01-04</HIRE_DATE>
  <SALARY>6200</SALARY>
</EMPLOYEE>
<EMPLOYEE>
  <EMPLOYEE_ID>183</EMPLOYEE_ID>
  <HIRE_DATE>2008-02-03</HIRE_DATE>
  <SALARY>2800</SALARY>
</EMPLOYEE>
<EMPLOYEE>
  <EMPLOYEE_ID>199</EMPLOYEE_ID>
  <HIRE_DATE>2008-01-13</HIRE_DATE>
  <SALARY>2600</SALARY>
</EMPLOYEE>
</EMPLOYEES>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 12-2: Generating XML Using `DBMS_XMLGEN`

In this practice, you use the procedures in the `DBMS_XMLGEN` package to generate XML data.

1. Open a SQL Worksheet for the `hr` schema. Create and run a PL/SQL procedure to generate details of the employees who work in department 60, in XML format from the `EMPLOYEES` table. Save your procedure in a text file named `lab_12_02_01.sql`, and then run the script. Confirm that the data was generated successfully by using a query.

   Answer:

```
SET long 20000

CREATE TABLE temp_clob_tab (result CLOB);

DECLARE
  qryCtx DBMS_XMLGEN.ctxHandle;
  result CLOB;
BEGIN
  qryCtx := DBMS_XMLGEN.newContext('SELECT * FROM hr.employees
where department_id=60');
  -- Set the row header to be EMPLOYEE
  DBMS_XMLGEN.setRowTag(qryCtx, 'EMPLOYEE');
  -- Get the result
  result := DBMS_XMLGEN.getXML(qryCtx);
  INSERT INTO temp_clob_tab VALUES(result);
  --Close context
  DBMS_XMLGEN.closeContext(qryCtx);
END;
/
SELECT * from temp_clob_tab;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Script Output

Task completed in 0.497 seconds

```
table TEMP_CLOB_TAB dropped.
table TEMP_CLOB_TAB created.
anonymous block completed
RESULT
--------------------------------------------------
<?xml version="1.0"?>
<ROWSET>
 <EMPLOYEE>
  <EMPLOYEE_ID>103</EMPLOYEE_ID>
  <FIRST_NAME>Alexander</FIRST_NAME>
  <LAST_NAME>Hunold</LAST_NAME>
  <EMAIL>AHUNOLD</EMAIL>
  <PHONE_NUMBER>590.423.4567</PHONE_NUMBER>
  <HIRE_DATE>03-JAN-06</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>9000</SALARY>
  <MANAGER_ID>102</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </EMPLOYEE>
 <EMPLOYEE>
  <EMPLOYEE_ID>104</EMPLOYEE_ID>
  <FIRST_NAME>Bruce</FIRST_NAME>
  <LAST_NAME>Ernst</LAST_NAME>
  <EMAIL>BERNST</EMAIL>
  <PHONE_NUMBER>590.423.4568</PHONE_NUMBER>
  <HIRE_DATE>21-MAY-07</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>6000</SALARY>
  <MANAGER_ID>103</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </EMPLOYEE>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
 </EMPLOYEE>
 <EMPLOYEE>
  <EMPLOYEE_ID>105</EMPLOYEE_ID>
  <FIRST_NAME>David</FIRST_NAME>
  <LAST_NAME>Austin</LAST_NAME>
  <EMAIL>DAUSTIN</EMAIL>
  <PHONE_NUMBER>590.423.4569</PHONE_NUMBER>
  <HIRE_DATE>25-JUN-05</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>4800</SALARY>
  <MANAGER_ID>103</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </EMPLOYEE>
 <EMPLOYEE>
  <EMPLOYEE_ID>106</EMPLOYEE_ID>
  <FIRST_NAME>Valli</FIRST_NAME>
  <LAST_NAME>Pataballa</LAST_NAME>
  <EMAIL>VPATABAL</EMAIL>
  <PHONE_NUMBER>590.423.4560</PHONE_NUMBER>
  <HIRE_DATE>05-FEB-06</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>4800</SALARY>
  <MANAGER_ID>103</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </EMPLOYEE>
 <EMPLOYEE>
  <EMPLOYEE_ID>107</EMPLOYEE_ID>
  <FIRST_NAME>Diana</FIRST_NAME>
  <LAST_NAME>Lorentz</LAST_NAME>
  <EMAIL>DLORENTZ</EMAIL>
  <PHONE_NUMBER>590.423.5567</PHONE_NUMBER>
```

```
 <EMPLOYEE>
  <EMPLOYEE_ID>107</EMPLOYEE_ID>
  <FIRST_NAME>Diana</FIRST_NAME>
  <LAST_NAME>Lorentz</LAST_NAME>
  <EMAIL>DLORENTZ</EMAIL>
  <PHONE_NUMBER>590.423.5567</PHONE_NUMBER>
  <HIRE_DATE>07-FEB-07</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>4200</SALARY>
  <MANAGER_ID>103</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </EMPLOYEE>
</ROWSET>
```

2. Modify the procedure in `lab_12_02_01.sql` to fetch two rows at a time. Save the new procedure in a text file named `lab_12_02_02.sql`. Run the procedure in the `hr` schema and confirm that the XML data was generated successfully.

   Answer:

```
CREATE TABLE temp_clob_tab (result clob);
DECLARE
  qryCtx DBMS_XMLGEN.ctxHandle;
  result CLOB;
BEGIN
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
  -- Get the query context;
  qryCtx := DBMS_XMLGEN.newContext('SELECT * FROM hr.employees
where department_id=60');
  -- Set the maximum number of rows to be 2
  DBMS_XMLGEN.setMaxRows(qryCtx, 2);
  LOOP
    -- Get the result
    result := DBMS_XMLGEN.getXML(qryCtx);
    -- If no rows were processed, then quit
    EXIT WHEN DBMS_XMLGEN.getNumRowsProcessed(qryCtx) = 0;

    -- Do some processing with the lob data
    --   Here, we insert the results into a table.
    --   You can print the lob out, output it to a stream,
    --   put it in a queue, or do any other processing.
    INSERT INTO temp_clob_tab VALUES(result);
  END LOOP;
  --close context
  DBMS_XMLGEN.closeContext(qryCtx);
END;
/

SELECT * FROM temp_clob_tab;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Script Output

Task completed in 0.009 seconds

```
table TEMP_CLOB_TAB dropped.
table TEMP_CLOB_TAB created.
anonymous block completed
RESULT
--------------------------------------------------
<?xml version="1.0"?>
<ROWSET>
 <ROW>
  <EMPLOYEE_ID>103</EMPLOYEE_ID>
  <FIRST_NAME>Alexander</FIRST_NAME>
  <LAST_NAME>Hunold</LAST_NAME>
  <EMAIL>AHUNOLD</EMAIL>
  <PHONE_NUMBER>590.423.4567</PHONE_NUMBER>
  <HIRE_DATE>03-JAN-06</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>9000</SALARY>
  <MANAGER_ID>102</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </ROW>
 <ROW>
  <EMPLOYEE_ID>104</EMPLOYEE_ID>
  <FIRST_NAME>Bruce</FIRST_NAME>
  <LAST_NAME>Ernst</LAST_NAME>
  <EMAIL>BERNST</EMAIL>
  <PHONE_NUMBER>590.423.4568</PHONE_NUMBER>
  <HIRE_DATE>21-MAY-07</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>6000</SALARY>
  <MANAGER_ID>103</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </ROW>
</ROWSET>


<?xml version="1.0"?>
<ROWSET>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
<?xml version="1.0"?>
<ROWSET>
 <ROW>
  <EMPLOYEE_ID>105</EMPLOYEE_ID>
  <FIRST_NAME>David</FIRST_NAME>
  <LAST_NAME>Austin</LAST_NAME>
  <EMAIL>DAUSTIN</EMAIL>
  <PHONE_NUMBER>590.423.4569</PHONE_NUMBER>
  <HIRE_DATE>25-JUN-05</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>4800</SALARY>
  <MANAGER_ID>103</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </ROW>
 <ROW>
  <EMPLOYEE_ID>106</EMPLOYEE_ID>
  <FIRST_NAME>Valli</FIRST_NAME>
  <LAST_NAME>Pataballa</LAST_NAME>
  <EMAIL>VPATABAL</EMAIL>
  <PHONE_NUMBER>590.423.4560</PHONE_NUMBER>
  <HIRE_DATE>05-FEB-06</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>4800</SALARY>
  <MANAGER_ID>103</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </ROW>
</ROWSET>
```

```
<?xml version="1.0"?>
<ROWSET>
 <ROW>
  <EMPLOYEE_ID>107</EMPLOYEE_ID>
  <FIRST_NAME>Diana</FIRST_NAME>
  <LAST_NAME>Lorentz</LAST_NAME>
  <EMAIL>DLORENTZ</EMAIL>
  <PHONE_NUMBER>590.423.5567</PHONE_NUMBER>
  <HIRE_DATE>07-FEB-07</HIRE_DATE>
  <JOB_ID>IT_PROG</JOB_ID>
  <SALARY>4200</SALARY>
  <MANAGER_ID>103</MANAGER_ID>
  <DEPARTMENT_ID>60</DEPARTMENT_ID>
 </ROW>
</ROWSET>
```


3. XML transformation in Oracle Database 11*g* is done: (Choose the correct answer.)
   a. Only by using XDK
   b. **In the server and by using XDK**
   c. Only in the server

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 13: Transforming and Validating XMLType Data

Chapter 13

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 13-1: Transforming and Validating XML Data

Overview

XML documents have structure but no format. To add format to the XML documents you can use Extensible Stylesheet Language (`XSL`). `XSL` provides a way of displaying XML semantics. It can map XML elements into other formatting or mark-up languages such as HTML.

In Oracle XML DB, `XMLType` instances or XML data stored in `XMLType` tables, columns, or views in Oracle Database, can be (formatted) transformed into HTML, XML, and other mark-up languages, using `XSL` style sheets and `XMLType` method `transform()`. This process conforms to the W3C `XSL` Transformations 1.0 Recommendation.

In this guided practice, you learn how to use Oracle SQL function `XMLtransform` and `XMLType` method `transform()` to transform XML data stored as `XMLType` to various formats.

Assumptions

Tasks

1. In this step, you set up an XML schema and tables that are needed to run other code examples in this practice. The call to `deleteSchema` here ensures that there is no existing XML schema before creating one. If no such schema exists, then `deleteSchema` produces an error. Log in as `oe/oe` in SQL*Plus.

```
sqlplus oe/oe
```

2. Review the script `sol_13_01_02.sql`, and then run it.

```
-- Delete the schema, if it already exists.
BEGIN
DBMS_XMLSCHEMA.deleteSchema('http://www.example.com/schemas/ipo.xsd'
,4);
END;
/
-- Register the schema
BEGIN
DBMS_XMLSCHEMA.registerSchema(
SCHEMAURL => 'http://www.example.com/schemas/ipo.xsd',
SCHEMADOC => '<schema targetNamespace="http://www.example.com/IPO"
xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:ipo="http://www.example.com/IPO">
<!-- annotation>
<documentation xml:lang="en">
International Purchase order schema for Example.com
Copyright 2000 Example.com. All rights reserved.
</documentation>
</annotation -->
<element name="purchaseOrder" type="ipo:PurchaseOrderType"/>
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
<element name="comment" type="string"/>
<complexType name="PurchaseOrderType">
<sequence>
<element name="shipTo" type="ipo:Address"/>
<element name="billTo" type="ipo:Address"/>
<element ref="ipo:comment" minOccurs="0"/>
<element name="items" type="ipo:Items"/>
</sequence>
<attribute name="orderDate" type="date"/>
</complexType>
<complexType name="Items">
<sequence>
<element name="item" minOccurs="0" maxOccurs="unbounded">
<complexType>
<sequence>
<element name="productName" type="string"/>
<element name="quantity">
<simpleType>
<restriction base="positiveInteger">
<maxExclusive value="100"/>
</restriction>
</simpleType>
</element>
<element name="USPrice" type="decimal"/>
<element ref="ipo:comment" minOccurs="0"/>
<element name="shipDate" type="date" minOccurs="0"/>
</sequence>
<attribute name="partNum" type="ipo:SKU" use="required"/>
</complexType>
</element>
</sequence>
</complexType>
<complexType name="Address">
<sequence>
<element name="name" type="string"/>
<element name="street" type="string"/>
<element name="city" type="string"/>
<element name="state" type="string"/>
<element name="country" type="string"/>
<element name="zip" type="string"/>
</sequence>
</complexType>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
<simpleType name="SKU">
<restriction base="string">
<pattern value="[0-9]{3}-[A-Z]{2}"/>
</restriction>
</simpleType>
</schema>',
LOCAL => TRUE,
GENTYPES => TRUE);

END;
/

-- Create table to hold XML instance documents.
DROP TABLE po_tab;
CREATE TABLE po_tab (id NUMBER, xmlcol XMLType)
XMLType COLUMN xmlcol
XMLSCHEMA "http://www.example.com/schemas/ipo.xsd"
ELEMENT "purchaseOrder";
/

-- Insert some data into the newly created table.
INSERT INTO po_tab
VALUES(1, XMLType(
'<?xml version="1.0"?>
<ipo:purchaseOrder
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ipo="http://www.example.com/IPO"
xsi:schemaLocation="http://www.example.com/IPO
http://www.example.com/schemas/ipo.xsd"
orderDate="1999-12-01">
<shipTo>
<name>Helen Zoe</name>
<street>121 Broadway</street>
<city>Cardiff</city>
<state>Wales</state>
<country>UK</country>
<zip>CF2 1QJ</zip>
</shipTo>
<billTo>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
SQL> @sol_13_01_02

PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.


Table dropped.


Table created.

CREATE TABLE po_tab (id NUMBER, xmlcol XMLType)
             *
ERROR at line 1:
ORA-00955: name is already used by an existing object



1 row created.
```

3. Review the script `sol_13_01_03.sql`, and then run it as `oe/oe`. This script demonstrates retrieving a style sheet using `XMLTRANSFORM` and `DBURITYPE`. Refer to chapter 20, *Accessing Data Through URIs*, in the *Oracle XML DB Developer's Guide 12c Release 1 (12.1)* for information about `DBURIType`.

```
DROP TABLE stylesheet_tab;
CREATE TABLE stylesheet_tab (id NUMBER, stylesheet XMLType);
INSERT INTO stylesheet_tab
VALUES (1,
XMLType(
'<?xml version="1.0" ?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="*">
<td>
<xsl:choose>
<xsl:when test="count(child::*) > 1">
<xsl:call-template name="nested"/>
</xsl:when>
<xsl:otherwise>
<xsl:value-of select="name(.)"/>:<xsl:value-of
select="text()"/>
</xsl:otherwise>
</xsl:choose>
</td>
</xsl:template>
<xsl:template match="*" name="nested" priority="-1" mode="nested2">
<b>
<!-- xsl:value-of select="count(child::*)"/ -->
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
<xsl:choose>
<xsl:when test="count(child::*) > 1">
<xsl:value-of select="name(.)"/>:<xsl:apply-templates
mode="nested2"/>
</xsl:when>
<xsl:otherwise>
<xsl:value-of select="name(.)"/>:<xsl:value-of
select="text()"/>
</xsl:otherwise>
</xsl:choose>
</b>
</xsl:template>
</xsl:stylesheet>'));
```

```
SQL> @sol_13_01_03

Table dropped.


Table created.


1 row created.

SQL>
```

4. Review the script `sol_13_01_04.sql`, and then run it as `oe/oe`.

```
SELECT XMLSerialize(DOCUMENT XMLtransform(x.xmlcol, y.stylesheet)
AS varchar2(2000))
AS result FROM po_tab x, stylesheet_tab y WHERE y.id = 1;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
SQL> @sol_13_01_04

RESULT
--------------------------------------------------------------------------------

<td>
 <b>ipo:purchaseOrder:
  <b>shipTo:
   <b>name:Helen Zoe</b>
   <b>street:121 Broadway</b>
   <b>city:Cardiff</b>
   <b>state:Wales</b>
   <b>country:UK</b>
   <b>zip:CF2 1QJ</b>
  </b>

RESULT
--------------------------------------------------------------------------------
  <b>billTo:
   <b>name:Robert Smith</b>
   <b>street:8 Oak Avenue</b>
   <b>city:Old Town</b>
   <b>state:CA</b>
   <b>country:US</b>
   <b>zip:95819</b>
  </b>
  <b>items:</b>
 </b>
</td>

RESULT
--------------------------------------------------------------------------------


SQL>
```

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 14: Working With the Oracle XML DB Repository

Chapter 14

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practice 14-1: Using Oracle XML DB Repository

In this practice, you create repository folders and resources by using both `DBMS_REPOS_XDB`, and Internet protocols such as FTP. You then use the `RESOURCE_VIEW` API and the Internet protocols to view the resource's metadata and content.

Creating and Accessing Resources

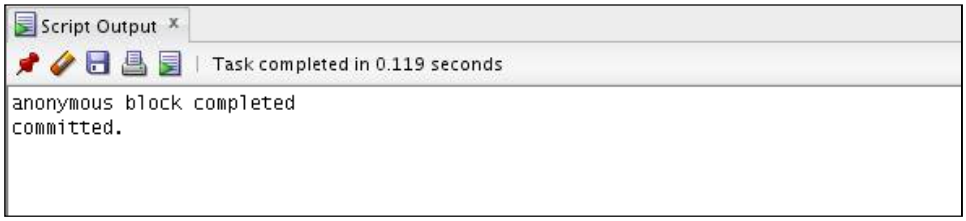
1. Open a SQL Worksheet for the `hr` schema. By using `DBMS_XDB_REPOS`, create an anonymous block to create the following table structure within '`/home`':

```
hr
hr/xsd
hr/xsl
hr/purchaseOrders
hr/purchaseOrders/2001
hr/purchaseOrders/2002
```

You may use `lab_14_01_01.sql` for help. Replace <***TODO***> with the name of the method used to create the repository folders. Save the file and run `lab_14_01_01.sql`. Alternatively, you can run `sol_14_01_01.sql`.

Answer:

```
Declare
  result BOOLEAN;
BEGIN
  result := DBMS_XDB_RPOS.createFolder('/home/HR/');
  result := DBMS_XDB_RPOS.createFolder('/home/HR/xsd');
  result := DBMS_XDB_RPOS.createFolder('/home/HR/xsl');
  result :=
DBMS_XDB_RPOS.createFolder('/home/HR/purchaseOrders');
result
:=DBMS_XDB_RPOS.createFolder('/home/HR/purchaseOrders/2001');
result
:=DBMS_XDB_RPOS.createFolder('/home/HR/purchaseOrders/2002');
END;
/
commit;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

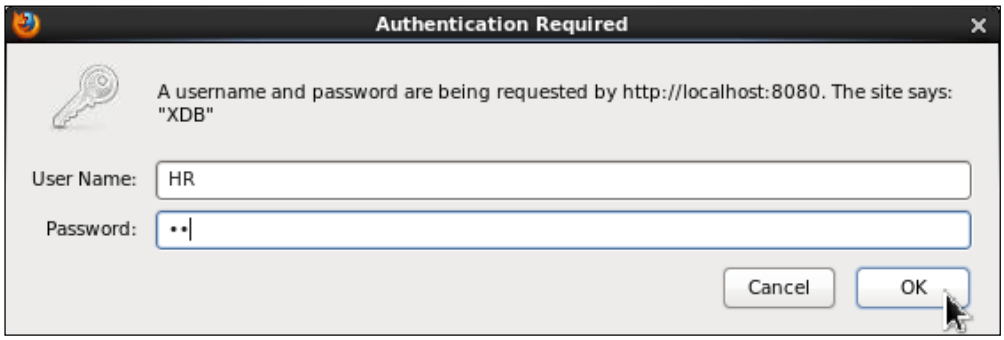
2. By using HTTP, view the folder structure that was created in step 1.

   Steps:

   a. To start an HTTP session by using the Internet browser, enter the machine name (or IP address) followed by a colon and port number as a parameter to FTP.

   Example: http://localhost:8080/

   b. Enter the **User Name** as `HR` and **Password** as `hr` if prompted, and then click **OK**.

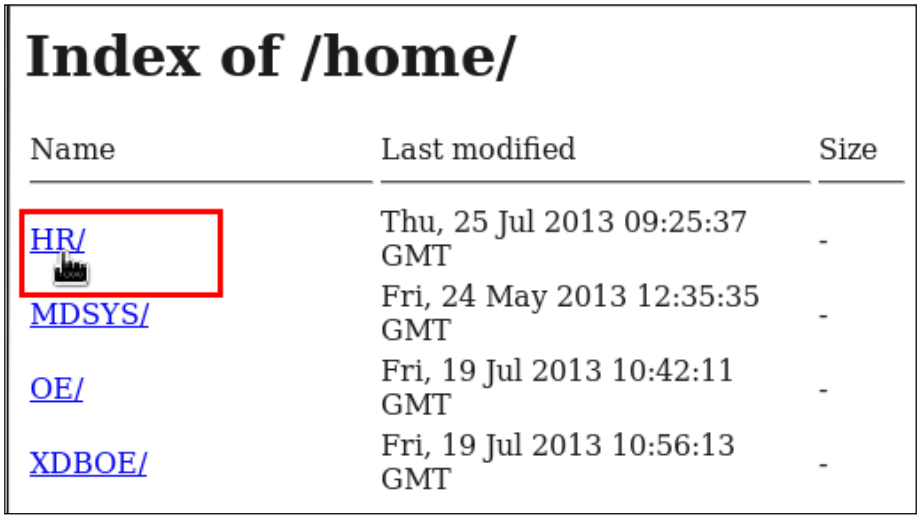


Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

c. You can now browse through the folders that you created in step 1.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Index of /home/HR/

Name	Last modified	Size
purchaseOrders/	Thu, 25 Jul 2013 09:25:37 GMT	-
xsd/	Thu, 25 Jul 2013 09:25:37 GMT	-
xsl/	Thu, 25 Jul 2013 09:25:37 GMT	-

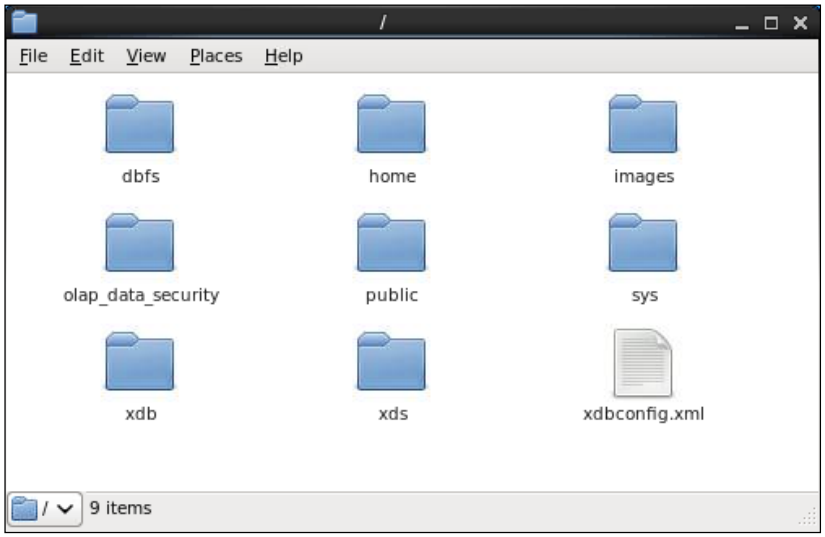
Index of /home/HR/purchaseOrders/

Name	Last modified	Size
2001/	Thu, 25 Jul 2013 09:25:37 GMT	-
2002/	Thu, 25 Jul 2013 09:25:37 GMT	-

3. By using the WEBDAV connection that you created in practice 12 (steps 1-3), download `/home/OE/purchaseOrder.xsd` to `/home/oracle/labs/xml_dir/PO_New`.

   Steps:

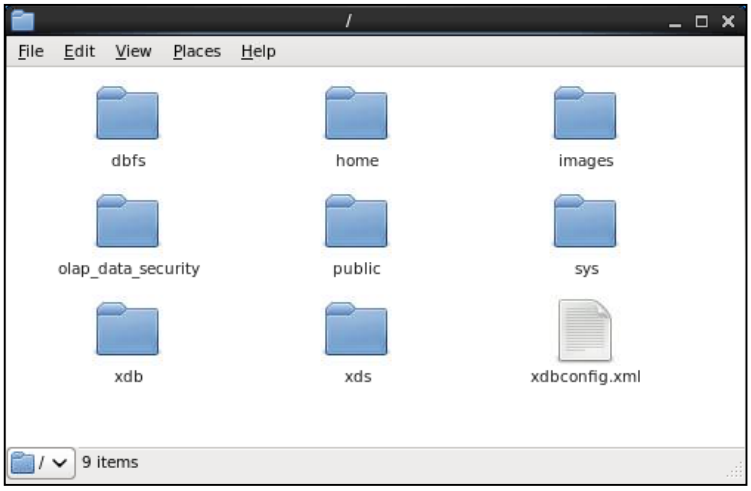
   a. Start a WEBDAV session by clicking the localhost-hr icon on your desktop (or whatever you called it) or from the Places menu.

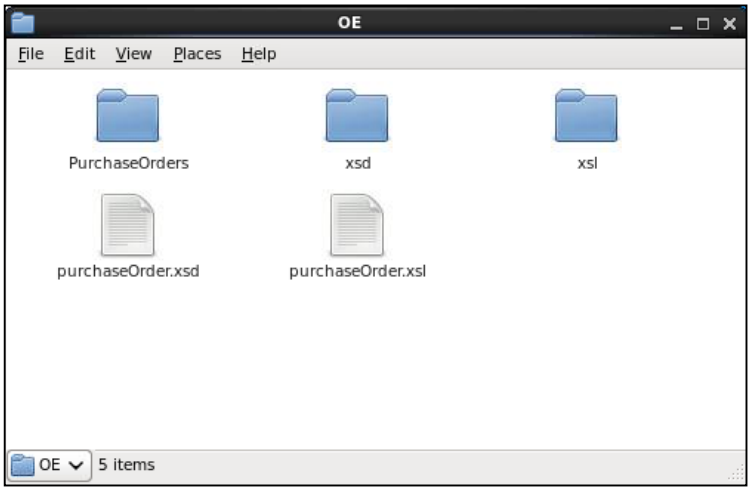


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

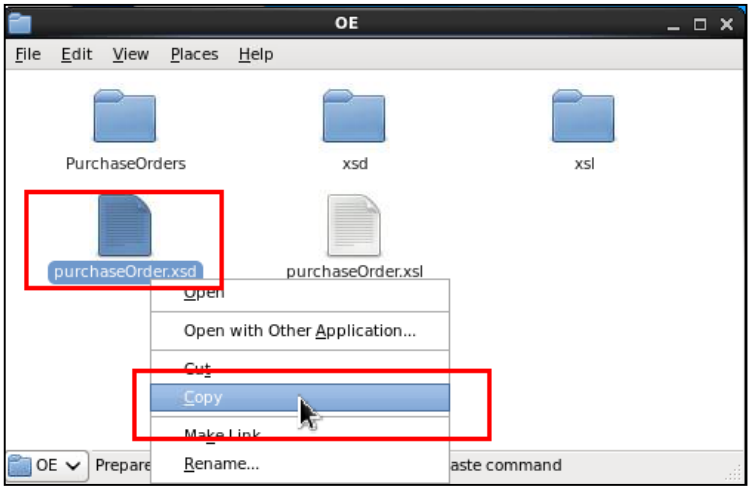
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

b. Browse to the `/home/OE/purchaseOrder.xsd` location.





c. Right-click the `purchaseOrder.xsd` file and copy it.

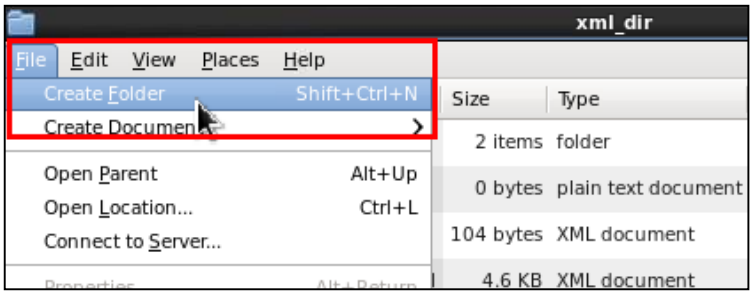


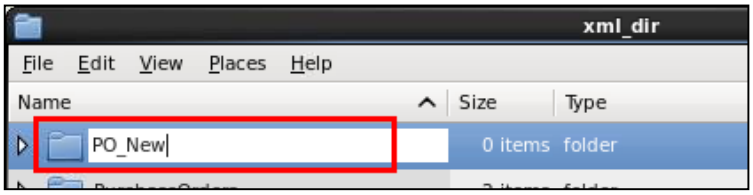
Oracle University and Error : You are not a Valid Partner use only

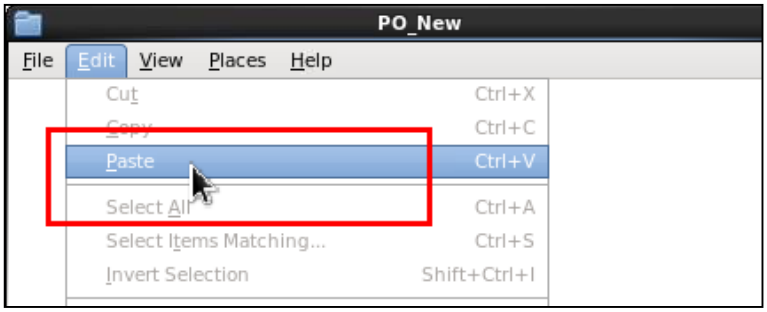
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

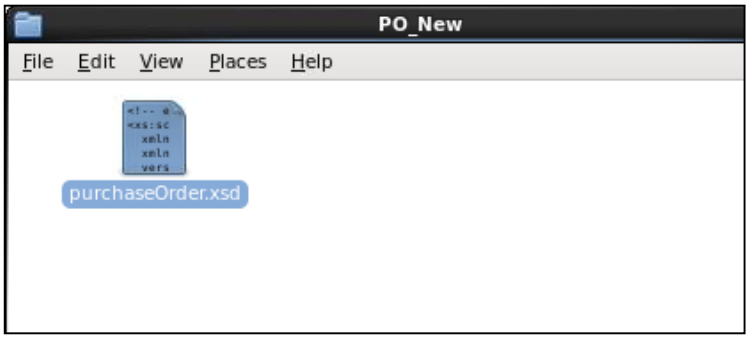
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

d. In your local file system, browse to the `/home/oracle/labs/xml_dir` location, create a new `PO_New` folder, and then paste the copied file into this new folder.









Oracle University and Error : You are not a Valid Partner use only

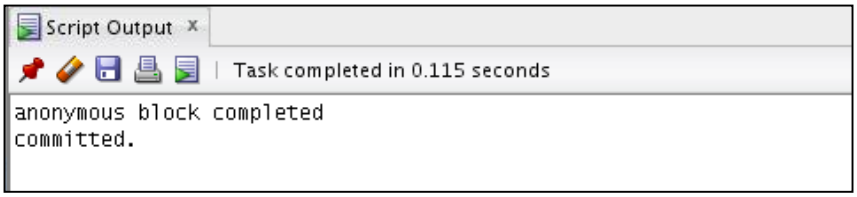
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Open a SQL Worksheet for the `hr` schema. By using `DBMS_XDB_REPOS`, create a '`purchaseOrder.xsd`' resource in the `/home/HR/xsd` folder and store the `/home/oracle/labs/xml_dir/purchaseOrder.xsd` file in it. You may use `lab_14_01_04.sql` for help. Replace <***TODO1***> with the name of the method used to create the repository resources and save the file. Run `lab_14_01_04.sql`. Alternatively, you can run `sol_14_01_04.sql`.

   Answer:

```
DECLARE
  res BOOLEAN;
BEGIN
  res :=
DBMS_XDB_REPOS.createResource('/home/HR/xsd/purchaseOrder.xsd',
bfilename('XML_DIR',  'purchaseOrder.xsd'),
nls_charset_id('AL32UTF8'));
END;
/
commit;
```



5. By using the `UNDER_PATH()` function, select the `any_path` column from the `RESOURCE_VIEW` API to display the tree structure in '`/home/hr`'. Use the `hr` schema.

   Answer:

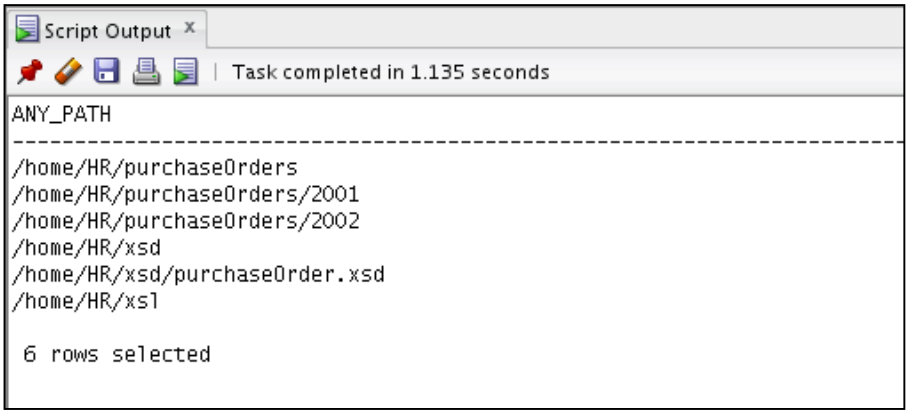
   Run `sol_14_01_05.sql`.

```
select any_path
from resource_view
where under_path (res,'/home/HR') = 1;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

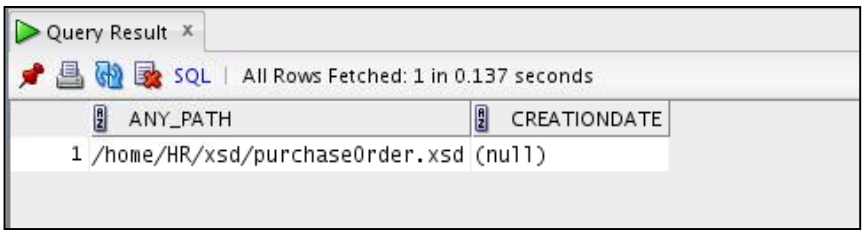


6. By using `XMLQUERY`, display the creation date ('`/Resource/CreationDate`') for the '`/home/HR/xsd`' path. Use the SQL Worksheet for the `hr` schema.

   Answer:

   Run `sol_14_01_06.sql`.

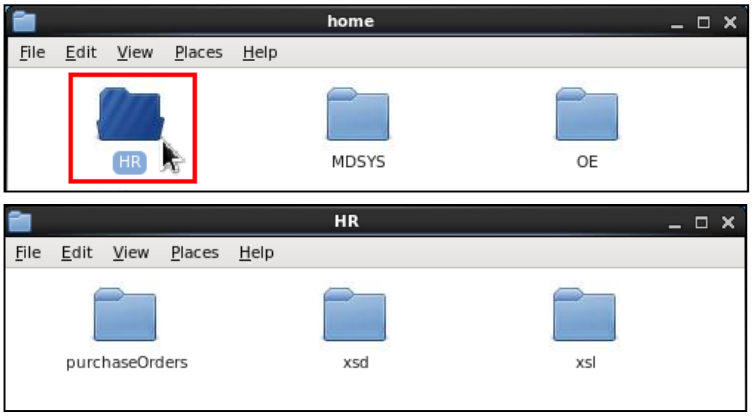
```
select ANY_PATH, XMLQUERY('//Resource/CreationDate'
  passing RES returning content) AS creationdate
from resource_view
where under_path (Res,'/home/HR/xsd', 1) = 1;
```



7. By using WEBDAV, create an `xml` folder in '`/home/HR/`'.

   Steps:

   a. Open a WEBDAV session and browse to the `/home/HR/` location.

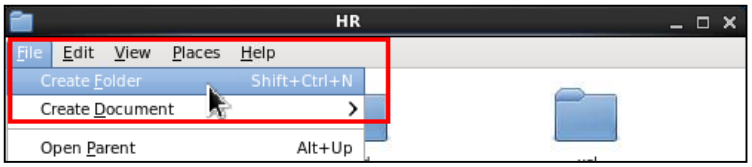


Oracle University and Error : You are not a Valid Partner use only

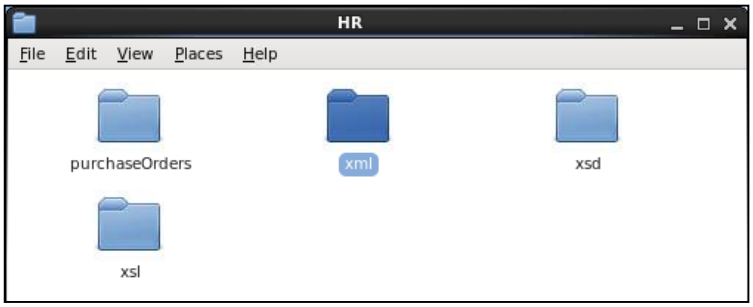
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

b. Click **File > Create Folder** from the File menu.



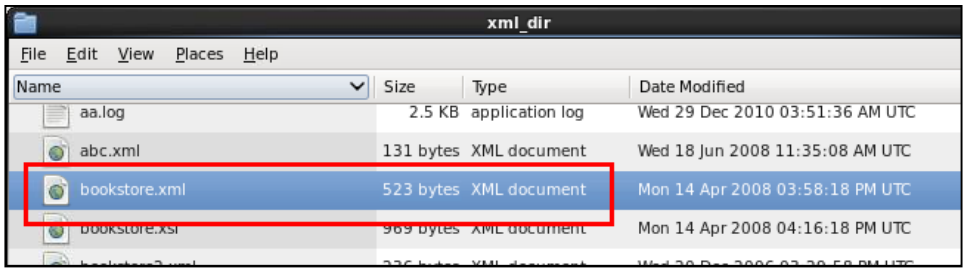
c. Name the folder `xml`.

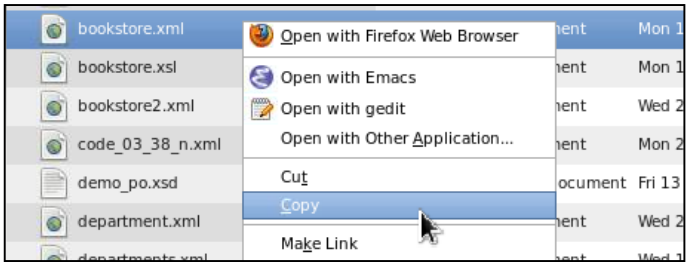


8. By using WEBDAV, upload the `/home/oracle/labs/xml_dir/bookstore.xml` file to `/home/hr/xml`.

Steps:

a. Copy the `bookstore.xml` file from `/home/oracle/labs/xml_dir`.

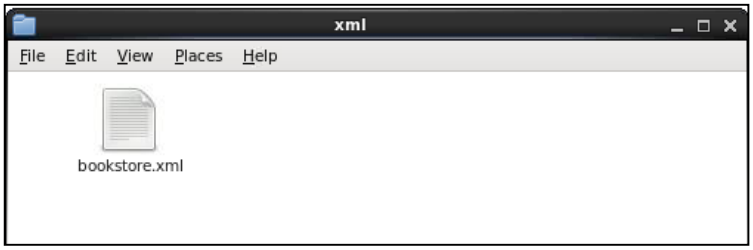




Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

b. Open a WEBDAV session, browse to the `/home/hr/xml` location, and then paste the copied file.



THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practice 14-2: Working with Access Control Lists (ACLs)

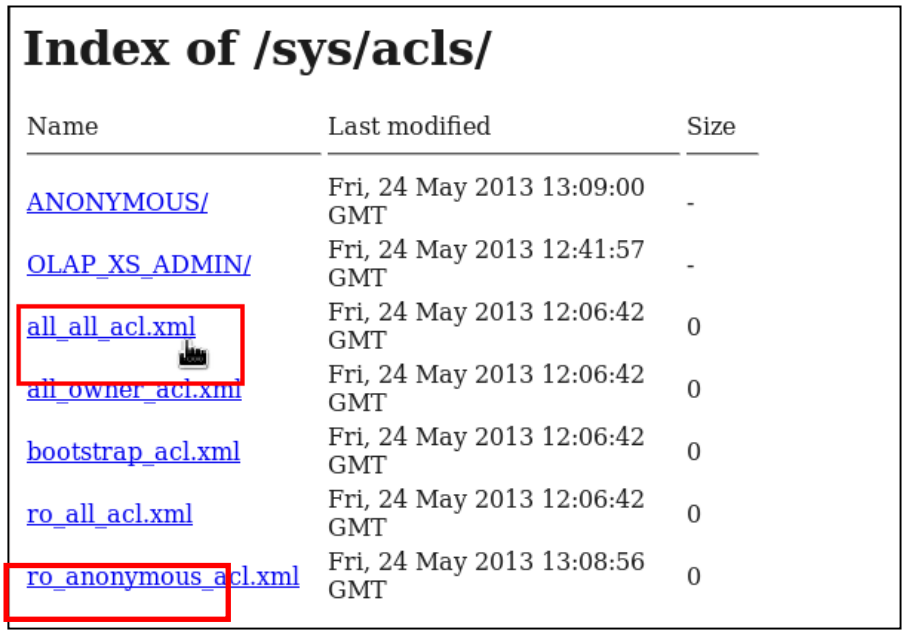
Overview

In this practice, you display the contents of the access control lists (ACLs), and view and change the ACLs attached to the repository resources. There is also a quiz to check your understanding of Oracle XML DB Repository architecture, compound documents, and repository events.

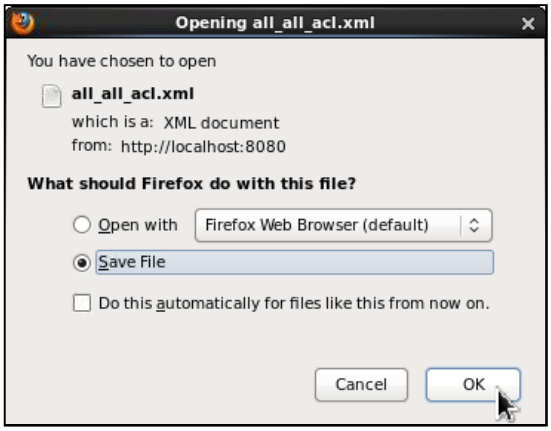
1. By using HTTP, display the contents of `/sys/acls/all_all_acl.xml`.

   Steps:

   a. Open an HTTP session in Firefox and browse to the `/sys/acls/all_all_acl.xml` location.

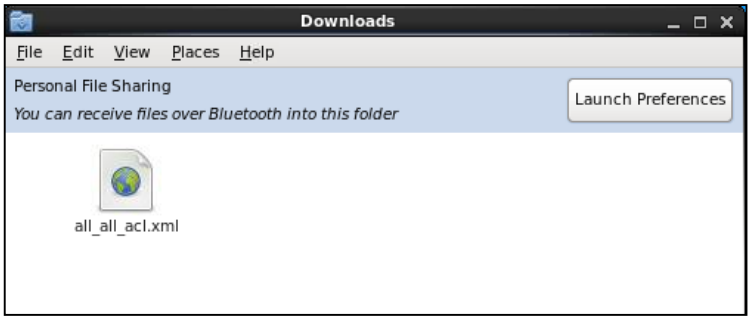


   b. Click the `all_all_acl.xml` file to view its content. (Enter your username and password, `HR` and `hr`, if prompted.) Select **Save File** and save the file. The file should be downloaded to the `/home/oracle/Downloads` folder.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

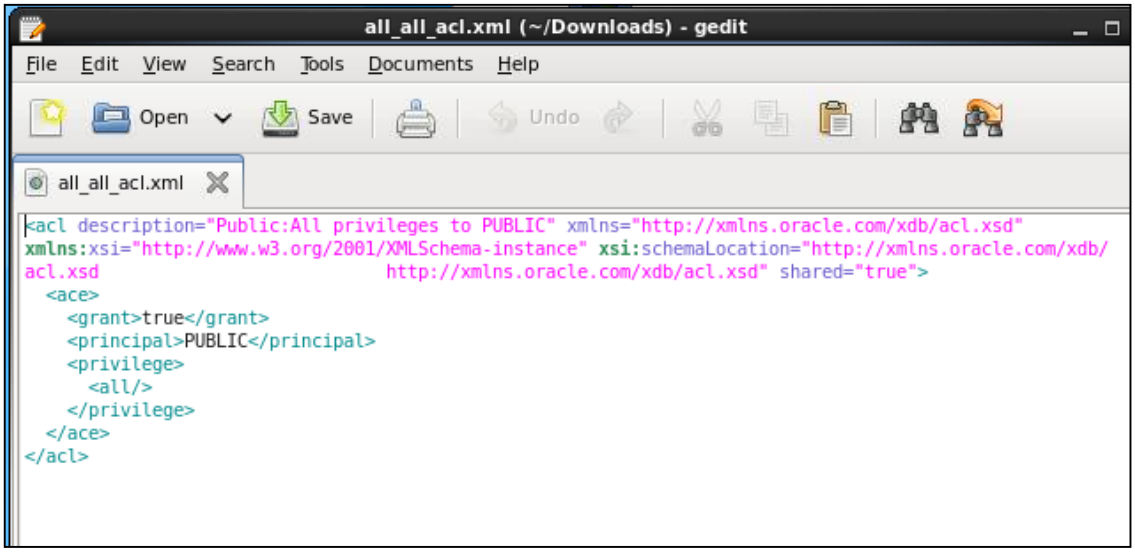


c. Double-click the `all_all_acl.xml` file to view its contents.

The contents of the file using Mozilla Firefox:

```
-<acl description="Public:All privileges to PUBLIC" xsi:schemaLocation="http://xmlns.oracle.com/xdb/acl.xsd ht
 shared="true">
  -<ace>
     <grant>true</grant>
     <principal>PUBLIC</principal>
   -<privilege>
      <all/>
     </privilege>
   </ace>
 </acl>
```

The contents of the file using gedit:



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

2. By using `DBMS_XDB_REPOS`, get the ACL document for '`/home/HR/xml/bookstore.xml`'. Use the SQL Worksheet for the `hr` schema.

   Answer:

   Run `sol_14_02_02.sql`.

```
SELECT
DBMS_XDB_REPOS.getACLDocument('/home/HR/xml/bookstore.xml') from
dual;
```

```
SQL> set long 20000
SQL> /

DBMS_XDB_REPOS.GETACLDOCUMENT('/HOME/HR/XML/BOOKSTORE.XML')

<acl description="Protected:Readable by PUBLIC and all privileges to OWNER" xmln
s="http://xmlns.oracle.com/xdb/acl.xsd" xmlns:dav="DAV:" xmlns:xsi="http://www.w
3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.oracle.com/xdb/a
cl.xsd                          http://xmlns.oracle.com/xdb/acl.xsd" shared="tr
ue">
  <ace>
    <grant>true</grant>
    <principal>dav:owner</principal>
    <privilege>
      <all/>
    </privilege>

DBMS_XDB_REPOS.GETACLDOCUMENT('/HOME/HR/XML/BOOKSTORE.XML')
--------------------------------------------------------------------------------
  </ace>
  <ace>
    <grant>true</grant>
    <principal>XDBADMIN</principal>
    <privilege>
      <all/>
    </privilege>
  </ace>
  <ace>
    <grant>true</grant>
    <principal>PUBLIC</principal>

DBMS_XDB_REPOS.GETACLDOCUMENT('/HOME/HR/XML/BOOKSTORE.XML')
--------------------------------------------------------------------------------
    <privilege>
      <read-properties/>
      <read-contents/>
      <read-acl/>
      <resolve/>
    </privilege>
  </ace>
</acl>


SQL>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. By using `DBMS_XDB`, set the ACL document for '`/home/HR/xml/bookstore.xml`' to `all_owner_acl.xml`. Display the changed ACL. Use the SQL Worksheet for the `hr` schema.

   Answer:

   Run `sol_14_02_03.sql`.

```
CALL
DBMS_XDB_REPOS.setACL('/home/HR/xml/bookstore.xml','/sys/acls/all_
owner_acl.xml');
SELECT DBMS_XDB.getACLDocument(
  '/home/HR/xml/bookstore.xml')
FROM dual;
```

```
SQL> set long 20000
SQL> CALL DBMS_XDB_REPOS.setACL('/home/HR/XML/bookstore.xml','/sys/acls/all_owner_acl.xml');
SELECT DBMS_XDB.getACLDocument(
  '/home/HR/XML/bookstore.xml')
FROM dual;

Call completed.

SQL>   2    3
DBMS_XDB.GETACLDOCUMENT('/HOME/HR/XML/BOOKSTORE.XML')

<acl description="Private:All privileges to OWNER only and not accessible to oth
ers" xmlns="http://xmlns.oracle.com/xdb/acl.xsd" xmlns:dav="DAV:" xmlns:xsi="htt
p://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://xmlns.oracle.
com/xdb/acl.xsd                          http://xmlns.oracle.com/xdb/acl.xsd" s
hared="true">
  <ace>
    <grant>true</grant>
    <principal>dav:owner</principal>
    <privilege>
      <all/>
    </privilege>

DBMS_XDB.GETACLDOCUMENT('/HOME/HR/XML/BOOKSTORE.XML')
--------------------------------------------------------------------------------
  </ace>
</acl>


SQL>
```

4. You can store only XML data in Oracle XML DB Repository. True/**False**
5. Which of the following statements are true about `XDB$RESOURCE`?
   a. It is a PL/SQL package used for accessing the repository resources.
   b. It is an `XMLType` table that stores a copy of the repository resources.
   c. **It is an `XMLType` table used to store the repository resources.**
   d. **It conforms to the `xdbresource.xsd` schema.**
6. Consider the following scenarios and identify where you would use `XInclude`, `XLink`, both, or none.
   a. In your database, you have separate documents that store information about your company's departments, products, and clients. You want to create an XML document that contains all this information. **`XInclude`**

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

b. You need to create an Advanced SQL book. Two of the chapters that you must include in this book are present in other books related to SQL. You want to include those chapters as is in your book. Also, at the end of your book, you want to give links to those books as references. **`XInclude`** and **`XLink`**

7. You want to create an XML document that contains information about a specific employee. The employee's details are available in three separate documents along with the details of other employees. Will you be able to create this document by using Oracle XML DB's support for `XInclude`? **No. Oracle XML DB does not support the `XPointer` attribute of `XInclude`. You can combine only complete documents into a single document and not parts of different documents.**
8. You can attach **repository** events to individual resources, as well as the entire repository. **True**/False.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 15: Using Native Oracle XML DB Web Services

Chapter 15

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

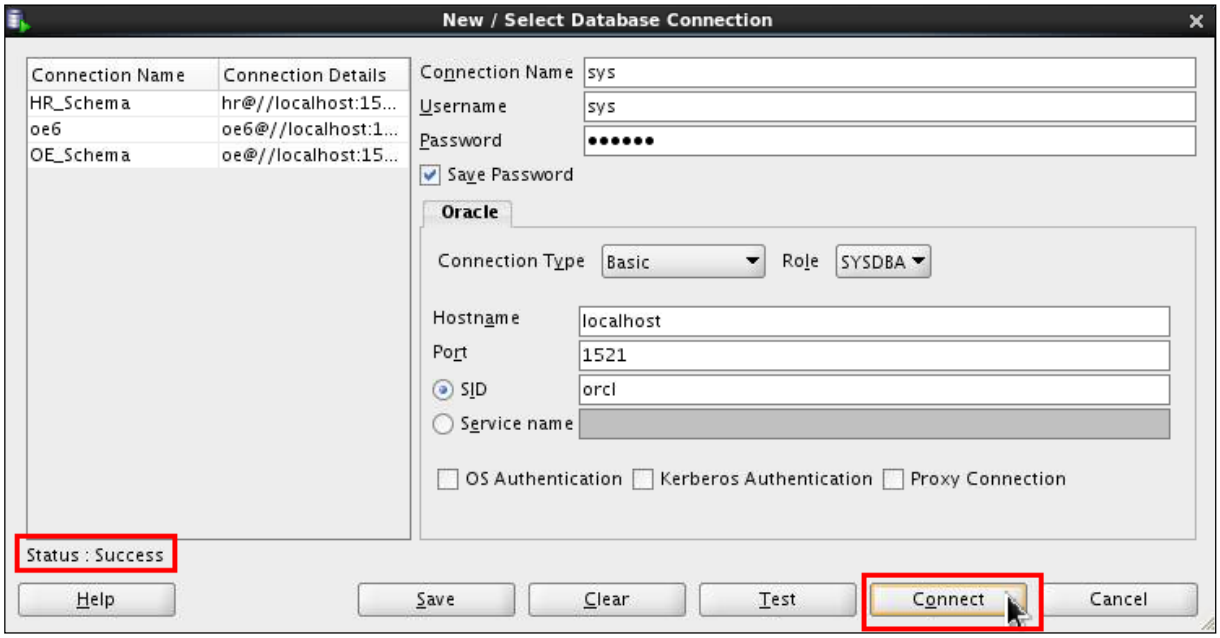
Oracle University and Error : You are not a Valid Partner use only

Practice 15-1: Using Oracle XML DB Web Service

In this practice, you execute a PL/SQL function using native Oracle XML DB web service.

1. Create a new database connection by using the information in the following table. Test the new connection. If the Status is Success, connect to the database by using the new `sys` database connection.

Connection Name	`sys`
Username	`sys`
Password	`oracle`
Save Password check box	`Enabled (selected)`
Role	`SYSDBA`
Hostname	`localhost`
Port	`1521`
SID	`orcl`



2. Open and run the `lab_15_01_02.sql` script file to configure a web service `orawsv`.

```
connect sys/oracle as sysdba
create or replace procedure addServletMapping (pattern varchar2,
                                               servletname varchar2,
                                               dispname varchar2,
                                               servletclass varchar2,
                                               servletschema varchar2,
                                               language varchar2,
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
                              description varchar2,
                              securityRole xmltype) as
  xdbconfig xmltype;
begin
   xdbconfig := dbms_xdb.cfg_get();
   select deleteXML
          (
            xdbconfig,

'/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/ser
vletconfig/servlet-list/servlet[servlet-name="' || servletname
|| '"]'
          )
          into xdbconfig
          from dual;

   if (language = 'C') then
     select insertChildXML
            (
              xdbconfig,

'/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/ser
vletconfig/servlet-list',
              'servlet',
              xmlElement
              (
                "servlet",

xmlAttributes('http://xmlns.oracle.com/xdb/xdbconfig.xsd' as
"xmlns"),
                xmlForest
                (
                   servletname as "servlet-name",
                   language as "servlet-language",
                   dispname as "display-name",
                   description as "description"
                ),
                securityRole
              )
            )
       into xdbconfig
       from dual;
   else
     select insertChildXML
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
            (
              xdbconfig,

'/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/ser
vletconfig/servlet-list',
              'servlet',
              xmlElement
              (
                "servlet",

xmlAttributes('http://xmlns.oracle.com/xdb/xdbconfig.xsd' as
"xmlns"),
                xmlForest
                (
                   servletname as "servlet-name",
                   language as "servlet-language",
                   dispname as "display-name",
                   description as "description",
                   servletclass as "servlet-class",
                   servletschema as "servlet-schema"
                )
              )
            )
       into xdbconfig
       from dual;
   end if;

   select deleteXML
          (
            xdbconfig,

'/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/ser
vletconfig/servlet-mappings/servlet-mapping[servlet-name="' ||
servletname || '"]'
          )
          into xdbconfig
          from dual;

   select insertChildXML
          (
            xdbconfig,

'/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/ser
vletconfig/servlet-mappings',
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
            'servlet-mapping',
            xmltype
            (
               '<servlet-mapping
xmlns="http://xmlns.oracle.com/xdb/xdbconfig.xsd">
                  <servlet-pattern>'||pattern||'</servlet-
pattern>
                  <servlet-name>'||servletname||'</servlet-name>
                </servlet-mapping>'
            )
          )
     into xdbconfig
     from dual;

  dbms_xdb.cfg_update(xdbconfig);

end;
/

call addServletMapping(
       '/orawsv/*',
       'orawsv',
       'Oracle Query Web Service',
       null,
       null,
       'C',
       'Web Services Servlet',
       xmltype(
         '<security-role-ref>
            <role-name>XDB_WEBSERVICES</role-name>
            <role-link>XDB_WEBSERVICES</role-link>
          </security-role-ref>'
       )
)
/

call addServletMapping(
       '/orawsdl/*',
       'orawsdl',
       'Oracle WSDLs',
       null,
       null,
       'C',
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
        'WSDL Servlet',
        xmltype(
          '<security-role-ref>
             <role-name>XDB_WEBSERVICES</role-name>
             <role-link>XDB_WEBSERVICES</role-link>
           </security-role-ref>'
        )
)
/
```

```
SQL*Plus: Release 12.1.0.1.0 Production on Thu Jul 25 12:57:33 2013

Copyright (c) 1982, 2013, Oracle.  All rights reserved.


Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing opt
ions

SQL>
```

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
SQL> @lab_15_01_02
SQL> connect sys/oracle as sysdba
Connected.
SQL>
SQL> create or replace procedure addServletMapping (pattern varchar2,
  2                                 servletname varchar2,
  3                                 dispname varchar2,
  4                                 servletclass varchar2,
  5                                 servletschema varchar2,
  6                                 language varchar2,
  7                                 description varchar2,
  8                                 securityRole xmltype) as
  9    xdbconfig xmltype;
 10  begin
 11     xdbconfig := dbms_xdb.cfg_get();
 12     select deleteXML
 13            (
 14              xdbconfig,
 15              '/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/se
rvletconfig/servlet-list/servlet[servlet-name="' || servletname || '"]'
 16            )
 17            into xdbconfig
 18            from dual;
 19
 20     if (language = 'C') then
 21       select insertChildXML
 22              (
 23                xdbconfig,
 24                '/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/
servletconfig/servlet-list',
 25                'servlet',
 26                xmlElement
 27                (
 28                  "servlet",
 29                  xmlAttributes('http://xmlns.oracle.com/xdb/xdbconfig.xsd' a
s "xmlns"),
 30                  xmlForest
 31                  (
 32                     servletname as "servlet-name",
 33                     language as "servlet-language",
 34                     dispname as "display-name",
 35                     description as "description"
 36                  ),
 37                  securityRole
 38                )
 39              )
 40         into xdbconfig
 41         from dual;
 42     else
 43       select insertChildXML
 44              (
 45                xdbconfig,
 46                '/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/
servletconfig/servlet-list',
 47                'servlet',
 48                xmlElement
 49                (
 50                  "servlet",
 51                  xmlAttributes('http://xmlns.oracle.com/xdb/xdbconfig.xsd' a
s "xmlns"),
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
 48                xmlElement
 49                (
 50                  "servlet",
 51                  xmlAttributes('http://xmlns.oracle.com/xdb/xdbconfig.xsd' a
s "xmlns"),
 52                  xmlForest
 53                  (
 54                     servletname as "servlet-name",
 55                     language as "servlet-language",
 56                     dispname as "display-name",
 57                     description as "description",
 58                     servletclass as "servlet-class",
 59                     servletschema as "servlet-schema"
 60                  )
 61                )
 62              )
 63         into xdbconfig
 64         from dual;
 65     end if;
 66
 67     select deleteXML
 68            (
 69              xdbconfig,
 70              '/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/se
rvletconfig/servlet-mappings/servlet-mapping[servlet-name="' || servletname || '
"]'
 71            )
 72            into xdbconfig
 73            from dual;
 74
 75     select insertChildXML
 76            (
 77              xdbconfig,
 78              '/xdbconfig/sysconfig/protocolconfig/httpconfig/webappconfig/se
rvletconfig/servlet-mappings',
 79              'servlet-mapping',
 80              xmltype
 81              (
 82                 '<servlet-mapping xmlns="http://xmlns.oracle.com/xdb/xdbconf
ig.xsd">
 83                    <servlet-pattern>'||pattern||'</servlet-pattern>
 84                    <servlet-name>'||servletname||'</servlet-name>
 85                  </servlet-mapping>'
 86              )
 87            )
 88       into xdbconfig
 89       from dual;
 90
 91    dbms_xdb.cfg_update(xdbconfig);
 92
 93  end;
 94  /

Procedure created.

SQL>
SQL> call addServletMapping(
  2         '/orawsv/*',
  3         'orawsv',
  4         'Oracle Query Web Service',
  5         null,
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
SQL> call addServletMapping(
  2         '/orawsv/*',
  3         'orawsv',
  4         'Oracle Query Web Service',
  5         null,
  6         null,
  7         'C',
  8         'Web Services Servlet',
  9         xmltype(
 10           '<security-role-ref>
 11              <role-name>XDB_WEBSERVICES</role-name>
 12              <role-link>XDB_WEBSERVICES</role-link>
 13            </security-role-ref>'
 14         )
 15  )
 16  /

Call completed.

SQL>
SQL> call addServletMapping(
  2         '/orawsdl/*',
  3         'orawsdl',
  4         'Oracle WSDLs',
  5         null,
  6         null,
  7         'C',
  8         'WSDL Servlet',
  9         xmltype(
 10           '<security-role-ref>
 11              <role-name>XDB_WEBSERVICES</role-name>
 12              <role-link>XDB_WEBSERVICES</role-link>
 13            </security-role-ref>'
 14         )
 15  )
 16  /

Call completed.

SQL>
SQL> grant XDB_WEBSERVICES to oe
  2  /

Grant succeeded.

SQL>
SQL> -- For 11g only
SQL> grant XDB_WEBSERVICES_OVER_HTTP to oe
  2  /

Grant succeeded.

SQL> grant XDB_WEBSERVICES_WITH_PUBLIC to oe
  2  /

Grant succeeded.

SQL>
SQL> -- Clean up afterward
SQL> drop procedure addServletMapping
  2  /
```

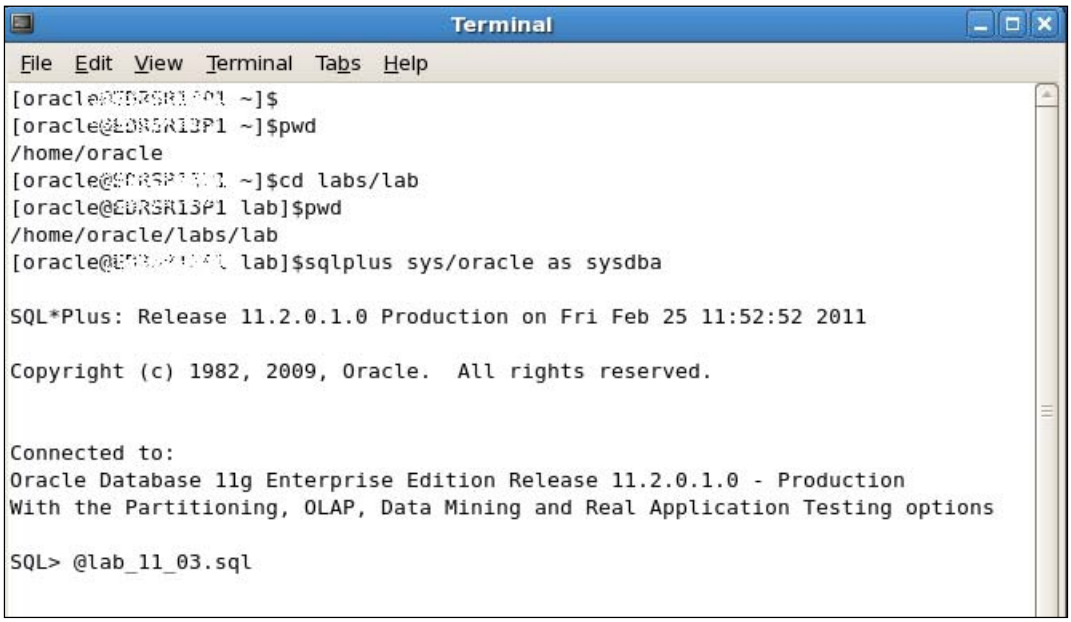
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. To verify successful configuration of the web service, execute `lab_15_01_03.sql` in SQL*Plus.

   Start a terminal window, change to the `/home/oracle/labs/lab` folder, start sqlplus as `sys/oracle as sysdba`, and then run the `lab_15_01_03.sql` script using the `@` command as shown in the following screenshot:




```
CONNECT sys/oracle AS SYSDBA

XQUERY declare default element namespace
"http://xmlns.oracle.com/xdb/xdbconfig.xsd";
for $doc in
fn:doc("/xdbconfig.xml")/xdbconfig/sysconfig/protocolconfig/httpcon
fig/webappconfig/servletconfig/servlet-list/servlet[servlet-
name='orawsv']
return $doc
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Output:

```
SQL> @lab_15_01_03
Connected.

Result Sequence
-----------------------------------------------------------------------------------------
<servlet xmlns="http://xmlns.oracle.com/xdb/xdbconfig.xsd">
  <servlet-name>orawsv</servlet-name>
  <servlet-language>C</servlet-language>
  <display-name>Oracle Query Web Service</display-name>
  <description>Web Services Servlet</description>
  <security-role-ref>
    <role-name>XDB_WEBSERVICES</role-name>
    <role-link>XDB_WEBSERVICES</role-link>
  </security-role-ref>
</servlet>


SQL>
```

4. Enable the web service `orawsv` for `OE`. Log on as the `SYS` user and grant the roles `XDB_WEBSERVICES`, `XDB_WEBSERVICES_OVER_HTTP`, and `XDB_WEBSERVICES_WITH_PUBLIC` to `OE`. Execute `lab_15_01_04.sql`.

```
CONNECT sys/oracle AS SYSDBA
GRANT XDB_WEBSERVICES TO OE;
GRANT XDB_WEBSERVICES_OVER_HTTP TO OE;
GRANT XDB_WEBSERVICES_WITH_PUBLIC TO OE;
```

```
SQL> @lab_15_01_04
Connected.

Grant succeeded.


Grant succeeded.


Grant succeeded.

SQL>
```

5. View the web service `orawsv` in a browser.

   Steps:

   a. Open Mozilla Firefox by using the icon on your desktop.

   b. Enter the following URL to view the servlet WSDL.

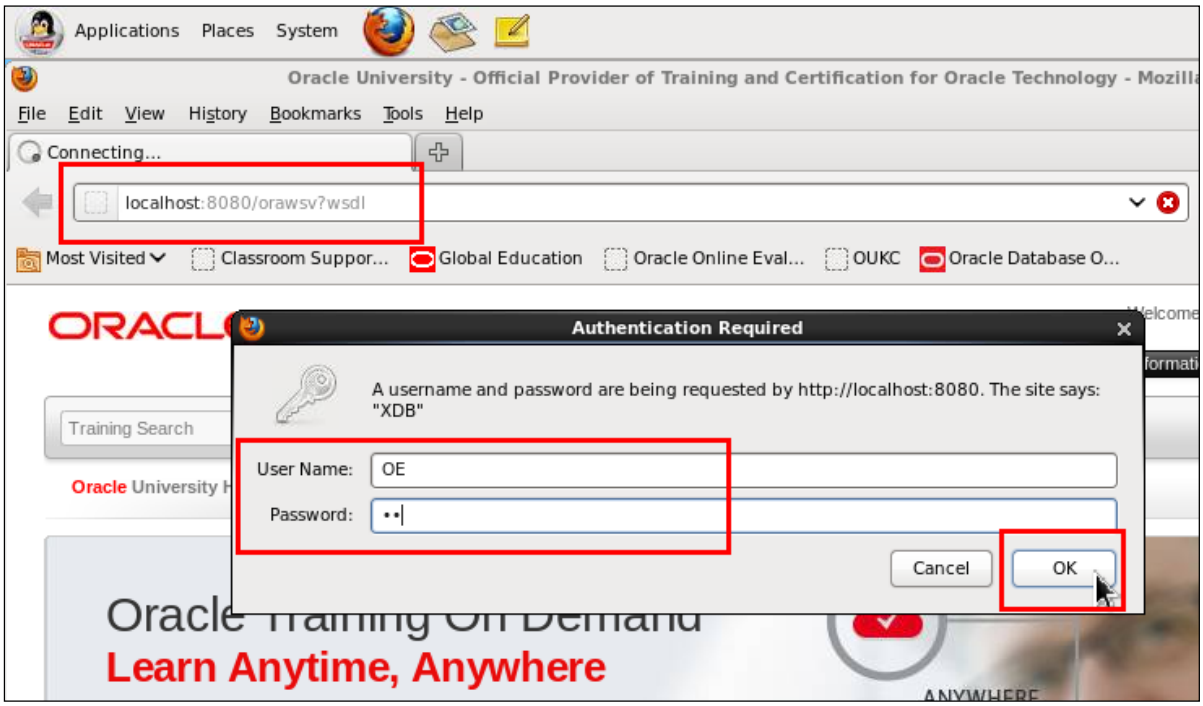
```
http://localhost:8080/orawsv?wsdl
```

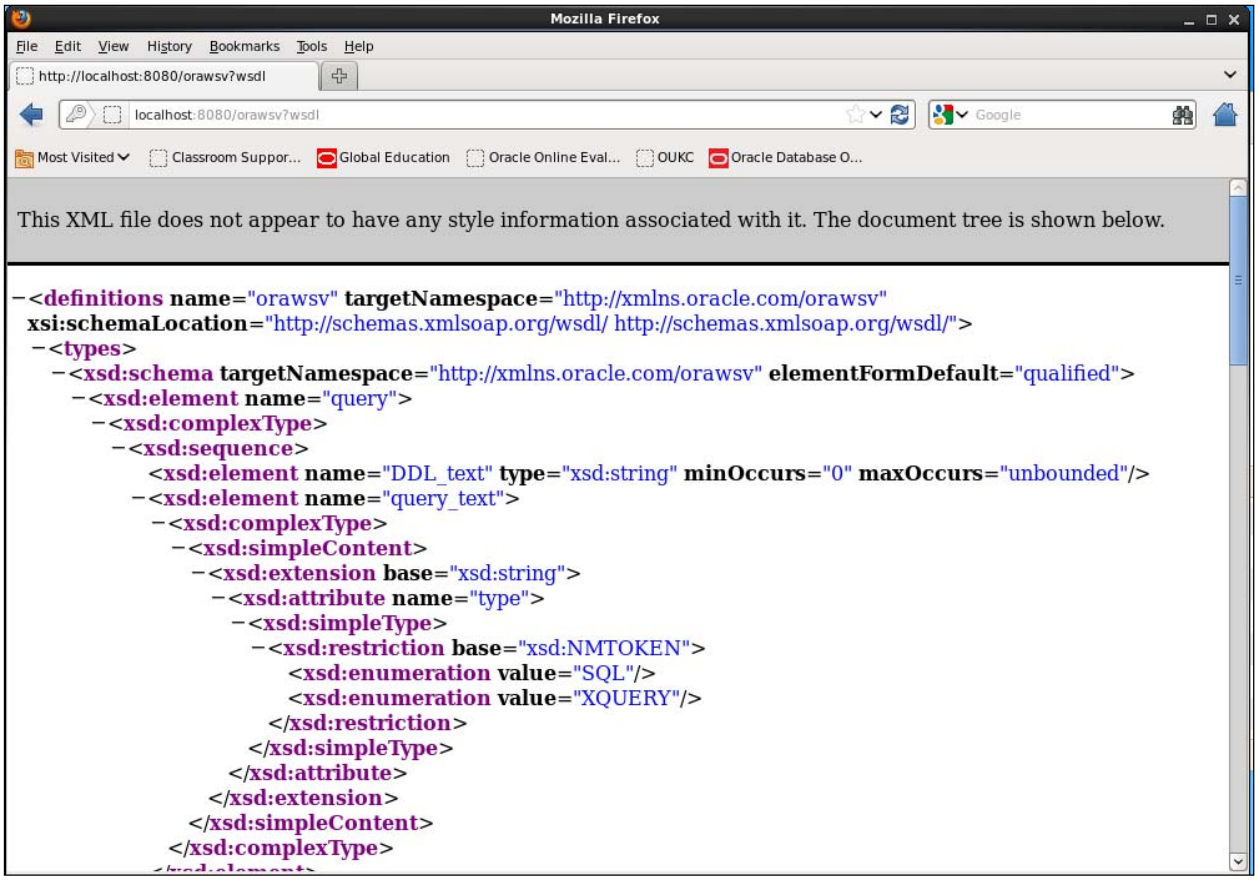
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

The Authentication Required window is displayed. Enter `OE` (uppercase) for the User Name and `oe` (lowercase) for the Password, and then click OK.





. . .

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

6. Open a SQL Worksheet for the `OE` schema. Run the `lab_15_01_06.sql` file to create the `test_tab` table and insert values into it. Run `lab_15_01_06.sql`.

```
Drop TABLE test_tab;
CREATE TABLE test_tab (
  id          NUMBER,
  description VARCHAR2(50),
  CONSTRAINT test_tab_pk PRIMARY KEY (id)
);

INSERT INTO test_tab (id, description) VALUES (1, 'ONE');
INSERT INTO test_tab (id, description) VALUES (2, 'TWO');
INSERT INTO test_tab (id, description) VALUES (3, 'THREE');
COMMIT;

CREATE OR REPLACE PROCEDURE get_description (
  p_id          IN  test_tab.id%TYPE,
  p_description OUT test_tab.description%TYPE) AS
BEGIN
  SELECT description
  INTO   p_description
  FROM   test_tab
  WHERE  id = p_id;
END;
/
```

Script Output

Task completed in 0.333 seconds

```
Error starting at line 2 in command:
Drop TABLE test_tab
Error report:
SQL Error: ORA-00942: table or view does not exist
00942. 00000 -  "table or view does not exist"
*Cause:
*Action:
table TEST_TAB created.
1 rows inserted.
1 rows inserted.
1 rows inserted.
committed.
PROCEDURE GET_DESCRIPTION compiled
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

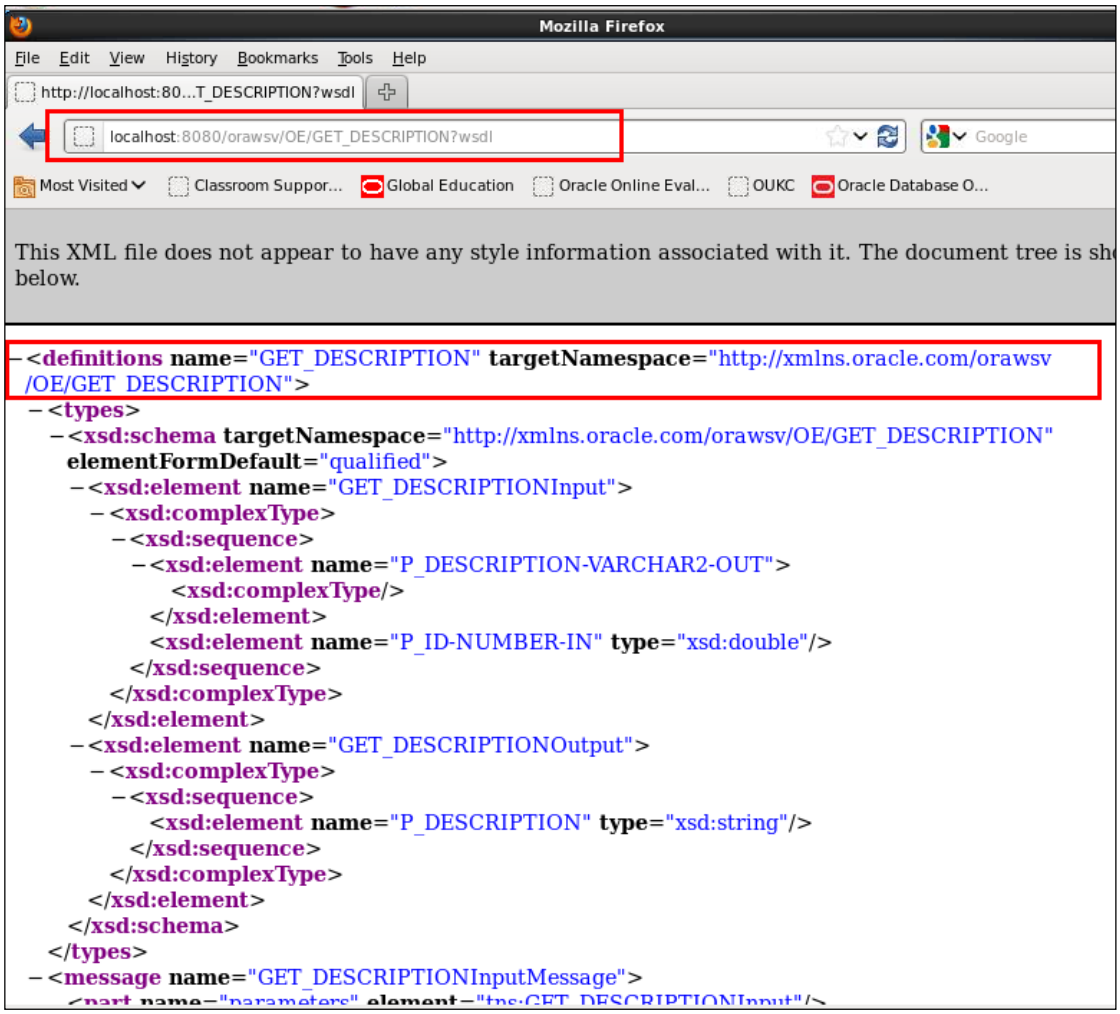
7. View the function's WSDL in a browser.

Steps:

a. Open Mozilla Firefox by using the icon on your desktop.

b. Enter the following URL to view the servlet WSDL.

```
http://localhost:8080/orawsv/OE/GET_DESCRIPTION?wsdl
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 16: Exporting and Importing XML Data

Chapter 16

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

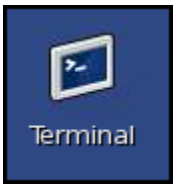
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practice 16-1: Exporting and Importing XML Data

In this practice, you load the XML data that is stored in a control file and an external file into an `XMLType` table named `prac_load` by using the Oracle utility `SQLLDR`. You also export and import an `XMLType` table from the `oe` schema to the `hr` schema.

Note: This practice requires you to use the command prompt when using the `SQLLDR` utility. You should access the terminal window by using the icon on your desktop.



Loading XML Data Using `SQL*Loader`

1. Open and run the `lab_16_01_01.sql` script file from the `hr` schema to create an `XMLType` table named `prac_load`.

```
DROP TABLE prac_load;

CREATE TABLE prac_load of XMLType;
```

```
SQL> @lab_16_01_01
DROP TABLE prac_load
           *
ERROR at line 1:
ORA-00942: table or view does not exist



Table created.

SQL>
```

2. Create the `lab_16_01_02.ctl` control file and load the following XML data into `prac_load` by using the `SQLLDR` utility. Store the logging information about the loading process in `mylog.log`. Store records that cause errors during insert or that are improperly formatted into `mybad.bad`. The data is also stored in a text file named `lab_16_01_02.txt` in the `/home/oracle/labs/lab` folder as follows:

```
<EMP><name>KING</name><SAL>8000</SAL></EMP>
<EMP><name>SCOTT</name><SAL>2000</SAL></EMP>
<EMP><name>SMITH</name><SAL>7000</SAL></EMP>
<EMP><name>CLARK</name><SAL>1000</SAL></EMP>
<EMP><name>TURNER</name><SAL>2000</SAL></EMP>
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

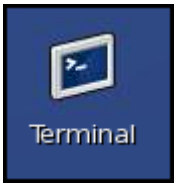
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Steps:

a. Create the `lab_16_01_02.ctl` control file as follows and save it in the `xml_dir` folder.

```
LOAD DATA
INFILE *
INTO TABLE prac_load
XMLType(xmldata)
FIELDS TERMINATED BY ',' OPTIONALLY ENCLOSED BY '"'
(
xmldata
)
BEGINDATA
<EMP><name>KING</name><SAL>8000</SAL></EMP>,
<EMP><name>SCOTT</name><SAL>2000</SAL></EMP>,
<EMP><name>SMITH</name><SAL>7000</SAL></EMP>,
<EMP><name>CLARK</name><SAL>1000</SAL></EMP>,
<EMP><name>TURNER</name><SAL>2000</SAL></EMP>
```

b. Open a terminal window by using the icon on your desktop.



c. Enter the following command at the `$` command prompt by using a terminal window. When prompted for the control file name, enter `lab_16_01_02.ctl`.

```
$ cd /home/oracle/labs/xml_dir
$ sqlldr hr/hr
$ control= lab_16_01_02.ctl
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. Query the `prac_load` table to verify the insert.

```
select * from prac_load;
```

```
SYS_NC_ROWINFO$
-------------------------------------------------------
<EMP>
  <name>KING</name>
  <SAL>8000</SAL>
</EMP>

<EMP>
  <name>SCOTT</name>
  <SAL>2000</SAL>
</EMP>

<EMP>
  <name>SMITH</name>
  <SAL>7000</SAL>
</EMP>

<EMP>
  <name>CLARK</name>
  <SAL>1000</SAL>
</EMP>

<EMP>
  <name>TURNER</name>
  <SAL>2000</SAL>
</EMP>


SQL>
```

4. Create the `lab_16_01_04.ctl` control file and load the XML data that is stored in the `/home/oracle/labs/xml_dir/james.xml` external file into `prac_load` by using the `SQLLDR` utility. Query the `prac_load` table to verify the successful insert of the employee James record.

   Steps:

   a. Create the `lab_16_01_04.ctl` control file as follows and save it in the `xml_dir` folder.

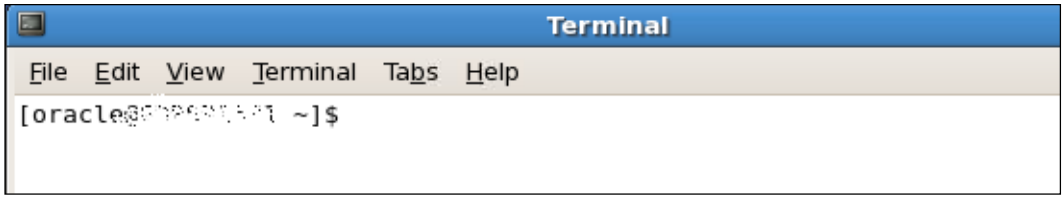
```
LOAD DATA
INFILE *
APPEND INTO TABLE prac_load
XMLType(xmldata)
(
lobfn FILLER CHAR TERMINATED BY ',',
XMLDATA LOBFILE(lobfn) TERMINATED BY EOF
)
BEGINDATA
james.xml
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

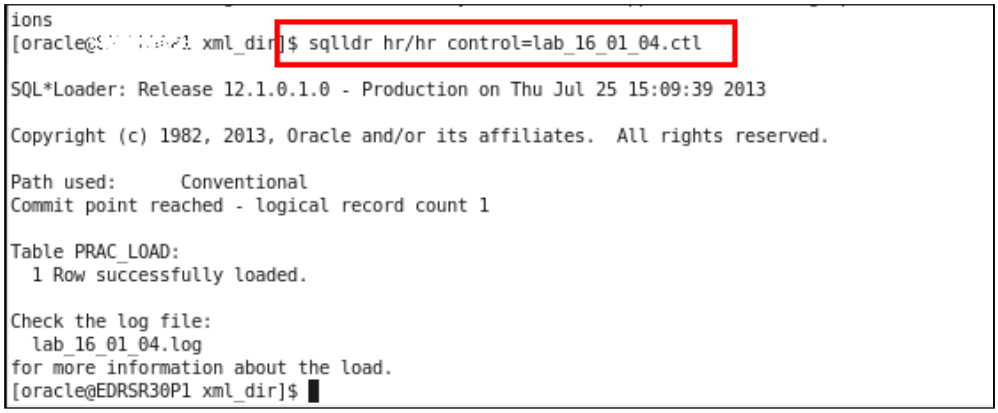
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

b. Open the command prompt by using the icon on your desktop.



c. Enter the following command at the command prompt on separate lines as shown in this code example, or you can enter the entire command at the $ prompt as shown in the following screenshot.

```
cd /home/oracle/labs/xml_dir
sqlldr hr/hr
control= lab_16_01_04.ctl
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

d. Query the `prac_load` table to verify the insert operation.

```
select * from prac_load;
```

```
SYS_NC_ROWINFO$
--------------------------------------------------------------------
<EMP>
  <name>KING</name>
  <SAL>8000</SAL>
</EMP>

<EMP>
  <name>SCOTT</name>
  <SAL>2000</SAL>
</EMP>

<EMP>

SYS_NC_ROWINFO$
--------------------------------------------------------------------
  <name>SMITH</name>
  <SAL>7000</SAL>
</EMP>

<EMP>
  <name>CLARK</name>
  <SAL>1000</SAL>
</EMP>

<EMP>
  <name>TURNER</name>

SYS_NC_ROWINFO$
--------------------------------------------------------------------
  <SAL>2000</SAL>
</EMP>

<EMP>
  <name>JAMES</name>
  <SAL>8000</SAL>
</EMP>

6 rows selected.

SQL>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Exporting and Importing an `XMLType` Table by Using Oracle Data Pump

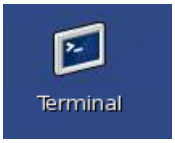
5. Open a SQL Worksheet for the `oe` schema and run the `lab_16_01_05.sql` file to create an `XMLType` table `prac_exp` and insert values into it.

```
table PRAC_EXP created.
1 rows inserted.
1 rows inserted.
1 rows inserted.
committed.
SYS_NC_ROWINFO$
--------------------------------------------------------------------
<jobs>
  <job_id>AD_PRES</job_id>
  <job_title>President</job_title>
  <min_salary>20000</min_salary>
  <max_salary>40000</max_salary>
</jobs>


<jobs>
  <job_id>FI_MGR</job_id>
  <job_title>Finance Manager</job_title>
  <min_salary>8200</min_salary>
  <max_salary>16000</max_salary>
</jobs>


<jobs>
  <job_id>AC_MGR</job_id>
  <job_title>Accounting Manager</job_title>
  <min_salary>8200</min_salary>
  <max_salary>16000</max_salary>
</jobs>
```

6. Export the `XMLType` table `prac_exp` to a dump file named `lab_16_06_exp.dmp` and store it in the `xml_dir` directory.

   Steps:

   a. Open a terminal window by using the icon on your desktop.



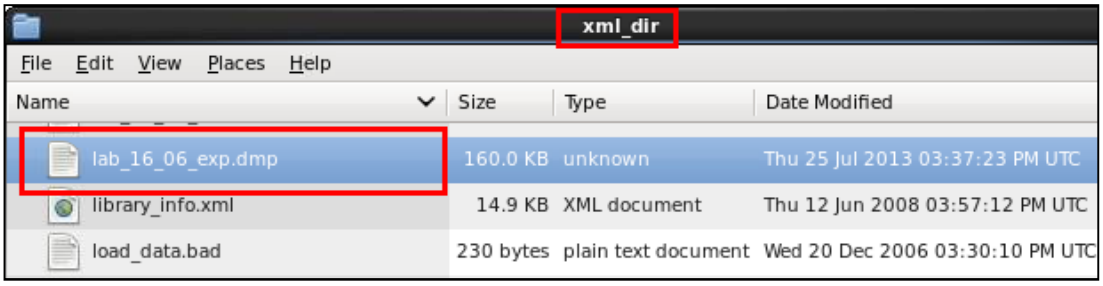
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

b. Enter the following command at the command prompt:

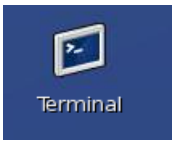
```
[oracle@EDRSR30P1 lab]$ expdp oe/oe tables=prac_exp dumpfile=xml_dir:lab_16_06_exp.dmp nologfile=y;

Export: Release 12.1.0.1.0 - Production on Thu Jul 25 15:36:32 2013

Copyright (c) 1982, 2013, Oracle and/or its affiliates.  All rights reserved.

Connected to: Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
Starting "OE"."SYS_EXPORT_TABLE_01":  oe/******** tables=prac_exp dumpfile=xml_dir:lab_16_06_exp.dmp nologfile=y
Estimate in progress using BLOCKS method...
Processing object type TABLE_EXPORT/TABLE/TABLE_DATA
Total estimation using BLOCKS method: 192 KB
Processing object type TABLE_EXPORT/TABLE/TABLE
Processing object type TABLE_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS
Processing object type TABLE_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS
Processing object type TABLE_EXPORT/TABLE/STATISTICS/MARKER
. . exported "OE"."PRAC_EXP"                              6.328 KB       3 rows
Master table "OE"."SYS_EXPORT_TABLE_01" successfully loaded/unloaded
******************************************************************************
Dump file set for OE.SYS_EXPORT_TABLE_01 is:
  /home/oracle/labs/xml_dir/lab_16_06_exp.dmp
Job "OE"."SYS_EXPORT_TABLE_01" successfully completed at Thu Jul 25 15:37:23 2013 elapsed 0 00:00:49

[oracle@EDRSR30P1 lab]$
```



7. Import the `lab_16_01_07_tryimp.dmp` file, stored in the `xml_dir` directory, to the `HR` schema.

Steps:

a. Open a terminal window by using the icon on your desktop.



b. Enter the following command at the command prompt:

```
[oracle@EDRSR30P1 lab]$ impdp hr/hr directory=xml_dir dumpfile=lab_16_01_07_tryimp.dmp nologfile=y;

Import: Release 12.1.0.1.0 - Production on Thu Jul 25 15:58:40 2013

Copyright (c) 1982, 2013, Oracle and/or its affiliates.  All rights reserved.

Connected to: Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 - 64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real Application Testing options
Master table "HR"."SYS_IMPORT_FULL_01" successfully loaded/unloaded
Starting "HR"."SYS_IMPORT_FULL_01":  hr/******** directory=xml_dir dumpfile=lab_16_01_07_tryimp.dmp nologfile=y

Processing object type TABLE_EXPORT/TABLE/TABLE
Processing object type TABLE_EXPORT/TABLE/TABLE_DATA
. . imported "HR"."PRAC_EXP"                              6.554 KB       3 rows
Job "HR"."SYS_IMPORT_FULL_01" successfully completed at Thu Jul 25 15:59:02 2013 elapsed 0 00:00:21

[oracle@EDRSR30P1 lab]$
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

c. Query the `prac_exp` that you imported into the `hr` schema as follows:

```
SET LONG 5000

SELECT *
FROM prac_exp;
```

```
SYS_NC_ROWINFO$
--------------------------------------------------------------------------------
<jobs>
  <job_id>AD_PRES</job_id>
  <job_title>President</job_title>
  <min_salary>20000</min_salary>
  <max_salary>40000</max_salary>
</jobs>

<jobs>
  <job_id>FI_MGR</job_id>
  <job_title>Finance Manager</job_title>
  <min_salary>8200</min_salary>

SYS_NC_ROWINFO$
--------------------------------------------------------------------------------
  <max_salary>16000</max_salary>
</jobs>

<jobs>
  <job_id>AC_MGR</job_id>
  <job_title>Accounting Manager</job_title>
  <min_salary>8200</min_salary>
  <max_salary>16000</max_salary>
</jobs>


SQL>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 17: Workshop

Chapter 17

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

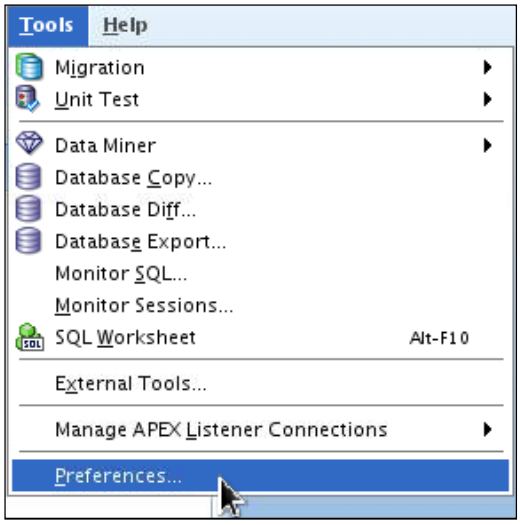
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

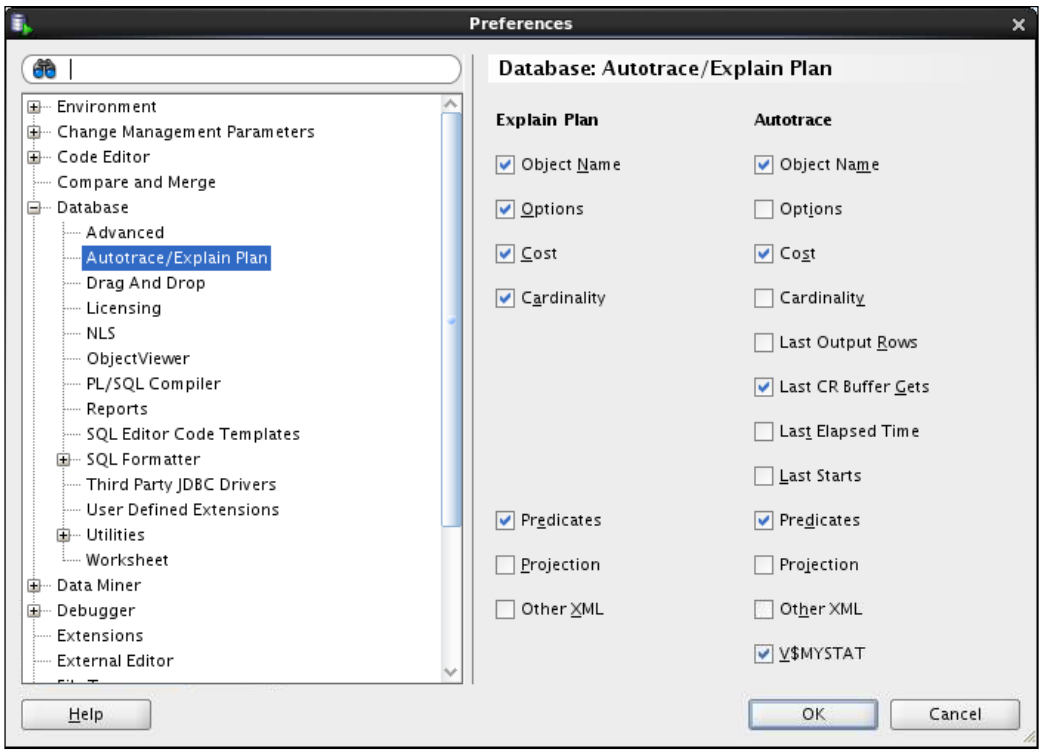
Practice 17-1: Workshop Workshops

In this practice, you use SQL Developer to perform all the exercises. For this workshop, you must be connected to the database as `hr` with the password `hr`.

1. Verify SQL Developer preferences.
   a. Select Tools > Preferences.



   b. Expand the **Database** node, and then select the **Autotrace/Explain Plan** Parameter. Ensure that you select the following check boxes, and then click **OK**.
      1) Object_Name
      2) Cost
      3) Cardinality



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Loading Library Data into Binary XML Storage

2. Create a temporary table by using the binary XML storage to load the library data. The table must contain the `citations` column. The `citations` column must be of `XMLType`. Name the table `library_info_tmp_tab`.

```
CREATE TABLE library_info_tmp_tab (citations XMLType)
XMLType COLUMN citations store as SECUREFILE binary XML;
```

```
Script Output
Task completed in 0.369 seconds
Error starting at line 3 in command:
drop table library_info_tmp_tab force
Error report:
SQL Error: ORA-00942: table or view does not exist
00942. 00000 -  "table or view does not exist"
*Cause:
*Action:
table LIBRARY_INFO_TMP_TAB created.
```

3. By using `lab_17_01_03_load_data1.sql`, insert library data from `library_info.xml`.

```
insert into library_info_tmp_tab
values(xmltype(bfilename('XML_DIR','library_info.xml'),
nls_charset_id('WE8MSWIN1252')));
```

```
Script Output
Task completed in 0.109 seconds
1 rows inserted
```

4. Create another table with an `XMLType` column by using the binary XML storage to store all the citations from the library data. The table must contain the `citation` column. The `citation` column must be of `XMLType`. Name the `library_info_tab`.

```
create table library_info_tab (citation xmltype)
xmltype column citation store as securefile binary xml;
```

```
Script Output
Task completed in 0.082 seconds
Error starting at line 3 in command:
drop table library_info_tab force
Error report:
SQL Error: ORA-00942: table or view does not exist
00942. 00000 -  "table or view does not exist"
*Cause:
*Action:
table LIBRARY_INFO_TAB created.
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

5. By using `lab_17_01_05_load_data2.sql`, insert the library data from the `library_info_tmp_tab` temporary table.

```
insert into library_info_tab select x.column_value from
library_info_tmp_tab, xmltable('//Library_InfoCitation' passing
citations) x;
/
commit
/
```

```
Script Output
Task completed in 0.107 seconds
5 rows inserted
```

Improving the Performance of XQuery Expressions Through Index Creation

6. Open `lab_17_01_06_cre_index1.sql`. Replace ***<TODO>*** with the appropriate code to create an `XMLIndex` index on the `citation` column of the `library_info_tab` table. The index can be called `citation_bix_ix`. Save the changes, and then run `lab_17_01_06_cre_index1.sql`.

```
drop index citation_bix_ix force
/
-- Create XML index on the binary XML table
create index citation_bix_ix on library_info_tab(citation)
indextype is xdb.xmlindex
parameters ('PATH TABLE citation_bix_path_table
PATH ID INDEX citation_bix_path_id_ix
ORDER KEY INDEX citation_bix_order_key_ix
VALUE INDEX     citation_bix_value_ix');
```

```
create index succeeded.
```

7. Create a secondary full-text index on the `VALUE` column of the path table. Open `lab_17_01_07_cre_index2.sql`. Replace <***TODO***> with the appropriate code, save the changes, and then run `lab_cre_index2.sql`.

```
drop index citation_bix_text_ix;
/
-- Create a secondary text index on the VALUE column of the path
table
CREATE INDEX citation_bix_text_ix ON citation_bix_path_table
(VALUE) INDEXTYPE IS CTXSYS.CONTEXT
parameters ('transactional');
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

```
Script Output
Task completed in 0.34 seconds
Error starting at line 2 in command:
drop index citation_bix_ix force
Error report:
SQL Error: ORA-01418: specified index does not exist
01418. 00000 -  "specified index does not exist"
*Cause:
*Action:
index CITATION_BIX_IX created.
```

8. By using `XMLExists`, write a query on the `library_info_tab` table to display the number of citations whose `DateCreated` year is greater than or equal to 2006. Use `lab_17_01_08_query1.sql` for help.

```
select count(*)
  from library_info_tab
where XMLExists('/Library_InfoCitation[DateCreated/Year
>="2006"]' passing citation)
/
```

```
Script Output
Task completed in 1.116 seconds
COUNT(*)
----------------------
2
```

9. By using `XMLTable` and `XMLExists`, write a query on the `library_info_tab` table to display the list of article titles for all those citations whose `DateCreated` year is greater than or equal to 2006. Use `lab_17_01_09_query2.sql` for help while replacing <***TODO***> with the appropriate code, save the changes, and then run the script.

Before editing:

```
-- Execute as hr/hr
select a.ARTICLE_TITLE
  from library_info_TAB,
       <TODO>
          columns
          ARTICLE_TITLE varchar2(4000) path 'ArticleTitle'
       ) a
 where <TODO> passing citation);
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

After editing:

```
select a.ARTICLE_TITLE
  from library_info_TAB,
       xmlTable
       (
          '/Library_InfoCitation/Article'
          passing citation
          columns
          ARTICLE_TITLE varchar2(4000) path 'ArticleTitle'
       ) a
where
XMLExists('/Library_InfoCitation[DateCreated/Year>="2006"]'passi
ng citation);
```

```
Script Output
Task completed in 0.07 seconds
ARTICLE_TITLE
--------------------------------------------------------------------------
Managing Risks.
Training for Librarianship.
```

10. In the SQL Worksheet, select the entire code from `lab_17_01_09_query2.sql` (or place the mouse cursor at the beginning of the `SELECT` statement), and then click the **Explain Plan** icon. Observe the Explain Plan output pane and note that the execution plan uses the `XMLIndex` index.

```
0.012 seconds
--Execute as HR
-- Q2
select a.ARTICLE_TITLE
  from library_info_TAB,
       xmlTable
       (
          '/Library_InfoCitation/Article'
          passing citation
          columns
          ARTICLE_TITLE varchar2(4000) path 'ArticleTitle'
       ) a
 where XMLExists('/Library_InfoCitation[DateCreated/Year>="2006"]'passing citation)
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Script Output | Explain Plan

SQL | 0.202 seconds

OPERATION	OBJECT_NAME	OPTIONS
SELECT STATEMENT		
FILTER		
Filter Predicates		
SYS_ORDERKEY_MAXCHILD(:B1)>:B2		
TABLE ACCESS	CITATION_BIX_PATH_TABLE	BY INDEX ROWID BATCHED
Filter Predicates		
AND		
SYS_P3.ORDER_KEY>:B1		
SYS_P3.ORDER_KEY<SYS_OF		
SYS_ORDERKEY_DEPTH(SYS_		
SYS_XMLI_LOC_ISNODE(SYS_		
INDEX	CITATION_BIX_PATH_ID_IX	RANGE SCAN
Access Predicates		
AND		
SYS_P3.PATHID=HEXTOF		
SYS_P3.RID=:B1		
NESTED LOOPS		
NESTED LOOPS		
NESTED LOOPS		
SORT		UNIQUE
TABLE ACCESS	CITATION_BIX_PATH_TABLE	BY INDEX ROWID BATCHED
Filter Predicates		
AND		
SYS_P8.VALUE>=		
SUBSTRB(VALUE,:		
SYS_XMLI_LOC_I:		
INDEX	CITATION_BIX_PATH_ID_IX	RANGE SCAN
Access Predicates		
SYS_P8.PATHID=		
TABLE ACCESS	LIBRARY_INFO_TAB	BY USER ROWID

OPERATION	OBJECT_NAME	OPTIONS
NESTED LOOPS		
NESTED LOOPS		
NESTED LOOPS		
SORT		UNIQUE
TABLE ACCESS	CITATION_BIX_PATH_TABLE	BY INDEX ROWID BATCHED
Filter Predicates		
AND		
SYS_P8.VALUE>=		
SUBSTRB(VALUE,:		
SYS_XMLI_LOC_I:		
INDEX	CITATION_BIX_PATH_ID_IX	RANGE SCAN
Access Predicates		
SYS_P8.PATHID=		
TABLE ACCESS	LIBRARY_INFO_TAB	BY USER ROWID
Access Predicates		
SYS_P8.PATHID=		
TABLE ACCESS	LIBRARY_INFO_TAB	BY USER ROWID
INDEX	CITATION_BIX_PATH_ID_IX	RANGE SCAN
Access Predicates		
AND		
INDEX	CITATION_BIX_PATH_ID_IX	RANGE SCAN
Access Predicates		
AND		
SYS_P1.PATHID=HEXTOF		
LIBRARY_INFO_TAB.ROW		
TABLE ACCESS	CITATION_BIX_PATH_TABLE	BY INDEX ROWID
LIBRARY_INFO_TAB.ROW		
TABLE ACCESS	CITATION_BIX_PATH_TABLE	BY INDEX ROWID
Filter Predicates		
SYS_XMLI_LOC_ISNODE(SYS_P1.		

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

11. By using `XMLExists`, write a query on the `library_info_tab` table to display the number of citations for which `ArticleTitle` contains the text "Training." Use `lab_17_01_11_query3.sql` for help.

```
select count(*)
  from library_info_TAB
where
XMLExists('/Library_InfoCitation/Article/ArticleTitle/text()[ora
:contains(.,"Training") > 0]' passing citation);
```

```
Script Output
Task completed in 0.311 seconds
COUNT(*)
---------------------
1
```

12. By using `XMLTable` and `XMLExists`, write a query on the `library_info_tab` table to display the list of article titles for all those citations where `ArticleTitle` contains the text "Training." Use `lab_17_01_12_query4.sql` for help.

```
select a.ARTICLE_TITLE
  from library_info_TAB,
       xmlTable
       (
          '/Library_InfoCitation/Article'
          passing citation
          columns
          ARTICLE_TITLE varchar2(4000) path 'ArticleTitle'
       ) a
where
XMLExists('/Library_InfoCitation/Article/ArticleTitle/text()[ora
:contains(.,"Training") > 0]' passing citation);
```

```
Script Output
Task completed in 0.132 seconds
ARTICLE_TITLE
--------------------------------------------------------------------------------
Training for Librarianship.
```

Oracle University and Error : You are not a Valid Partner use only

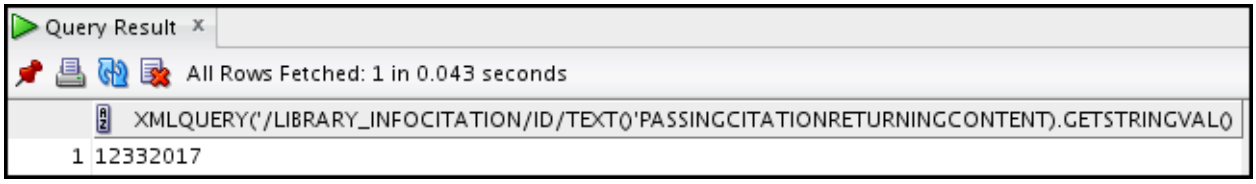
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Performing DML Operations

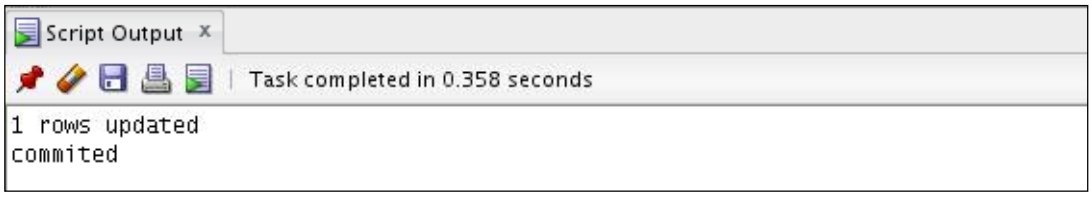
13. In the SQL Worksheet, open `lab_17_01_13_dml_query1.sql`. Click **Execute Statement**.

```
select XMLQuery('/Library_InfoCitation/ID/text()' passing
citation returning content).getStringVal()
from Library_Info_TAB
  where XMLExists('/Library_InfoCitation[ID="12332017"]' passing
citation);
```



14. Update the ID value `12332017` to `12332017_a`. Use `lab_17_01_14_dml.sql` for help. Save the changes, and then run `lab_17_01_14_dml.sql`.

```
--set autotrace traceonly
UPDATE Library_Info_TAB
  SET citation = updateXML(citation,
'/Library_InfoCitation/ID/text()[.="12332017"]', '12332017_a')
  where XMLExists('/Library_InfoCitation[ID="12332017"]' passing
citation);
/

commit
/
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

15. In the SQL Worksheet, select the entire code in `lab_17_01_14_dml.sql` and click the **Explain Plan** icon. Observe the Explain Plan output pane and note that the execution plan uses the `XMLIndex` index.

```
--Execute as HR
UPDATE Library_Info_TAB
  SET citation = updateXML(citation, '/Library_InfoCitation/ID/text()[.="12332017"]', '12332017_a')
  where XMLExists('/Library_InfoCitation[ID="12332017"]' passing citation);
/

commit
/
```

Explain Plan

SQL | 0.039 seconds

OPERATION	OBJECT_NAME	OPTIONS
UPDATE STATEMENT		
UPDATE	LIBRARY_INFO_TAB	
NESTED LOOPS		
SORT		UNIQUE
TABLE ACCESS	CITATION_BIX_PATH_TABLE	BY INDEX ROWID BATCHED
Filter Predicates		
AND		
SYS_P5.PATHID=HEX		
SYS_P5.VALUE='123		
SYS_XMLI_LOC_ISNO		
INDEX	CITATION_BIX_VALUE_IX	RANGE SCAN
Access Predicates		
SUBSTRB(VALUE,1,15		
TABLE ACCESS	LIBRARY_INFO_TAB	BY USER ROWID

16. Verify whether the ID value is updated. Run `lab_17_01_16_dml_query2.sql`.

Script Output

Task completed in 0.044 seconds

```
XMLQUERY('/LIBRARY_INFOCITATION/ID/TEXT()'PASSINGCITATIONRETURNINGCONTENT).GETSTRINGVAL()
--------------------------------------------------------------------------------
12332017_a
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 17-2: Workshop II

In this practice or task, you use SQL Developer to perform all the exercises. For this workshop, you must be connected to the database as `hr` with the password `hr`.

1. Run `lab_17_02_01_del_resources.sql`.

```
BEGIN
   IF
DBMS_XDB_REPOS.existsResource('/public/Tourist_Info/BangalorePal
ace.xml')
   THEN

DBMS_XDB_REPOS.deleteResource('/public/Tourist_Info/BangalorePal
ace.xml',DBMS_XDB_REPOS.DELETE_FORCE);
   END IF;


   IF
DBMS_XDB_REPOS.existsResource('/public/Tourist_Info/TechnologyMu
seum.xml')
   THEN

DBMS_XDB_REPOS.deleteResource('/public/Tourist_Info/TechnologyMu
seum.xml',DBMS_XDB_REPOS.DELETE_FORCE);
   END IF;


   IF
DBMS_XDB_REPOS.existsResource('/public/Tourist_Info/MysorePalace
.xml')
   THEN

DBMS_XDB_REPOS.deleteResource('/public/Tourist_Info/MysorePalace
.xml',DBMS_XDB_REPOS.DELETE_FORCE);
   END IF;


   IF
DBMS_XDB_REPOS.existsResource('/public/Tourist_Info/Handicrafts.
xml')
   THEN

DBMS_XDB_REPOS.deleteResource('/public/Tourist_Info/Handicrafts.
xml',DBMS_XDB_REPOS.DELETE_FORCE);
   END IF;


   IF
DBMS_XDB_REPOS.existsResource('/public/Tourist_Info/NationalPark
.xml')
   THEN
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
DBMS_XDB_REPOS.deleteResource('/public/Tourist_Info/NationalPark
.xml',DBMS_XDB_REPOS.DELETE_FORCE);
   END IF;

   IF
DBMS_XDB_REPOS.existsResource('/public/Tourist_Info/ChennaiMuseu
m.xml')
   THEN

DBMS_XDB_REPOS.deleteResource('/public/Tourist_Info/ChennaiMuseu
m.xml',DBMS_XDB_REPOS.DELETE_FORCE);
   END IF;
END;
/

commit;
```

```
Script Output
Task completed in 0.024 seconds
anonymous block completed
committed.
```

2. You need a folder within Oracle XML DB Repository to store all the information about tourist attractions in XML files. By using `DBMS_XDB`, create an anonymous block to create the `Tourist_Info` folder in `/public`. Use `lab_17_02_02_cre_resources1.sql` for help. Replace <***TODO***> with the appropriate code, save the changes, and then run `lab_17_02_02_cre_resources1.sql`.

```
Declare
  result BOOLEAN;
BEGIN
  result := DBMS_XDB_REPOS.createFolder('/public/Tourist_Info');
  END;
/
commit;
```

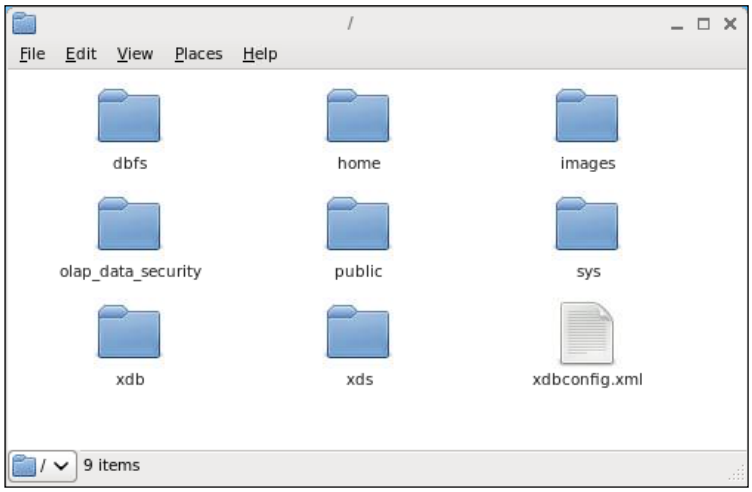
```
Script Output
Task completed in 0.024 seconds
anonymous block completed
committed.
```

Oracle University and Error : You are not a Valid Partner use only

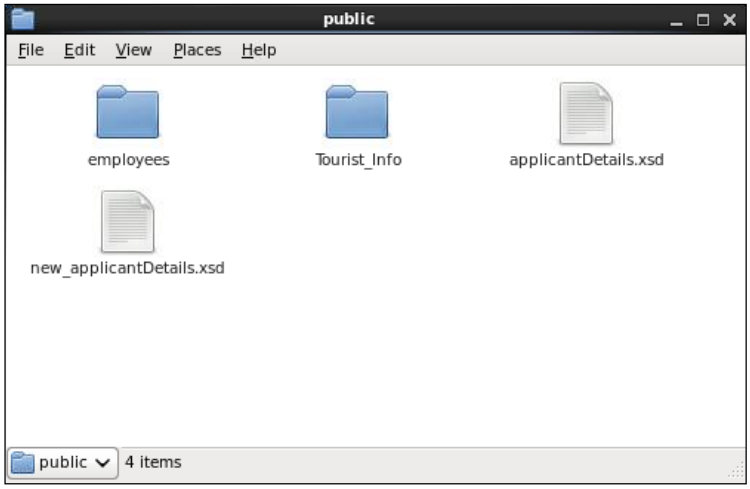
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. By using WebDAV, view the folder structure that was created in step 2.
   a. Start up the WebDav connection, `localhost`, that you created in Practice 12 or create a new WebDav connection if you get an error.



   b. Open the `public` folder.



4. Now that you have the `Tourist_Info` folder, you want to create XML files for each tourist attraction. By using `DBMS_XDB_REPOS`, create resources within the `/public/Tourist_Info` folder. The `Tourist_Info` folder should contain the following files:
   - `BangalorePalace.xml`
   - `TechnologyMuseum.xml`
   - `MysorePalace.xml`
   - `NationalPark.xml`
   - `ChennaiMuseum.xml`

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

You may use `lab_17_02_04_cre_resources2.sql` for help. Replace <***TODO***> with the appropriate code to create the respective repository resources and save the file. Run `lab_17_02_04_cre_resources2.sql`.

```
--Tourist attractions go into XML DB files (called resources)
DECLARE
   res BOOLEAN;
BEGIN
   res :=
DBMS_XDB_REPOS.createResource('/public/Tourist_Info/BangalorePal
ace.xml',
      '<?xml version="1.0"?>
   <ta>
     <ta_name>Bangalore Palace</ta_name>
       <ta_contacts>
       <contact_name>Head Office</contact_name>
       <contact_phone>(80)2478 2143</contact_phone>
     </ta_contacts>
     <ta_contacts>
       <contact_name>Visitor Desk</contact_name>
       <contact_phone>(80)2478 2144</contact_phone>
     </ta_contacts>
     </ta>



   ');


   res :=
DBMS_XDB_REPOS.createResource('/public/Tourist_Info/TechnologyMu
seum.xml',
      '
     <ta>
     <ta_name>Bangalore State Technology Museum</ta_name>
       <ta_contacts>
       <contact_name>Museum Executive Officer</contact_name>
       <contact_phone>(80)2378 2156</contact_phone>
     </ta_contacts>
     <ta_contacts>
       <contact_name>Visitor Desk</contact_name>
       <contact_phone>(80)2378 2157</contact_phone>
     </ta_contacts>
     </ta>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

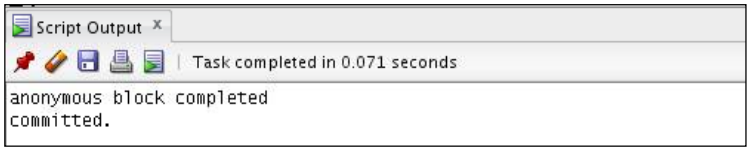
```
   ');

   res :=
DBMS_XDB_REPOS.createResource('/public/Tourist_Info/MysorePalace
.xml',
      '<ta>
     <ta_name>Maharaja Palace Mysore</ta_name>
      <ta_contacts>
       <contact_name>Visitor Desk</contact_name>
       <contact_phone>(821)2345 7656</contact_phone>
     </ta_contacts>
     </ta>



   ');

   res :=
DBMS_XDB_REPOS.createResource('/public/Tourist_Info/Handicrafts.
xml',
      '
       <ta>
     <ta_name>Mysore Handicrafts Gallery</ta_name>
      <ta_contacts>
       <contact_name>Manager</contact_name>
       <contact_phone>(821)2357 6212</contact_phone>
     </ta_contacts>
     </ta>

   ');

   res :=
DBMS_XDB_REPOS.createResource('/public/Tourist_Info/NationalPark
.xml',
      '
      <ta>
     <ta_name>Chennai National Park</ta_name>
      <ta_contacts>
       <contact_name>Manager</contact_name>
       <contact_phone>(44)2491 4570</contact_phone>
     </ta_contacts>
     <ta_contacts>
       <contact_name>Visitor Desk</contact_name>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
        <contact_phone>(44)2491 4571</contact_phone>
      </ta_contacts>
      </ta>

   ');

   res :=
DBMS_XDB_REPOS.createResource('/public/Tourist_Info/ChennaiMuseu
m.xml',
      '
        <ta>
      <ta_name>Chennai Museum</ta_name>
        <ta_contacts>
        <contact_name>Visitor Desk</contact_name>
        <contact_phone>(44)2493 4471</contact_phone>
      </ta_contacts>
      </ta>

   ');

END;
/

commit;
```

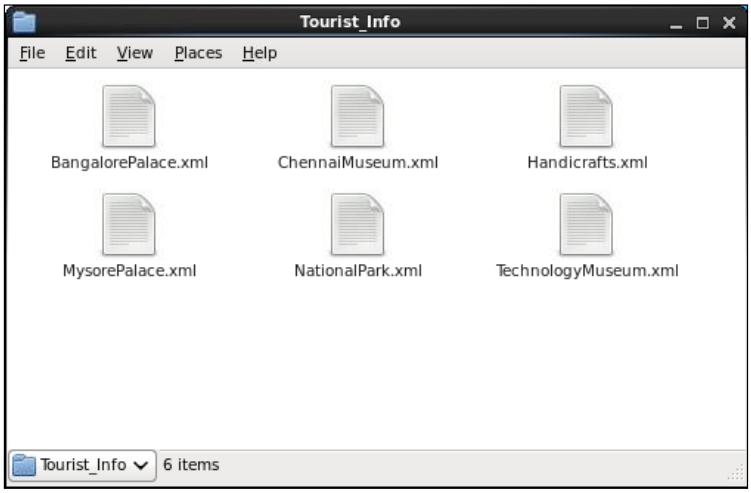


Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

5. By using WebDAV, view the folder structure that was created in step 4.



6. You need a relational table to store the information related to all tourist divisions. You want to store the tourist division ID, division name, telephone number, and the name of the main region. Create a table named `TOURIST_DIVISION` and insert data into it. Run `lab_17_02_06_cre_td_tab.sql`.

```
DROP TABLE tourist_division
/

--tourist divisions data goes into a table
CREATE TABLE tourist_division (
   td_id NUMBER,
   td_name VARCHAR2(50),
   td_phone VARCHAR2(15),
   td_region VARCHAR2(15)
);
/
INSERT INTO tourist_division VALUES
   (1, 'Bangalore Tourism Corp', '(80)2663 2142', 'Bangalore');
INSERT INTO tourist_division VALUES
   (2, 'Mysore Tourism Corp', '(821)2345 7634', 'Mysore');
INSERT INTO tourist_division VALUES
   (3, 'Chennai Tourism Center', '(44)2490 7291', 'Chennai');
```

```
CREATE TABLE succeeded.
1 rows inserted
1 rows inserted
1 rows inserted
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

7. Create a table to store the population data of the regions. Name the table `REGION_COUNT`. This table should contain two columns: `rc_region VARCHAR(15)` and `rc_city_population` as an `XMLType` column.

```
DROP TABLE region_count;
--Population data goes into an XMLType column
CREATE TABLE region_count (
   rc_region VARCHAR2(15),
   rc_city_population XMLType
);
```

```
Error starting at line 2 in command:
DROP TABLE region_count
Error report:
SQL Error: ORA-00942: table or view does not exist
00942. 00000 -  "table or view does not exist"
*Cause:
*Action:
table REGION_COUNT created.
```

8. Insert the population data into the `REGION_COUNT` table. Run `lab_17_02_08_load_rc_tab.sql`.

```
INSERT INTO region_count VALUES ('Bangalore', XMLType(
'<?xml version="1.0"?>
   <regions>
     <region>
     <region_name>Bangalore District</region_name>
     <population>5423</population>
     </region>
     <region>
     <region_name>Bangalore City</region_name>
     <population>432</population>
     </region>
</regions>'));


INSERT INTO region_count VALUES ('Mysore', XMLType(
   '<?xml version="1.0"?>
   <regions>
     <region>
     <region_name>Mysore District</region_name>
     <population>3421</population>
     </region>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
     <region>
     <region_name>Mysore City</region_name>
     <population>256</population>
     </region>
     <region>
     <region_name>Mysore Central</region_name>
     <population>312</population>
     </region>
</regions>'));


INSERT INTO region_count VALUES ('Chennai', XMLType(
     '<?xml version="1.0"?>
   <regions>
     <region>
     <region_name>Chennai District</region_name>
     <population>6790</population>
     </region>
     <region>
     <region_name>Chennai City</region_name>
     <population>378</population>
     </region>
     <region>
     <region_name>Chennai Central</region_name>
     <population>257</population>
     </region>
</regions>'));
```

9. By using `XMLQuery`, generate a set of region nodes from the `TOURIST_DIVISION` table. Use `lab_17_02_09_xquery1.sql` for help. Replace <***TODO***> with the appropriate code, save the changes, and then run `lab_17_02_09_xquery1.sql`. Use **Run Script** or **F5** in SQL Developer.

```
SELECT XMLQuery(
   'for $c in fn:collection("oradb:/HR/TOURIST_DIVISION")
    return $c' RETURNING CONTENT)
FROM dual;
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

```
Script Output
Task completed in 0.032 seconds
XMLQUERY('FOR$CINORA:VIEW("TOURIST_DIVISION")RETURN$C'RETURNINGCONTENT)
--------------------------------------------------------------------------------
<ROW><TD_ID>1</TD_ID><TD_NAME>Bangalore Tourism Corp</TD_NAME><TD_PHONE>(80)2663 2142</TD_PHONE><TD_REGION>Ban
```

10. By using `XMLQuery`, generate data from the `TOURIST_DIVISION` table, and then transform the data. The output should look similar to the following:

```
<regions><region><name>region name</name><destination
phone="Tourist Division Phone">Tourist Division
Name</destination></region></regions>
```

Use `lab_17_02_10_xquery2.sql` for help. Replace <***TODO***> with the appropriate code, save the changes, and then run `lab_17_02_10_xquery2.sql`. Use **Run Script** or **F5** in SQL Developer.

```
SELECT XMLQuery(
   '<regions>
    {for $c in fn:collection("oradb:/HR/TOURIST_DIVISION")
     let $td_region := $c/ROW/TD_REGION/text(),
         $td_name := $c/ROW/TD_NAME,
         $td_phone := $c/ROW/TD_PHONE/text()
     order by $td_region
     return
        <region>
           <name>{$td_region}</name>
           <destination
phone="{$td_phone}">{$td_name/text()}</destination>
        </region>}
    </regions>' RETURNING CONTENT)
FROM dual;
```

```
Script Output
Task completed in 0.024 seconds
XMLQUERY('<REGIONS>{FOR$CINFN:COLLECTION("ORADB:/HR/TOURIST_DIVISION")LET$TD_REGION:=$C/ROW/TD_REGION/TEXT(),$TD_NAME:=
--------------------------------------------------------------------------------
<regions><region><name>Bangalore</name><destination phone="(80)2663 2142">Bangalore Tourism Corp</destination></region>
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

11. By using `XMLQuery`, generate the sum of population in each region from the `REGION_COUNT` table. Use `lab_17_02_11_xquery3.sql` for help. Replace <***TODO***> with the appropriate code, save the changes, and then run `lab_17_02_11_xquery3.sql`. Use **Run Script** or **F5** in SQL Developer.

```
SELECT XMLQuery(
   '<regions>
    {for $c in fn:collection("oradb:/HR/TOURIST_DIVISION"),
         $rc in fn:collection("oradb:/HR/REGION_COUNT")
     let $td_region := $c/ROW/TD_REGION/text(),
         $td_name := $c/ROW/TD_NAME,
         $td_phone := $c/ROW/TD_PHONE/text(),
         $rc_region := $rc/ROW/RC_REGION/text(),
         $rc_population :=
$rc/ROW/RC_CITY_POPULATION/regions/region/population
     where $td_region = $rc_region
     order by $td_region
     return
        <region population
="{xs:unsignedInt(sum($rc_population))}">
           <name>{$td_region}</name>
           <destination
phone="{$td_phone}">{$td_name/text()}</destination>
           </region>}
    </regions>' RETURNING CONTENT)
FROM dual;
```

```
Script Output
Task completed in 0.08 seconds
XMLQUERY('<REGIONS>{FOR$CINFN:COLLECTION("ORADB:/HR/TOURIST_DIVISION"),$RCINFN:COLLECTION("ORADB:/HR/REGION_COUN
--------------------------------------------------------------------------------------------------------------
<regions><region population="5855"><name>Bangalore</name><destination phone="(80)2663 2142">Bangalore Tourism Co
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practices for Lesson 18: Case Study

Chapter 18

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-1: Creating a New Database Connection and Setting the Autotrace Parameters

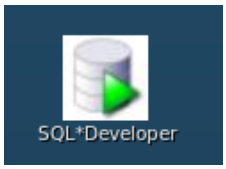
Overview

In this practice, you create a new database connection to the `XDBOE` schema, connect to the schema, and then set the Autotrace parameters.

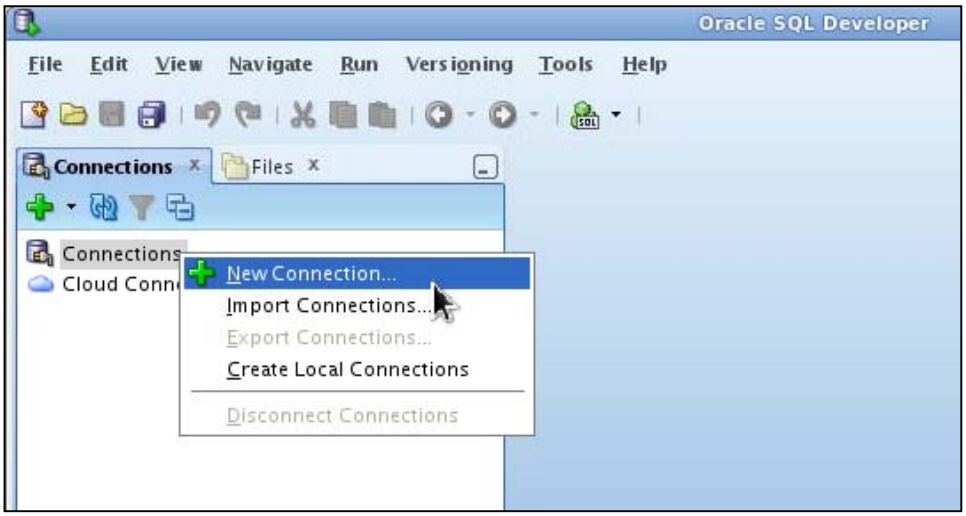
Assumptions

Tasks

1. Double-click the SQL Developer icon on the desktop to start the application.



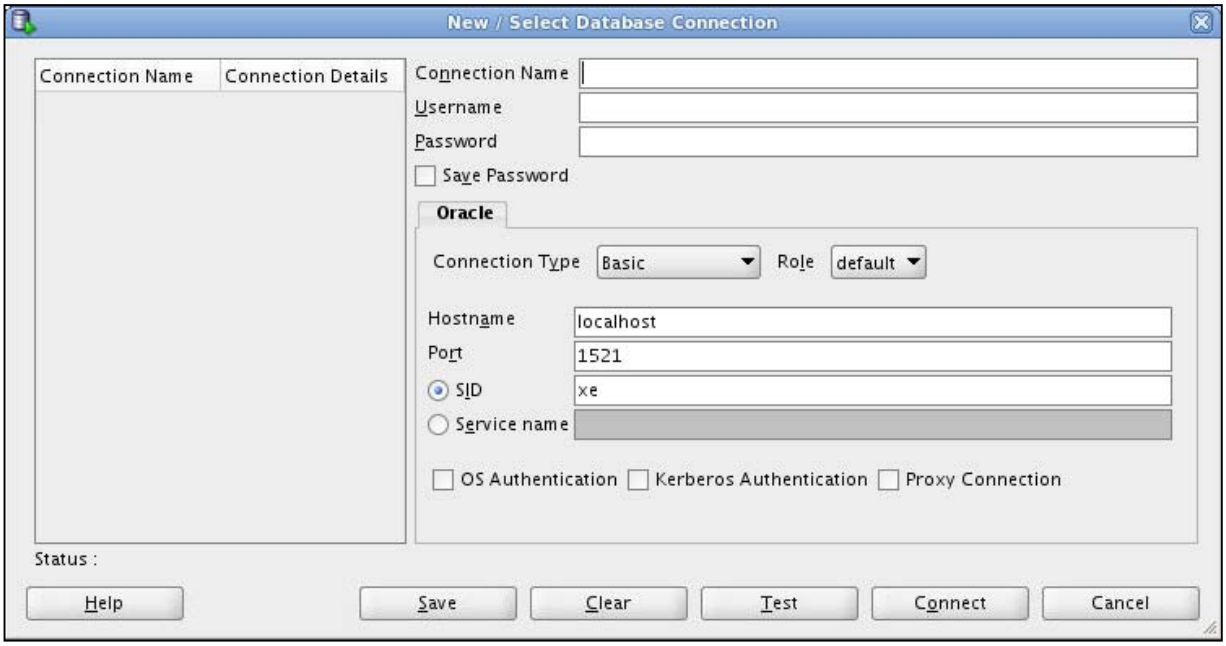
2. Create a database connection named `XDBOE`. On the Connections tab, right-click Connections, and then select New Connection. The New / Select Database Connection window is displayed.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



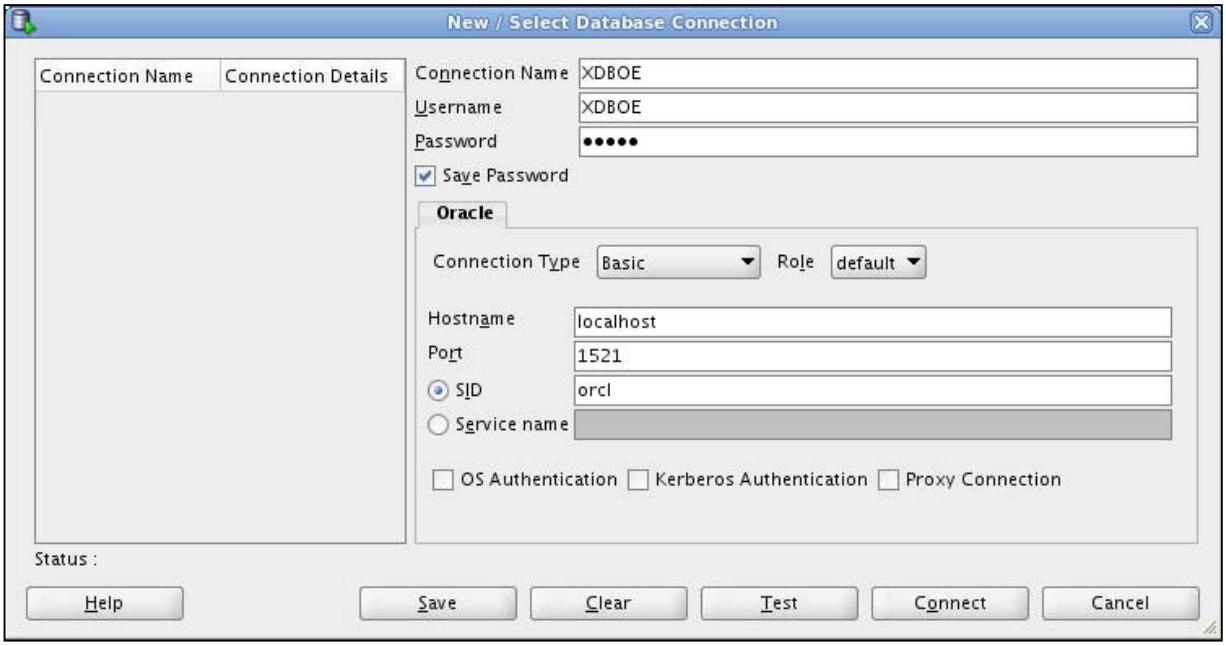
3. Enter the following details in the New / Select Database Connection window:

Property	Value
Connection Name	XDBOE
Username	`XDBOE`
Password	`oracle`
Save Password	Enable
Hostname	localhost
SID	orcl

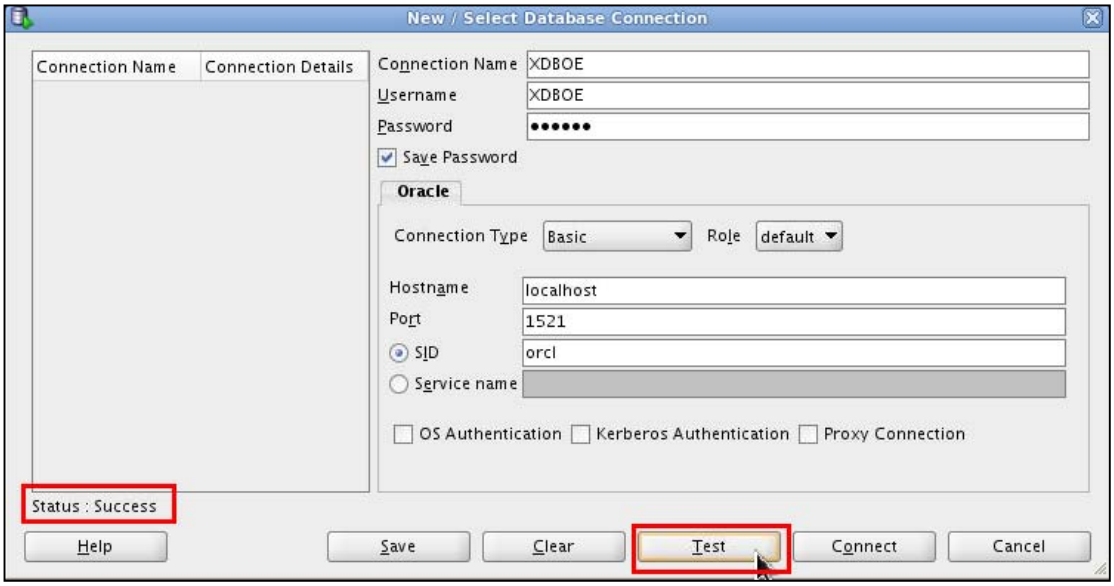
Oracle University and Error : You are not a Valid Partner use only

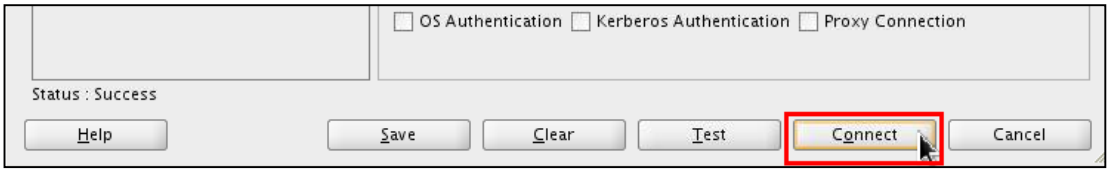
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



4. Test the connection, and then connect to the database by using the new database connection. Click Test to make sure that the connection has been set correctly. Click Connect after the test status shows Success.





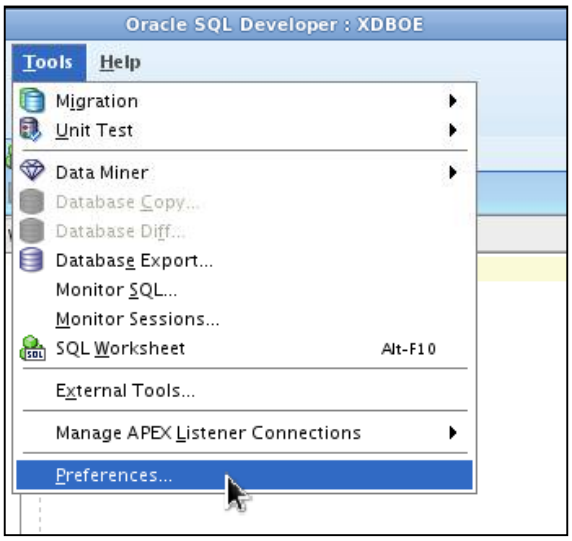
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

5. Set the Autotrace parameters as follows:

a. Select Tools > Preferences. The Preferences dialog box is displayed.

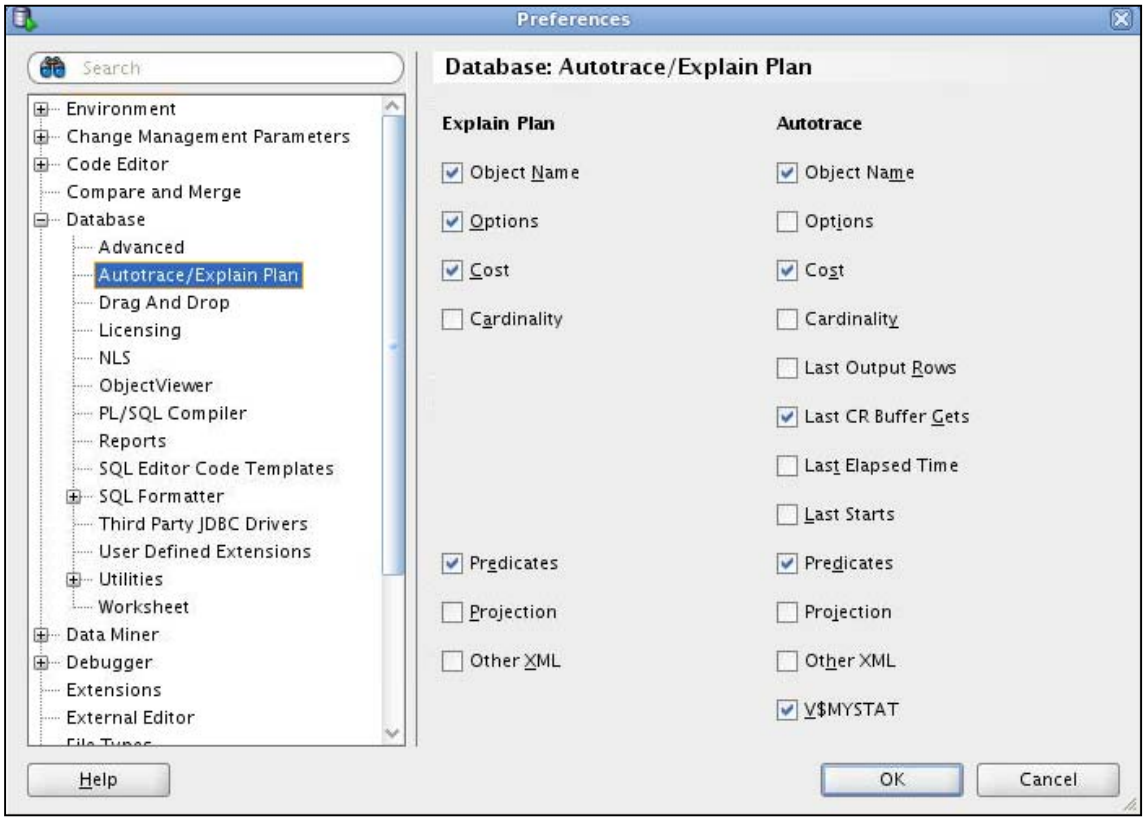


Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

b. Expand the Database node and then select the Autotrace/Explain Plan parameter.

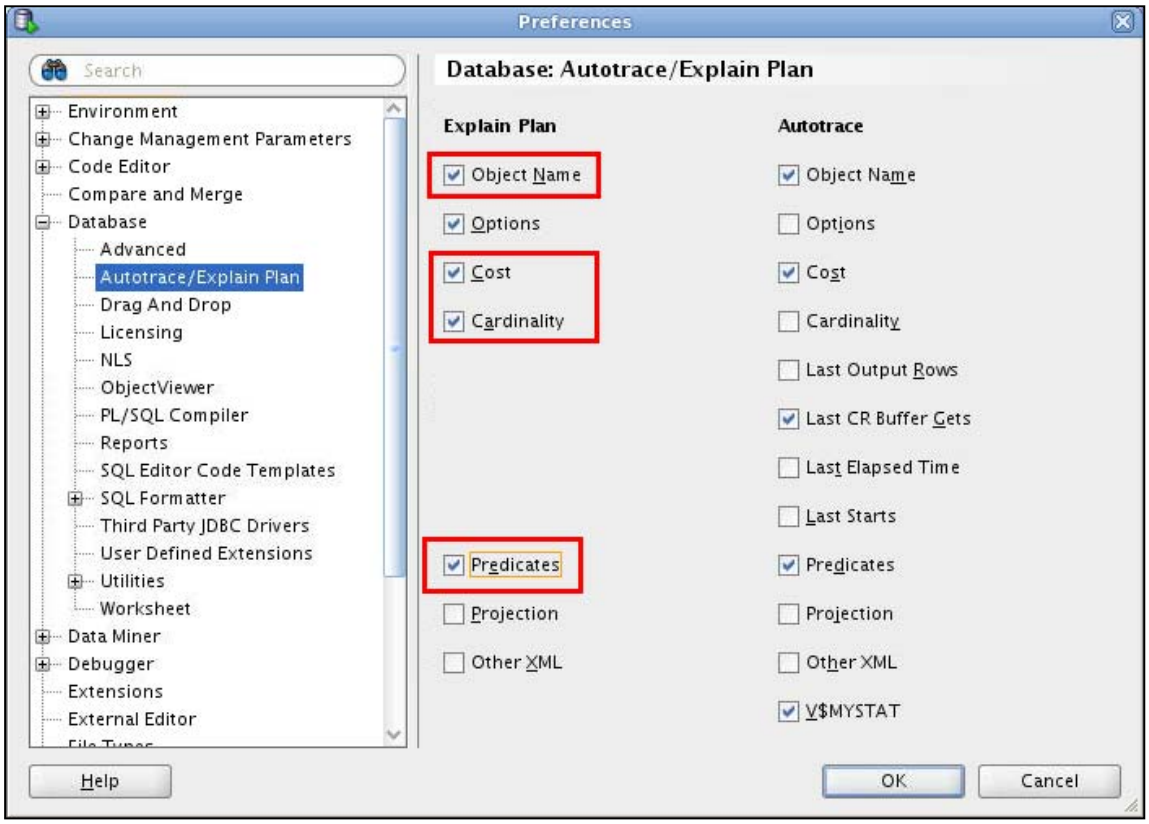


Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

c. Make sure to select the following check boxes if not already selected, and then click OK:
   1) Object_Name
   2) Cost
   3) Cardinality
   4) Predicates



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-2: Creating an `XMLType` Table and Loading Data

Overview

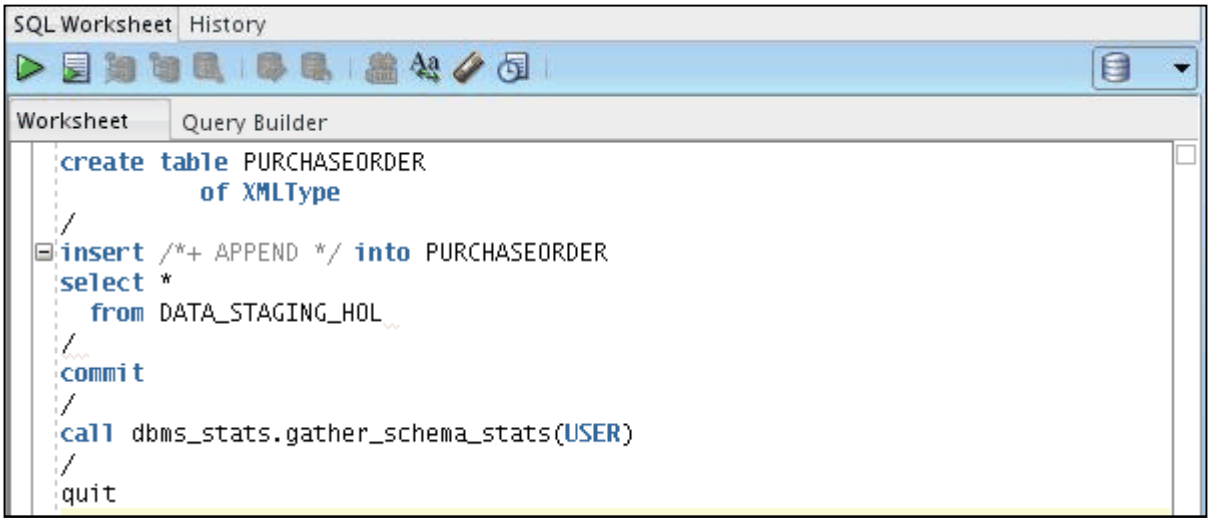
In this practice, you create an `XMLType` table in the `XDBOE` schema and then load XML documents into the newly created table. Finally, you gather some statistics and then display the new table in SQL Developer.

Assumptions

Tasks

1. Use the `lab_18_02_01.sql` script to create a simple binary XML `XMLType` table named `purchaseorder`. Replace **<TODO>** with the appropriate syntax to create the table. This script also loads data into the newly created table, and then gathers some statistics.

   Use the Files tab to navigate and open the `lab_18_02_01.sql` script, and then select Open to display it in the SQL Worksheet area. Alternatively, you can click the Open icon in the toolbar to quickly access the script.



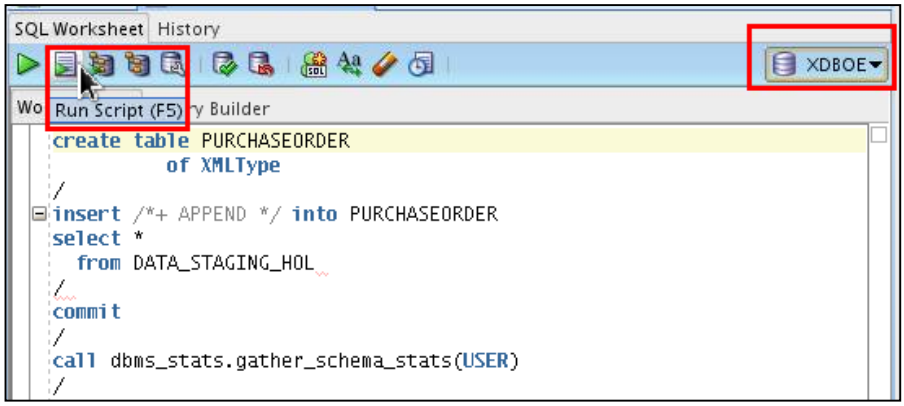
2. Run the script using the `XDBOE` database connection.

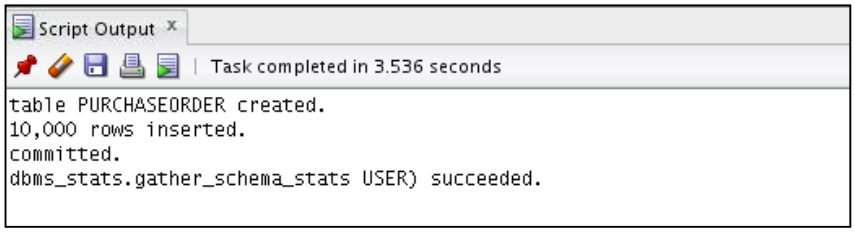
   Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon (or press the F5 key) to execute the script. This script creates a very simple `XMLType` table named `PURCHASEORDER` using the default settings. In Oracle Database Release 12.1.0.1.0, this is `XMLType` stored as Binary XML in a Secure File LOB. Binary XML stores the XML in a "post-parsed" native XML format that allows for efficient storage, indexing, and processing of XML. Because the format is "post-parsed", the database does not need to perform any additional parsing operations when performing XQuery operations on the XML content. The script then loads 10,000 XML documents into the `PURCHASEORDER` table. The output is displayed in the Script Output tab.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

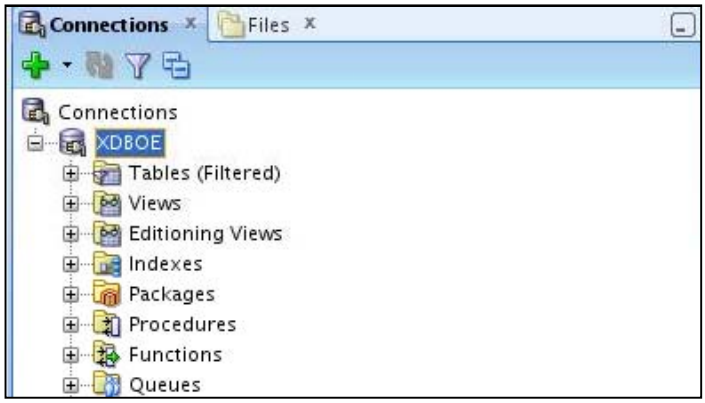
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED





3. Review the `PURCHASEORDER` table in SQL Developer.

   a. Select the Connections tab, and then click the + icon to the left of the `XDBOE` connection to expand it.

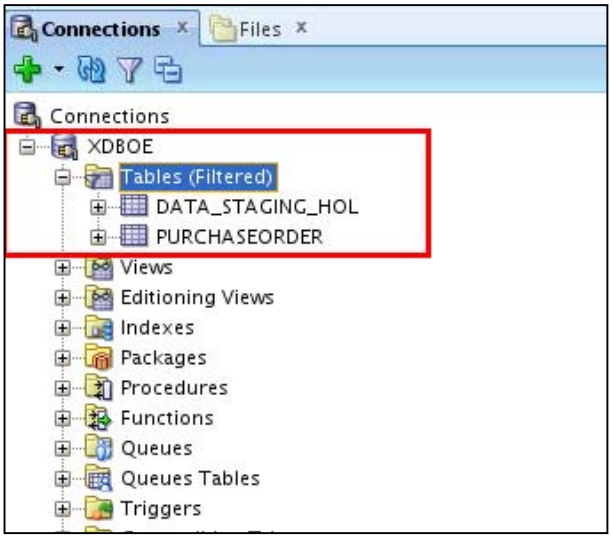


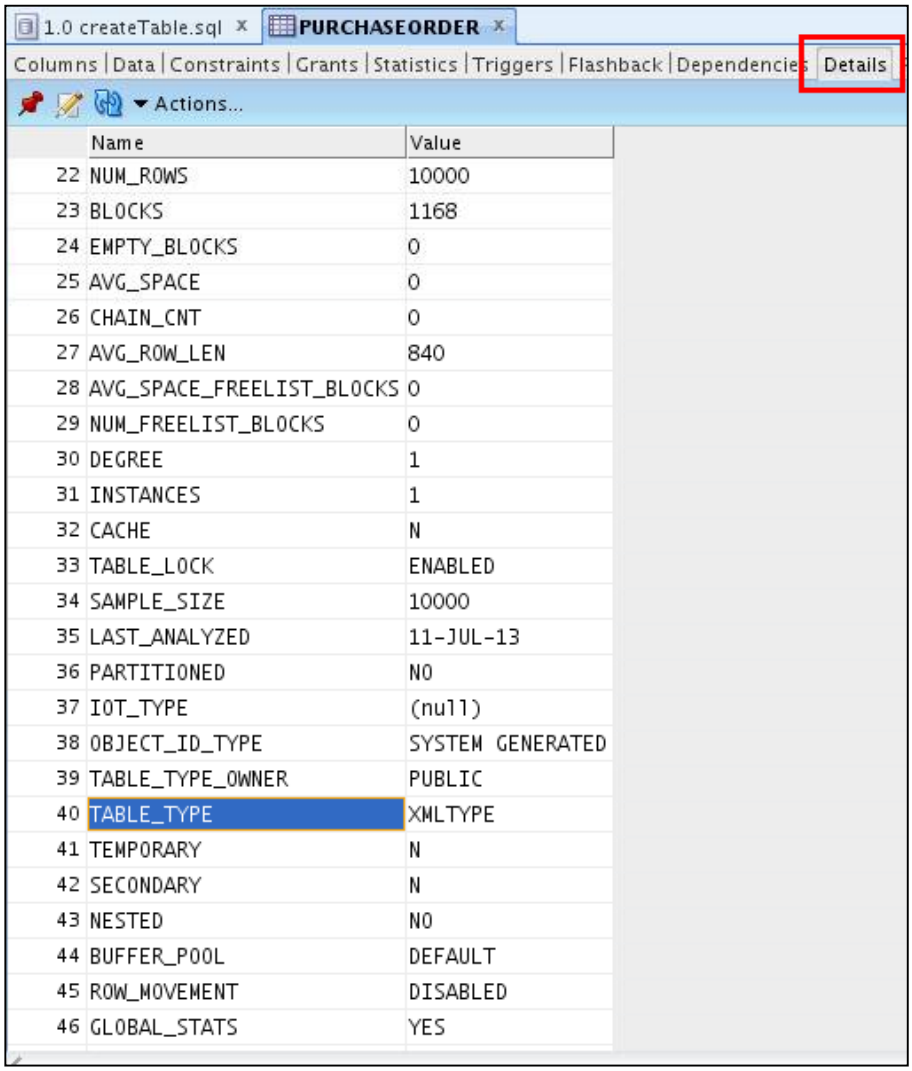
   b. Expand Tables, and then select `PURCHASEORDER`. In the right pane, select the Details tab to examine the values of the `NUM_ROWS` and the `TABLE_TYPE` parameters. Notice the highlighted `TABLE_TYPE` parameter with the value `XMLTYPE`.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED





	Name	Value
22	NUM_ROWS	10000
23	BLOCKS	1168
24	EMPTY_BLOCKS	0
25	AVG_SPACE	0
26	CHAIN_CNT	0
27	AVG_ROW_LEN	840
28	AVG_SPACE_FREELIST_BLOCKS	0
29	NUM_FREELIST_BLOCKS	0
30	DEGREE	1
31	INSTANCES	1
32	CACHE	N
33	TABLE_LOCK	ENABLED
34	SAMPLE_SIZE	10000
35	LAST_ANALYZED	11-JUL-13
36	PARTITIONED	NO
37	IOT_TYPE	(null)
38	OBJECT_ID_TYPE	SYSTEM GENERATED
39	TABLE_TYPE_OWNER	PUBLIC
40	TABLE_TYPE	XMLTYPE
41	TEMPORARY	N
42	SECONDARY	N
43	NESTED	NO
44	BUFFER_POOL	DEFAULT
45	ROW_MOVEMENT	DISABLED
46	GLOBAL_STATS	YES

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Practice 18-3: Using XQuery to Access XML Content

Overview

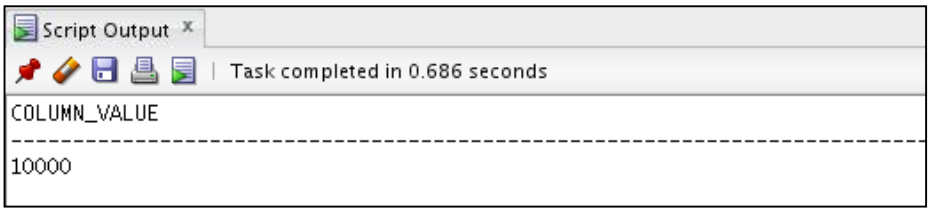
In this practice, you use XQuery to access XML content. XQuery is to XML content what SQL is to relational data. The XQuery standard was developed by the W3C and it is the natural language for querying, manipulating, and updating XML content. In this practice, you use XQuery to work with XML content stored in an Oracle database.

Assumptions

Tasks

1. Use XQuery and the `XMLTable` to count the number of documents in the `PURCHASEORDER` table using the standard XQuery function `fn:collection()` to access the contents of the table. Remember that `fn:collection()` expects a path that identifies the set of XML documents to be processed. In this case, the path is prefixed with the protocol `oradb`, indicating that the components of the path should be interpreted as `DATABASE_SCHEMA` and `TABLE`.

```
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/XDBOE/PURCHASEORDER"))'
       )
/
```



2. Use XQuery to select a single document from the `PURCHASEORDER` table. Use a predicate on the `Reference` element to return one single document with the following `Reference` element: `AFRIPP-2012060818343243PDT`.

   The query use predicates to restrict the documents that are returned. In this case the predicate on the `Reference` element uniquely identifies a single document. The document is returned as an `XMLType` object, and the SQL Developer Script Output window shows this by outputting the text `(XMLTYPE)`. In order to see the XML in the SQL Developer script output tab, the XML must be serialized, or converted into a textual representation, using the `XMLSerialize()` function. `XMLSerialize()` generates a textual representation of the XML and returns it as a `CLOB`, `BLOB` or `VARCHAR2` data type.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
select *
  from XMLTABLE(
           'for $i in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[Refere
nce/text()=$REFERENCE]
            return $i'
            passing 'AFRIPP-2012060818343243PDT' as "REFERENCE"
       )
/
```

Script Output

Task completed in 0.186 seconds

```
COLUMN_VALUE
--------------------------------------------------------------------------------------------------------------
<PurchaseOrder xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="http://loca
  <Reference>AFRIPP-2012060818343243PDT</Reference>
  <Actions>
    <Action>
      <User>MHARTSTE</User>
    </Action>
  </Actions>
  <Rejection/>
  <Requestor>Adam Fripp</Requestor>
  <User>AFRIPP</User>
  <CostCenter>A50</CostCenter>
  <ShippingInstructions>
    <name>Adam Fripp</name>
    <Address xsi:type="address_US">
      <street>200 Sporting Green</street>
      <city>South San Francisco</city>
      <state>CA</state>
      <zipCode>99236</zipCode>
      <country>United States of America</country>
    </Address>
    <telephone>983-814-9068</telephone>
  </ShippingInstructions>
  <SpecialInstructions>COD</SpecialInstructions>
  <LineItems>
    <LineItem ItemNumber="1">
      <Part Description="Star Trek III: the Search for Spock" UnitPrice="19.95">97360162141</Part>
      <Quantity>6.0</Quantity>
    </LineItem>
    <LineItem ItemNumber="2">
      <Part Description="Agatha Christie&apos;s Miss Marple 1" UnitPrice="32.95">733961702576</Part>
      <Quantity>7.0</Quantity>
    </LineItem>
    <LineItem ItemNumber="3">
      <Part Description="Coyote Ugly &amp; Pretty Woman 10th Anniv" UnitPrice="32.95">786936161274</Part>
      <Quantity>6.0</Quantity>
    </LineItem>
  </LineItems>
</PurchaseOrder>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. Use XQuery with multiple predicates, to return a single `Reference` node. Use `A60` as the predicate on `CostCenter`, `Diana Lorentz` as the `Requestor`, and `5` as the `Quantity`. Use the following expression as the last predicate:
   `count(LineItems/LineItem) > $QUANTITY]/Reference`

   In a real-world example, these values would be supplied as bind variables rather than hard-coded literals. Also note that the XQuery expression in this example terminates in the `Reference` element. Consequently the result consists of just the `Reference` element from the documents that match the supplied predicates.

```
select *
  from XMLTABLE(
          'for $i in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[CostCenter
=$CC and Requestor=$REQUESTOR and count(LineItems/LineItem) >
$QUANTITY]/Reference
           return $i'
           passing 'A60' as "CC", 'Diana Lorentz' as "REQUESTOR", 5
as "QUANTITY"
       )
/
```

```
XMLSERIALIZE(CONTENTCOLUMN_VALUEASCLOB)
--------------------------------------------------------------------------------
<Reference>DLORENTZ-20120411133920295PDT</Reference>

<Reference>DLORENTZ-2012042922031647PDT</Reference>

<Reference>DLORENTZ-20121219144418724PST</Reference>

<Reference>DLORENTZ-20120206180918983PST</Reference>

<Reference>DLORENTZ-20120710194550792PDT</Reference>

<Reference>DLORENTZ-20120723165626922PDT</Reference>

<Reference>DLORENTZ-20120728151113368PDT</Reference>

<Reference>DLORENTZ-20120602200119333PDT</Reference>

<Reference>DLORENTZ-20120318211347802PST</Reference>

<Reference>DLORENTZ-20120320185010571PST</Reference>
```

. . .

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
<Reference>DLORENTZ-20120513165042874PDT</Reference>

<Reference>DLORENTZ-20120515153151962PDT</Reference>

<Reference>DLORENTZ-20121103141148589PST</Reference>

<Reference>DLORENTZ-2012112521245631PST</Reference>

<Reference>DLORENTZ-20120907132707431PDT</Reference>


 15 rows selected
```

4. Review and run the `lab_18_03_04.sql` script. This script uses XQuery to construct a new summary document from the documents that match the specified predicates. This example also demonstrates the use of nested `FOR` loops, one for the set of `PurchaseOrder` documents, and another for the `LineItem` elements.

```
select *
  from XMLTable(
         '<Summary UPC="{$UPC}">
          {
           for $p in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder
            for $l in $p/LineItems/LineItem[Quantity > $Quantity
and Part/text() =$UPC]
            order by $p/Reference
               return
                 <PurchaseOrder
reference="{$p/Reference/text()}"
lineItem="{fn:data($l/@ItemNumber)}" Quantity="{$l/Quantity}"/>
          }
          </Summary>'
          passing  '707729113751' as UPC, 3 as "Quantity"
       )
/
```

```
Script Output
Task completed in 1.155 seconds
COLUMN_VALUE
--------------------------------------------------------------------------------
<Summary UPC="707729113751"><PurchaseOrder reference="DBERNSTE-20120427202452121PDT" lineItem="5" Quantity="6
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

The query uses XQuery to create a new document from the documents that match the supplied predicates. Note that the generated XMLType is not "pretty printed". Pretty Printing is a function of serialization.

5. Use `XMLSerialize` in the previous query to convert the `XMLType` to a `CLOB`. This allows the result to be viewed in SQL Developer versions that do not support rendering `XMLType`.

```
set long 5000

select XMLSERIALIZE(CONTENT COLUMN_VALUE AS CLOB INDENT SIZE=2)
  from XMLTable(
         '<Summary UPC="{$UPC}">
          {
           for $p in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder
            for $l in $p/LineItems/LineItem[Quantity > $Quantity and
Part/text() =$UPC]
            order by $p/Reference
              return
                <PurchaseOrder reference="{$p/Reference/text()}"
lineItem="{fn:data($l/@ItemNumber)}" Quantity="{$l/Quantity}"/>
          }
          </Summary>'
          passing  '707729113751' as UPC, 3 as "Quantity"
       )
/
```

```
Script Output
Task completed in 1.424 seconds
<Summary UPC="707729113751">
  <PurchaseOrder reference="DBERNSTE-20120427202452121PDT" lineItem="5" Quantity="6.0"/>
  <PurchaseOrder reference="LBISSOT-20120815132845576PDT" lineItem="5" Quantity="4.0"/>
  <PurchaseOrder reference="LDORAN-20120408225231939PDT" lineItem="3" Quantity="4.0"/>
  <PurchaseOrder reference="PVARGAS-20121030162426359PST" lineItem="2" Quantity="4.0"/>
  <PurchaseOrder reference="RLADWIG-20120406190600968PDT" lineItem="4" Quantity="9.0"/>
  <PurchaseOrder reference="WSMITH-20120224224555785PST" lineItem="1" Quantity="5.0"/>
</Summary>
```

The query uses `XMLSerialize()` to convert the document into a serialized form that is stored in a `CLOB` data type.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

6. Review and run the `lab_18_03_06.sql` script, which demonstrates how you can use `XMLTable` to create an inline relational view from the documents that match the XQuery expression. This query shows you how to use the `columns` clause of the `XMLTable` operator to create an inline relational view from the documents that match the supplied predicates. In this case, the XQuery expression generates a result document from each of the `PurchaseOrder` documents that match the supplied predicates, and then the `columns` clause maps elements and attributes in the result document into the columns of the inline view. This allows a conventional relational result to be created by executing an XQuery operation on XML content.

```
select *
  from xmlTable
       (
          'for $p in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder
            for $l in $p/LineItems/LineItem[Quantity > 3 and
Part/text() = "707729113751"]
              return
              <Result ItemNumber="{fn:data($l/@ItemNumber)}">
                {
                  $p/Reference,
                  $p/Requestor,
                  $p/User,
                  $p/CostCenter,
                  $l/Quantity
                }

<Description>{fn:data($l/Part/@Description)}</Description>
                <UnitPrice>{fn:data($l/Part/@UnitPrice)}</UnitPrice>
                <PartNumber>{$l/Part/text()}</PartNumber>
             </Result>'
             columns
             SEQUENCE      for ordinality,
             ITEM_NUMBER       NUMBER(3) path '@ItemNumber',
             REFERENCE     VARCHAR2( 30) path 'Reference',
             REQUESTOR     VARCHAR2(128) path 'Requestor',
             USERID        VARCHAR2( 10) path 'User',
             COSTCENTER    VARCHAR2(  4) path 'CostCenter',
             DESCRIPTION   VARCHAR2(256) path 'Description',
             PARTNO        VARCHAR2( 14) path 'PartNumber',
             QUANTITY       NUMBER(12,4) path 'Quantity',
             UNITPRICE      NUMBER(14,2) path 'UnitPrice'
       )
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
SEQUENCE ITEM_NUMBER REFERENCE                      REQUESTOR
-------- ----------- ------------------------------ ------------------------------
       1           5 DBERNSTE-20100427202452121PDT  David Bernstein
       2           3 LDORAN-20100408225231939PDT    Louise Doran
       3           4 RLADWIG-20100406190600968PDT   Renske Ladwig
       4           5 LBISSOT-20100815132845576PDT   Laura Bissot
       5           1 WSMITH-20100224224555785PST    William Smith
       6           2 PVARGAS-20101030162426359PST   Peter Vargas

6 rows selected
```

Note: The above result does not show all of the generated columns. You can see all the columns by running the statement using the Run statement icon on the toolbar. The output is displayed in the Query Result tab as follows:

Script Output | Query Result

SQL | All Rows Fetched: 6 in 2.271 seconds

	SEQUENCE	ITEM_NUMBER	REFERENCE	REQUESTOR	USERID	COSTCENTER	DESCRIPTION	PARTNO	QUANTITY	UNITPRICE
1	1	5	DBERNSTE-20120427202452121PDT	David Bernstein	DBERNSTE	A80	Dead Man's Walk	707729113751	6	27.95
2	2	3	LDORAN-20120408225231939PDT	Louise Doran	LDORAN	A80	Dead Man's Walk	707729113751	4	27.95
3	3	4	RLADWIG-20120406190600968PDT	Renske Ladwig	RLADWIG	A50	Dead Man's Walk	707729113751	9	27.95
4	4	5	LBISSOT-20120815132845576PDT	Laura Bissot	LBISSOT	A50	Dead Man's Walk	707729113751	4	27.95
5	5	1	WSMITH-20120224224555785PST	William Smith	WSMITH	A80	Dead Man's Walk	707729113751	5	27.95
6	6	2	PVARGAS-20121030162426359PST	Peter Vargas	PVARGAS	A50	Dead Man's Walk	707729113751	4	27.95

7. Review and run the `lab_18_03_07.sql` script, which joins relational and XML tables using XQuery.

```
select REQUESTOR, DEPARTMENT_NAME
  from HR.EMPLOYEES e, HR.DEPARTMENTS d,
       XMLTABLE
       (
         'for $p in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder
            where $p/User=$EMAIL and $p/Reference=$REFERENCE
            return $p'
          passing 'AFRIPP-2012060818343243PDT' as "REFERENCE",
e.EMAIL as "EMAIL"
          COLUMNS
          REQUESTOR path 'Requestor/text()',
          USERNAME  path 'User'
       )
 where e.DEPARTMENT_ID = d.DEPARTMENT_ID;
```

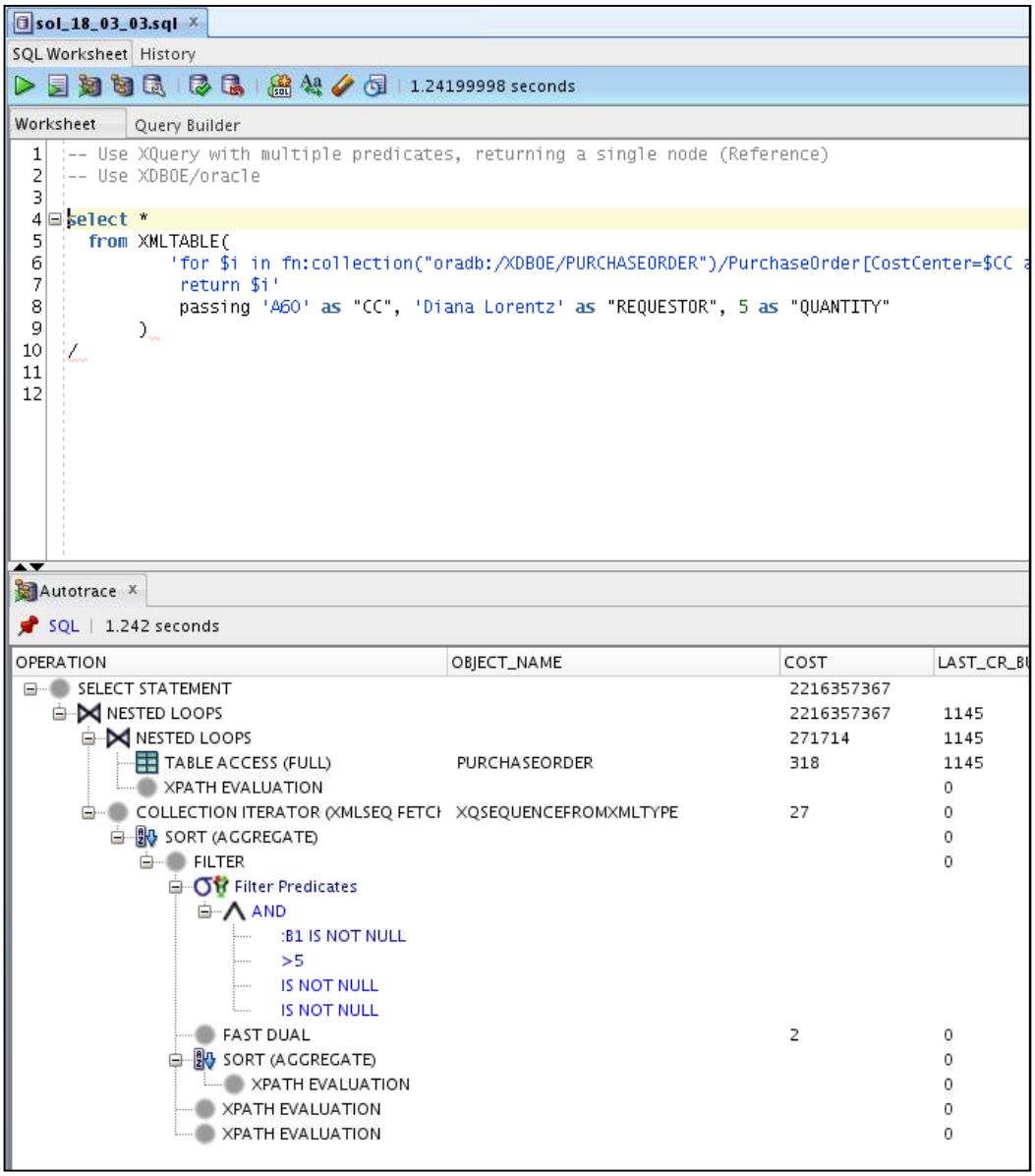
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
REQUESTOR
----------------------------------------------------------------------
Adam Fripp
```

8. You can use SQL Developer to examine the execution plans for XQuery expressions, in the same way it can be used when working with SQL queries.
   a. Using the Files tab, right-click the `sol_18_03_03.sql` script, and then select Open to display it in the SQL Worksheet area.
   b. Position the cursor at the start of the Query in the `sol_18_03_03.sql` script, and then click the Autotrace icon. This displays the execution plan for the query.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Because there are no indexes on the `PURCHASEORDER` table, the XQuery expression is executed by using streaming XPath evaluation. This means that each document is examined to see if it contains a node that matches the supplied predicate. Streaming XPath is a very efficient technique for performing the equivalent of a relational full-table scan on binary XML content.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-4: Optimizing XQuery Performance with XML Indexing

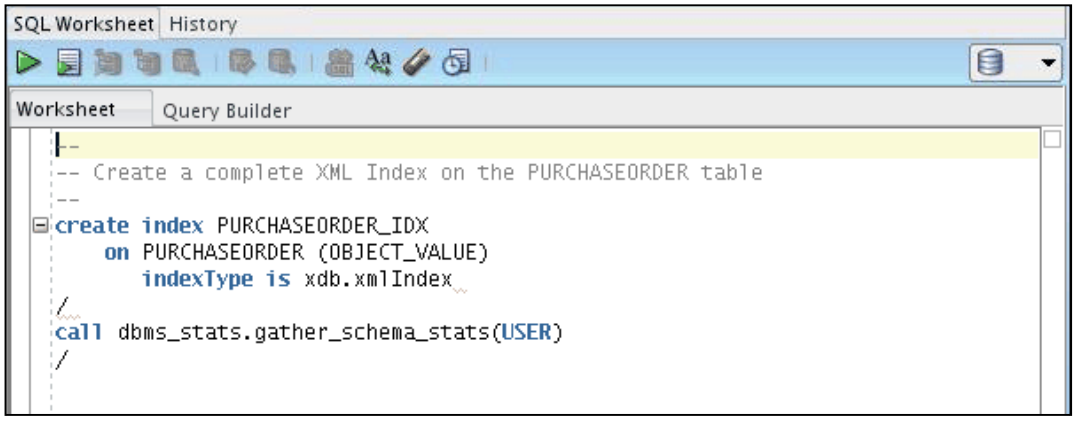
Overview

Just like with SQL, XQuery operations can be improved by creating appropriate indexes. Again just like with SQL, XML indexing leads to trade-offs between DML and Query operations. This practice introduces techniques for indexing binary XML content stored in the Oracle database.

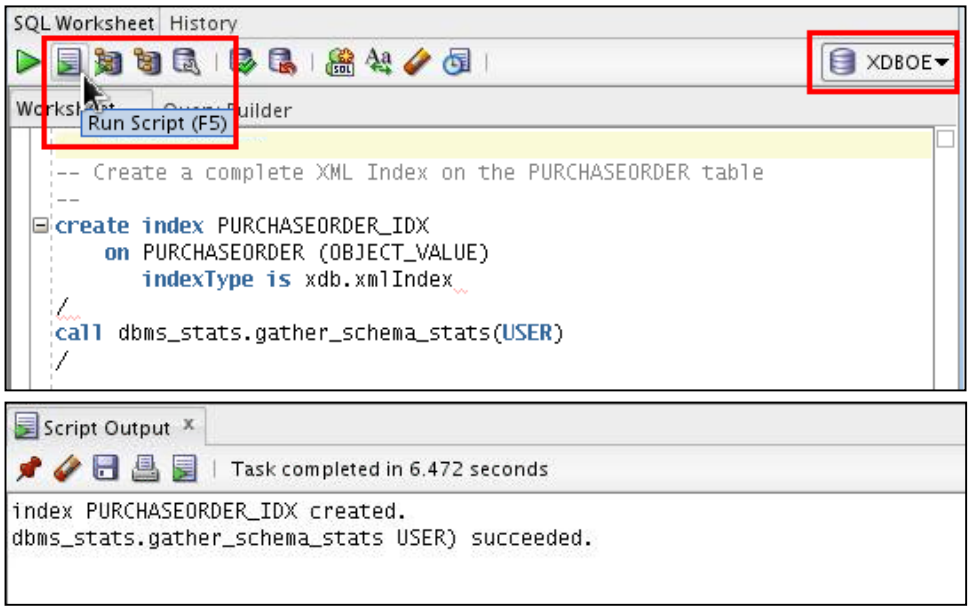
Assumptions

Tasks

1. Create a script to create a full unstructured `XMLIndex` on the `PURCHASEORDER` table named `PURCHASEORDER_IDX`. Next, add the following statement to the script to gather statistics: `call dbms_stats.gather_schema_stats(USER).`



2. Run the script as user `XDBOE`.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

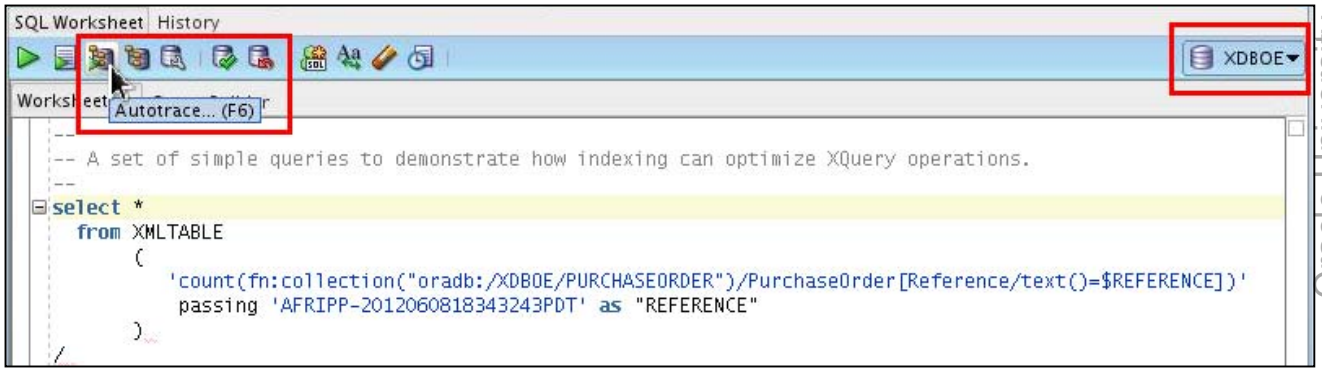
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

This script creates a full, unstructured `XMLIndex` on the `PURCHASEORDER` table. The index will contain an entry for every node in every document. The full index requires no up-front knowledge about the structure of the XML being indexed. The index can be used to optimize both path (does the node exist?), and path-value (does the node exist and does it have a particular value?) searches.

3. Open the `lab_18_04_03.sql` script.

```
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[Reference/text()=$REFERENCE])'
           passing 'AFRIPP-2012060818343243PDT' as "REFERENCE"
       )
/
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[User/text()=$USER])'
           passing 'AFRIPP' as "USER"
       )
/
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[LineItems/LineItem[Part/text()=$UPC]])'
           passing '707729113751' as "UPC"
       )
/
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[LineItems/LineItem[Part/text() = $UPC an
           passing '707729113751' as "UPC", 8 as "QUANTITY"
       )
/
```

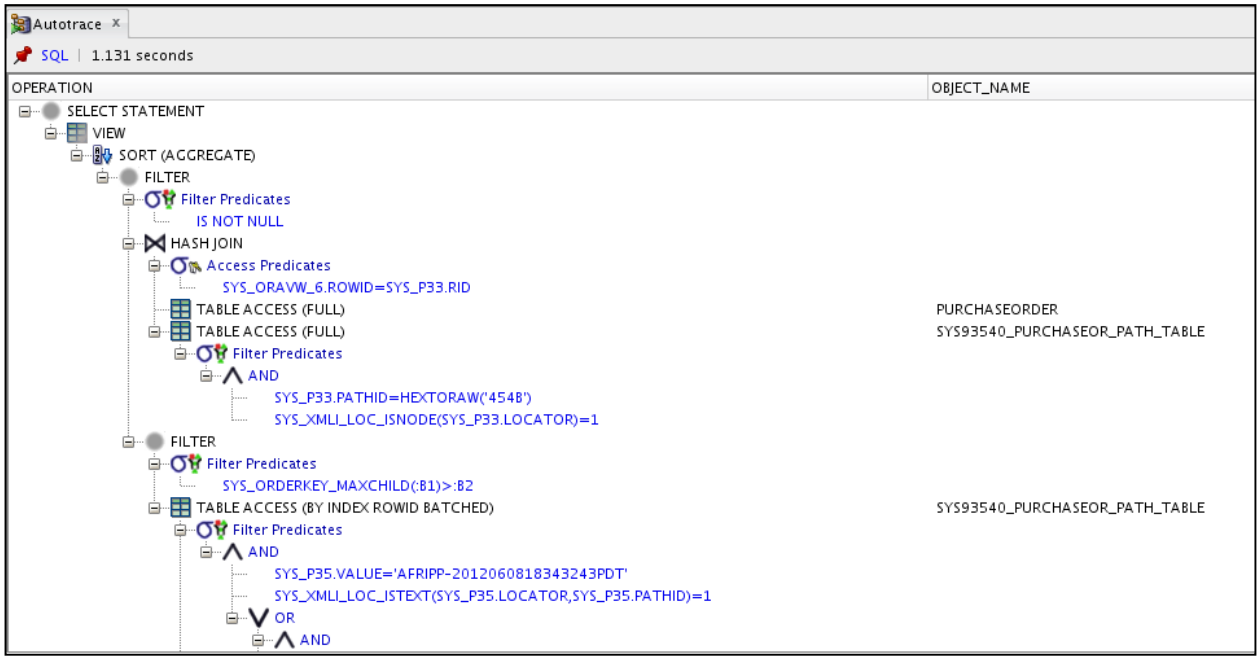
4. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, position the cursor at the start of the first query, and then click the Autotrace icon.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



The execution plan for the query is displayed in the Autotrace tab. The Autotrace output shows that the `XMLIndex` is used to determine which documents match the specified predicates.

5. Repeat step 4 for each of the remaining queries. In each case the Autotrace output shows that the `XMLIndex` is used to optimize the execution of the XQuery expressions.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-5: Optimizing XQuery Performance by Using a Path-Subsetted XML Index

Overview

A full unstructured index can be an expensive proposition in terms of both disk space usage and index maintenance overhead. To address these issues, XML Index provides two options:

- The first option is the ability to maintain the index asynchronously. With an asynchronous index, the DML operation does not wait for indexing to complete before returning control to the application that invoked it. Instead, the indexing takes place in near real-time, ensuring that index maintenance operations do not interfere with DML throughput.
- The second option is the ability to use path-subsetting to explicitly include or exclude certain sections of an XML document from the index. This reduces both the size of the index and the overhead associated with index maintenance, by indexing only those nodes that will be referenced in predicates. While a path-subsetted index does not require upfront knowledge of the structure of the XML being indexed, it does require some knowledge of the kinds of searches that will be performed on the XML in question.

In this practice, you create a path-subsetted XML index.

Assumptions

Tasks

1. Create a path-subsetted XML index on the `PURCHASEORDER` table. Narrow the focus of indexing by pruning the set of XPath expressions (paths) corresponding to XML fragments to be indexed, specifying a subset of all possible paths. This statement creates an index that indexes only the top-level element `PurchaseOrder` and some of its children, as follows:
   a. All `Part` elements and their descendants
   b. All `Reference` elements

```
drop index PURCHASEORDER_IDX
/
create index PURCHASEORDER_IDX
    on PURCHASEORDER (OBJECT_VALUE)
       indextype is XDB.XMLINDEX
       parameters (
        'paths (
           include (
              /PurchaseOrder/Reference
              /PurchaseOrder/LineItems/LineItem/Part/* ))'
       );
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

2. When gathering statistics for the optimizer on an `XMLType` table that is stored object-relationally, Oracle recommends that you gather statistics on all of the tables defined by the XML schema, that is, all of the tables in `USER_XML_TABLES`. You can use the `DBMS_STATS.gather_schema_stats` procedure to do this, or use `DBMS_STATS.gather_table_stats` on each such table. This informs the optimizer about all of the dependent tables that are used to store the `XMLType` data. Use the `DBMS_STATS.gather_schema_stats` procedure to gather statistics.

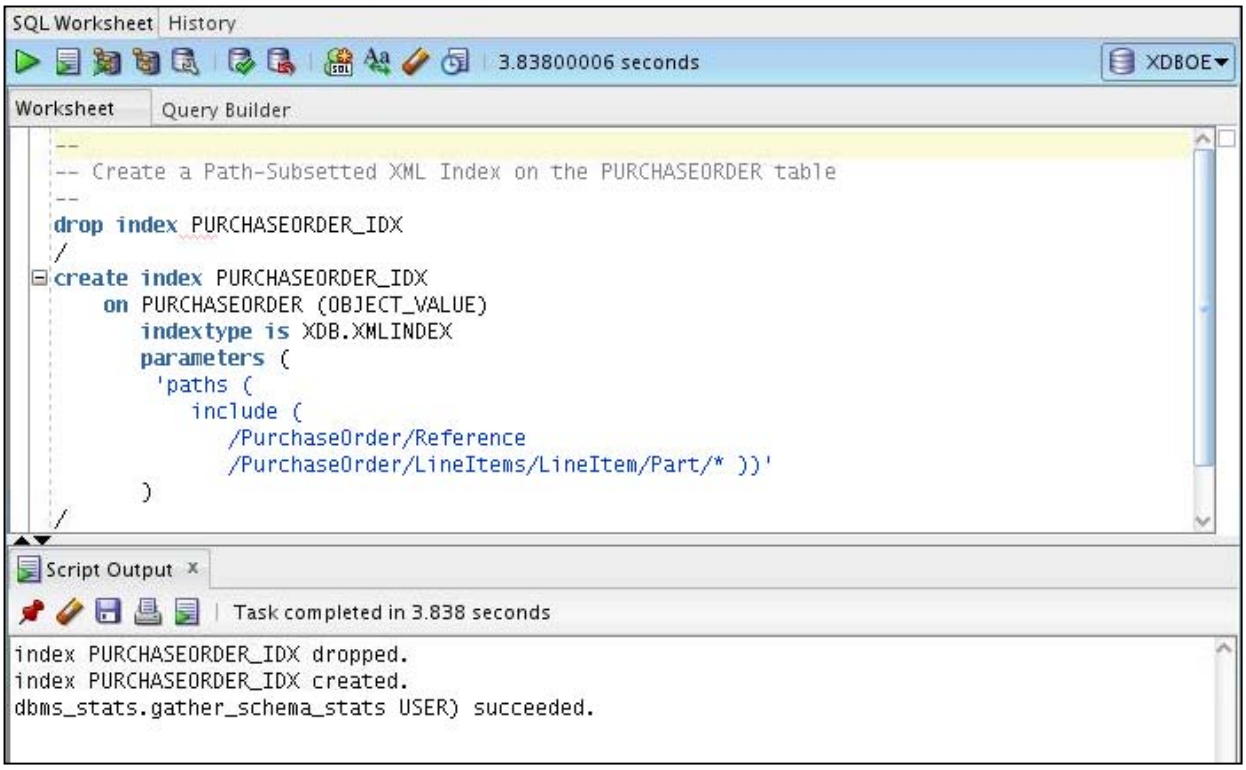
```
drop index PURCHASEORDER_IDX
/
create index PURCHASEORDER_IDX
    on PURCHASEORDER (OBJECT_VALUE)
       indextype is XDB.XMLINDEX
       parameters (
        'paths (
           include (
              /PurchaseOrder/Reference
              /PurchaseOrder/LineItems/LineItem/Part/* ))'
       )
/
call dbms_stats.gather_schema_stats(USER)
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. Run the script as user `XDBOE`. The output is displayed in the Script Output tab.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Return to the SQL Worksheet containing the `lab_18_04_03.sql` script.

```
--
-- A set of simple queries to demonstrate how indexing can optimize XQuery ope
--
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/OE/PURCHASEORDER")/PurchaseOrder[Refere
           passing 'AFRIPP-2010060818343243PDT' as "REFERENCE"
       )
/
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/OE/PURCHASEORDER")/PurchaseOrder[User/t
           passing 'AFRIPP' as "USER"
       )
/
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/OE/PURCHASEORDER")/PurchaseOrder[LineIt
           passing '707729113751' as "UPC"
       )
/
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/OE/PURCHASEORDER")/PurchaseOrder[LineIt
           passing '707729113751' as "UPC", 8 as "QUANTITY"
       )
/
```

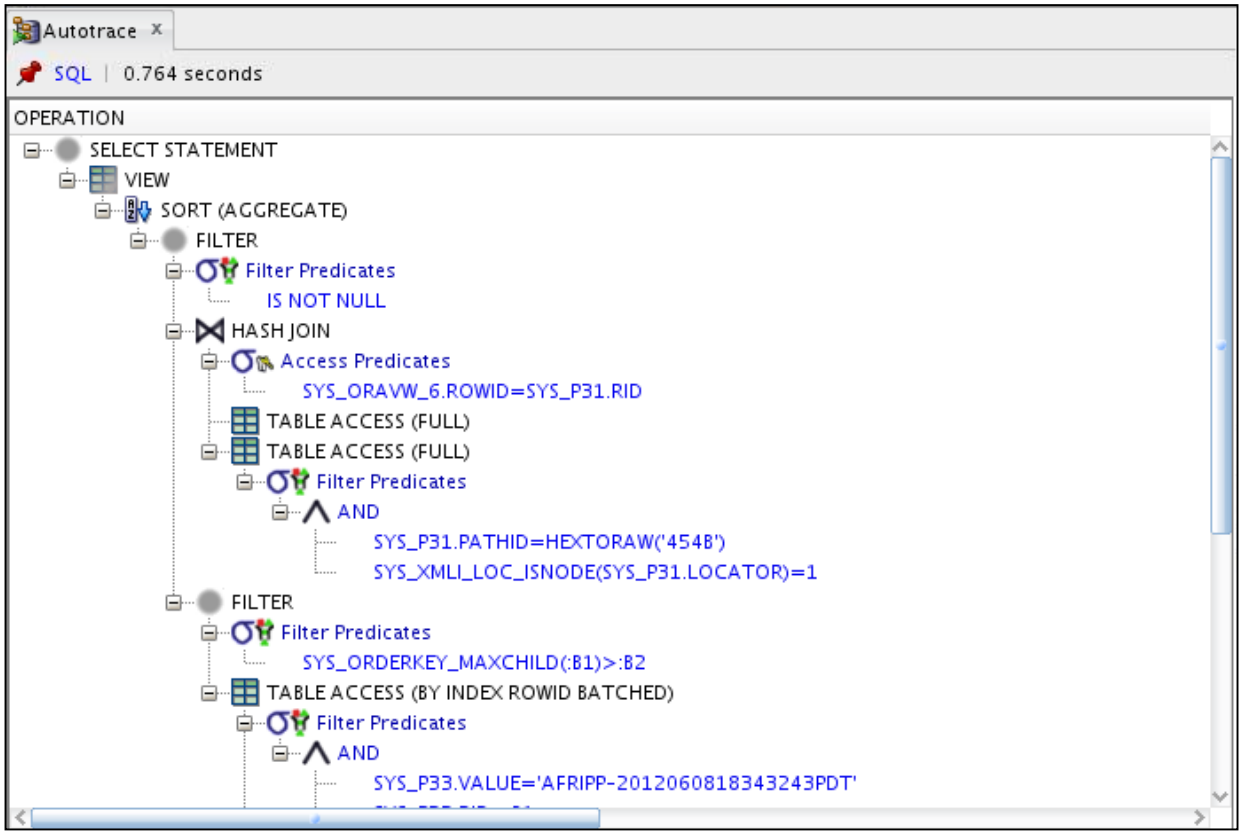
5. Use Autotrace to generate the execution plan for each of the four queries. The Autotrace output shows that the XML Index is used when the XQuery expression contains nodes that are included in the index. However, when the XQuery expression does not contain any of the nodes in the index, the explain plans show that the optimizer reverts to using streaming XPath evaluation.
6. **Query 1:**

```
--
-- A set of simple queries to demonstrate how indexing can optimize XQuery operations.
--
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[Reference/text()=
           passing 'AFRIPP-2012060818343243PDT' as "REFERENCE"
       )
/
```

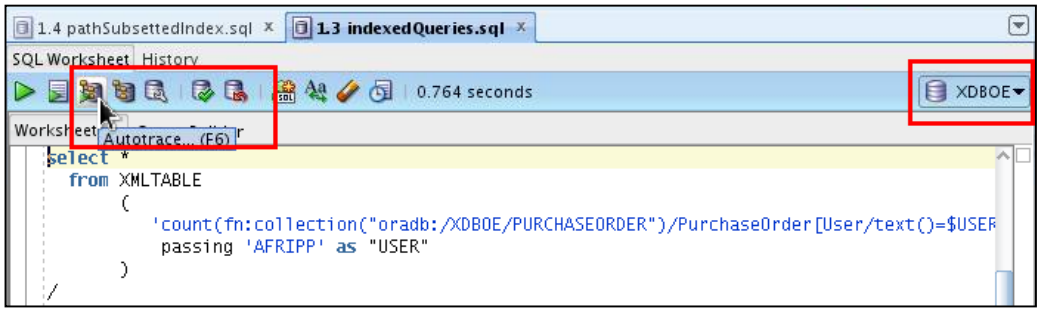
Oracle University and Error : You are not a Valid Partner use only

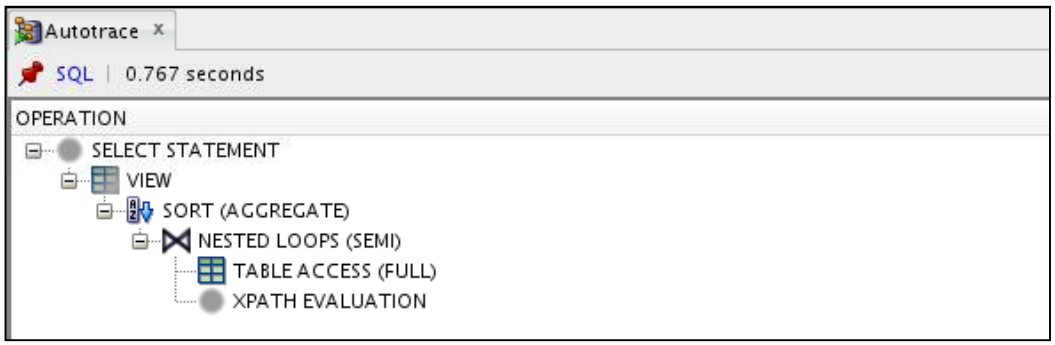
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



7. **Query 2:**



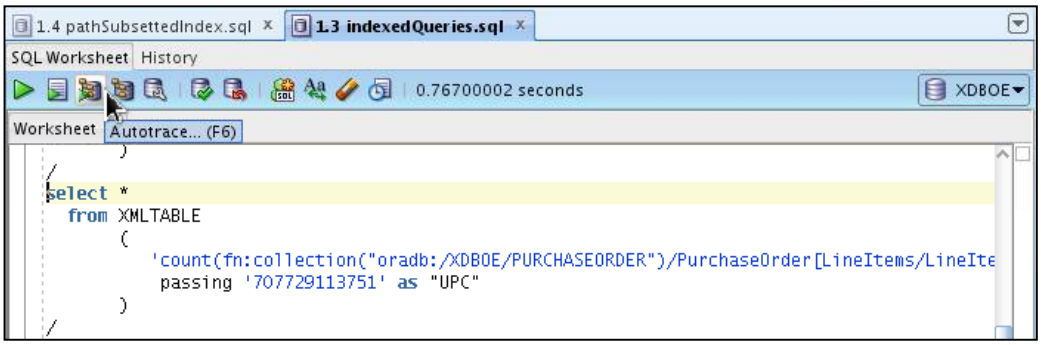


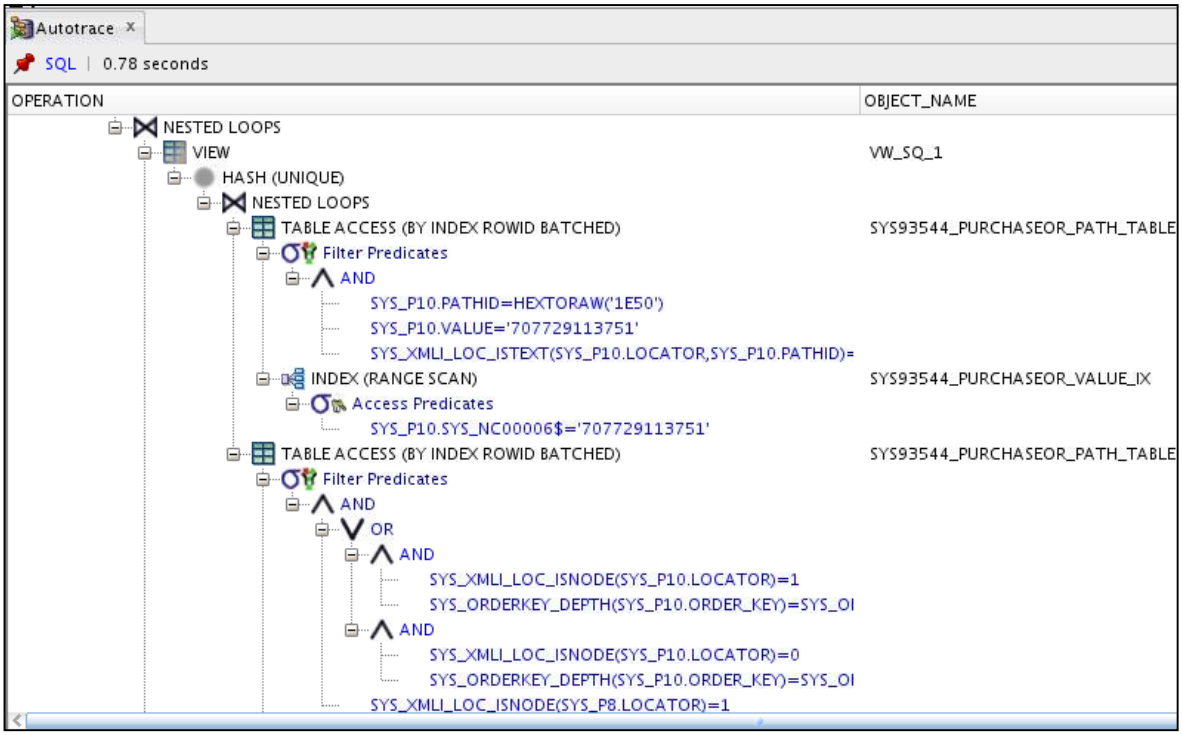
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

8. **Query 3:**

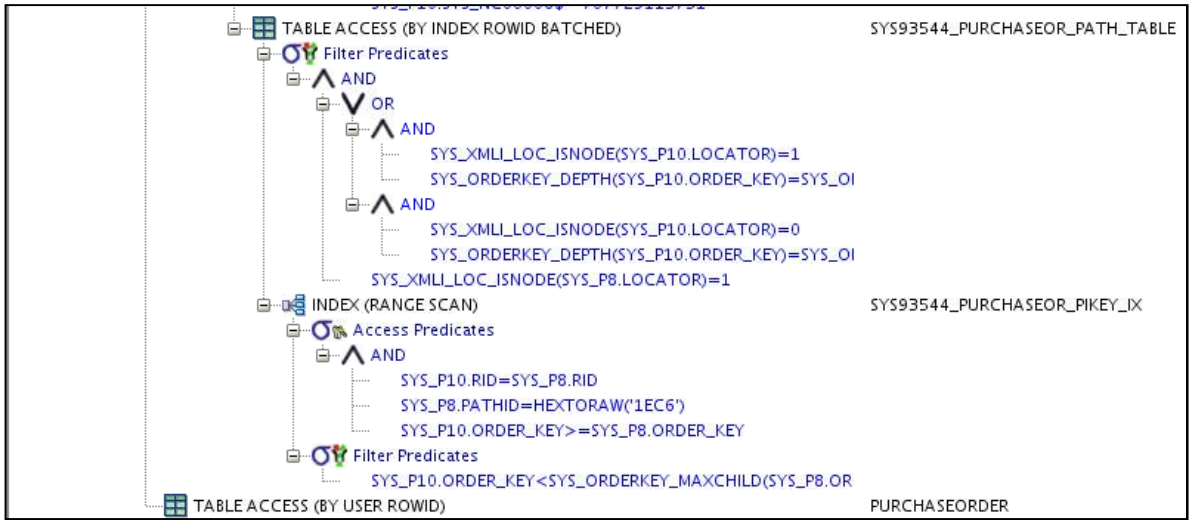




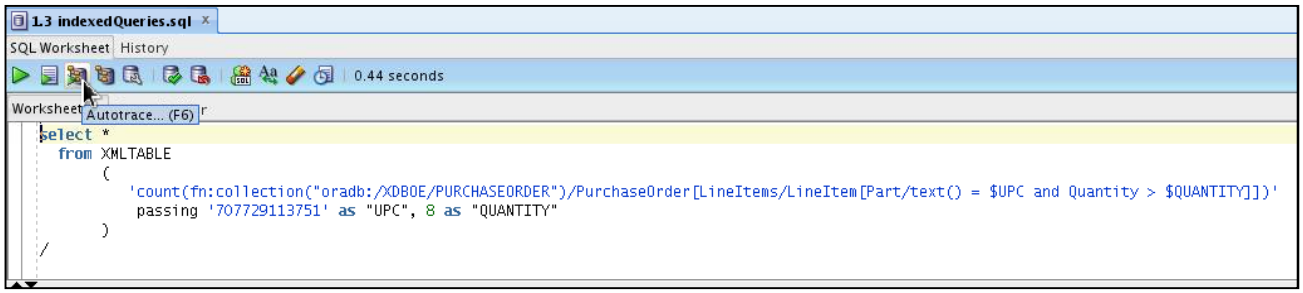
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



9. **Query 4:**



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Autotrace

SQL | 0.44 seconds

OPERATION	OBJECT_NAME
SELECT STATEMENT	
VIEW	
SORT (AGGREGATE)	
FILTER	
Filter Predicates	
IS NOT NULL	
TABLE ACCESS (FULL)	PURCHASEORDER
NESTED LOOPS	
NESTED LOOPS	
TABLE ACCESS (BY INDEX ROWID BATCHED)	SYS93544_PURCHASEOR_PATH_TABLE
Filter Predicates	
AND	
SYS_P11.PATHID=HEXTORAW('1E50')	
SYS_P11.VALUE='707729113751'	
SYS_P11.RID=:B1	
SYS_XMLI_LOC_ISTEXT(SYS_P11.LOCATOR,SYS_P11.PATHID)=1	
INDEX (RANGE SCAN)	SYS93544_PURCHASEOR_VALUE_IX
Access Predicates	
SYS_P11.SYS_NC00006$='707729113751'	
TABLE ACCESS (BY INDEX ROWID BATCHED)	SYS93544_PURCHASEOR_PATH_TABLE
Filter Predicates	
AND	
OR	
AND	
SYS_XMLI_LOC_ISNODE(SYS_P11.LOCATOR)=1	
SYS_ORDERKEY_DEPTH(SYS_P11.ORDER_KEY)=SYS_ORDE	
AND	
SYS_XMLI_LOC_ISNODE(SYS_P11.LOCATOR)=1	
SYS_ORDERKEY_DEPTH(SYS_P11.ORDER_KEY)=SYS_ORDE	
AND	
SYS_XMLI_LOC_ISNODE(SYS_P11.LOCATOR)=0	
SYS_ORDERKEY_DEPTH(SYS_P11.ORDER_KEY)=SYS_ORDE	
SYS_XMLI_LOC_ISNODE(SYS_P9.LOCATOR)=1	
INDEX (RANGE SCAN)	SYS93544_PURCHASEOR_PIKEY_IX
Access Predicates	
AND	
SYS_P11.RID=SYS_P9.RID	
SYS_P9.PATHID=HEXTORAW('1EC6')	
SYS_P9.ORDER_KEY<=SYS_P11.ORDER_KEY	
Filter Predicates	
AND	
SYS_P9.RID=:B1	
SYS_P11.ORDER_KEY<SYS_ORDERKEY_MAXCHILD(SYS_P9.OR	
XPATH EVALUATION	

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-6: Optimizing XQuery Performance by Using a Structured XMLIndex

Overview

The unstructured index is best suited for working with highly unstructured XML. In situations where the XML is highly structured, but no XML schema is available, or the XML is semi-structured, containing islands of structure floating in a sea of unstructured content, the structured XML Index can be used to optimize operations on the XML.

With a structured XML Index, one or more `XMLTable` operators are used to define which nodes in the document will be indexed. The selected nodes are projected into a series of relational tables that form the basis of the index. When the index has been created, the system will identify the XQuery expressions that can be evaluated using the index, and use the index appropriately. Secondary indexes can be created on the tables that make up the index, enabling further performance optimizations.

Structured indexes are very useful for optimizing XPath expressions that return a set of sibling nodes, or when the predicates in XQuery expressions reference sibling nodes. With an unstructured index, an independent index operation is required to find each of the nodes in question and then further processing is required to ensure that the selected nodes are indeed siblings. With a structured index, all of the nodes for a given XPath expression are stored in the same row and can be accessed in a single index operation, and no additional processing is required for sibling verification.

In this practice, you create a structured `XMLIndex`.

Assumptions

Tasks

1. Open the `lab_18_06_01.sql` script.

   Using the Files tab, right-click the `lab_18_06_01.sql` script and then select Open to display it in the SQL Worksheet area.

   This script creates a structured `XMLIndex`. The definition of the `XMLIndex` is provided by using `DBMS_XMLINDEX.createParameters`. Using a parameters clause simplifies the actual DDL needed to create the index. This is very useful when indexing complex XML structures, with multiple levels of nesting. Note how each XMLTable operator has a table name associated with it. The script also creates B-TREE indexes on the tables underlying the index to further optimize performance.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

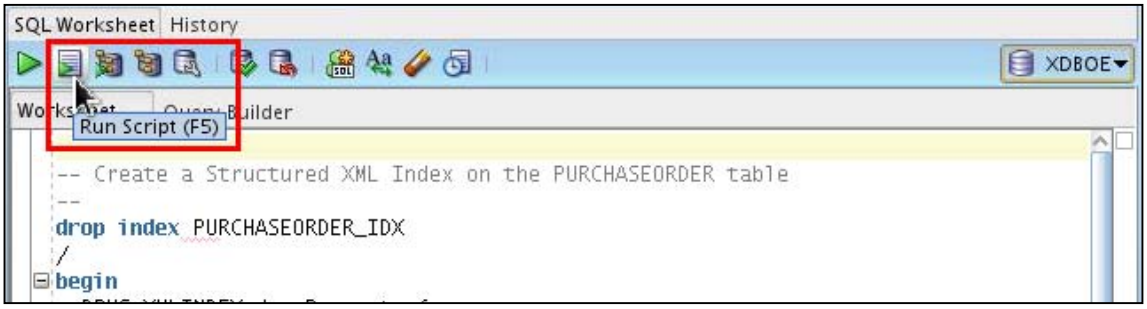
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

```
--
-- Create a Structured XML Index on the PURCHASEORDER table
--
drop index PURCHASEORDER_IDX
/
begin
  DBMS_XMLINDEX.dropParameter(
                 'PO_SXI_PARAMETERS');
end;
/
begin
  DBMS_XMLINDEX.registerParameter(
                 'PO_SXI_PARAMETERS',
                 'GROUP   PO_LINEITEM
                    xmlTable PO_INDEX_MASTER ''/PurchaseOrder''
                       COLUMNS
                         REFERENCE       varchar2(30)   PATH ''Reference/text
                         LINEITEM            xmlType    PATH ''LineItems/Line
                    VIRTUAL xmlTable PO_INDEX_LINEITEM ''/LineItem''
                       PASSING lineitem
                       COLUMNS
                         ITEMNO         number(38)     PATH ''@ItemNumber'',
                         UPC            number(14)     PATH ''Part/text()'',
                         DESCRIPTION    varchar2(256)  PATH ''Part/@Descripti
                 ');
end;
/
create index PURCHASEORDER_IDX
          on PURCHASEORDER (OBJECT_VALUE)
             indextype is XDB.XMLINDEX
             parameters ('PARAM PO_SXI_PARAMETERS')
/
create unique index REFERENCE_IDX
       on PO_INDEX_MASTER (REFERENCE)
/
create index UPC_IDX
       on PO_INDEX_LINEITEM (UPC)
/
call dbms_stats.gather_schema_stats(USER)
```

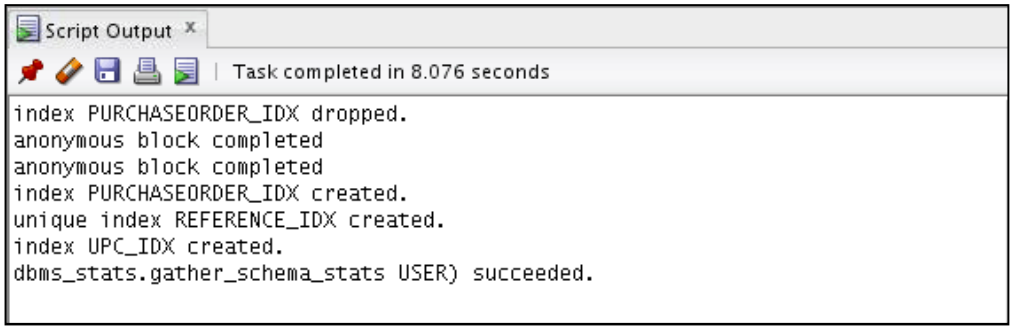
2. Run the script using the `XDBOE` database connection.

   Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon (or press the F5 key). The output is displayed in the Script Output tab.



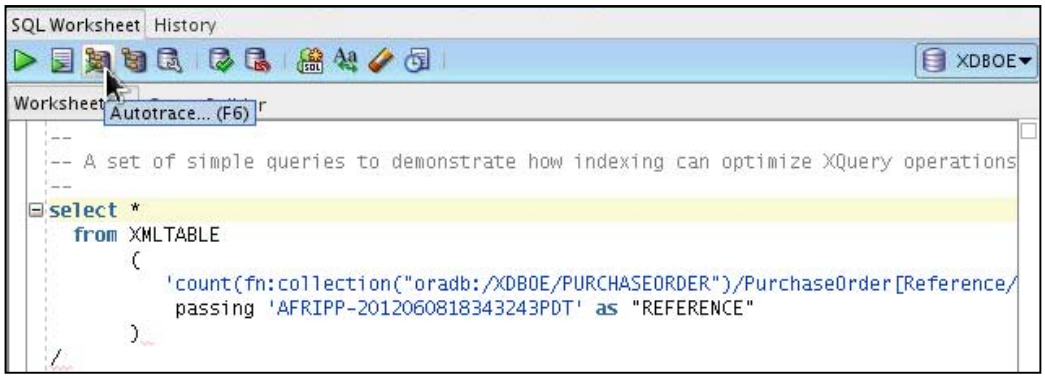
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

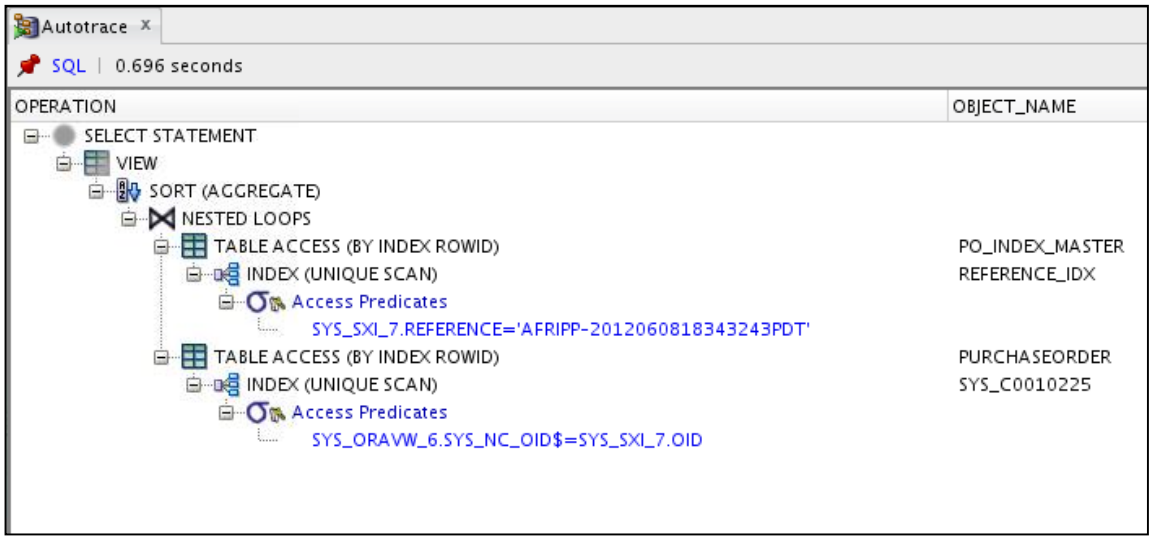
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



3. Return to the SQL worksheet containing the `lab_18_04_03.sql` script or re-open the `lab_18_04_03.sql` script. Use Autotrace to generate the execution plan for each of the four queries. The Autotrace output shows that the XML Index is used when the XQuery expression contains nodes that are included in the index. However, when the XQuery expression does not contain any of the nodes in the index, the explain plans show that the optimizer reverts to using streaming XPath evaluation. This section will only show the execution plans for the first two queries.

Query 1:





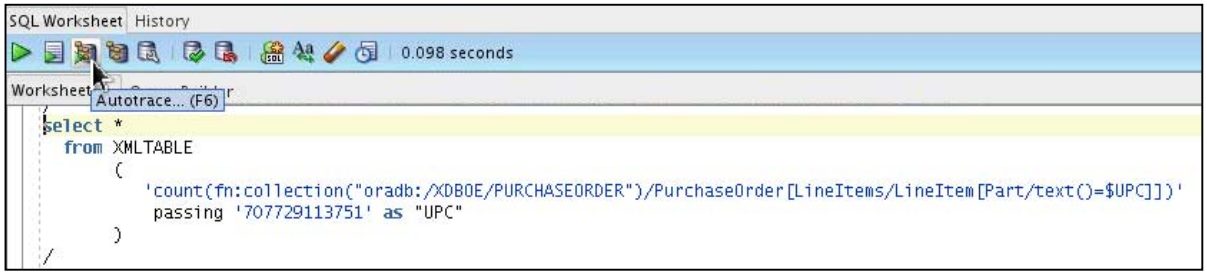
Oracle University and Error : You are not a Valid Partner use only

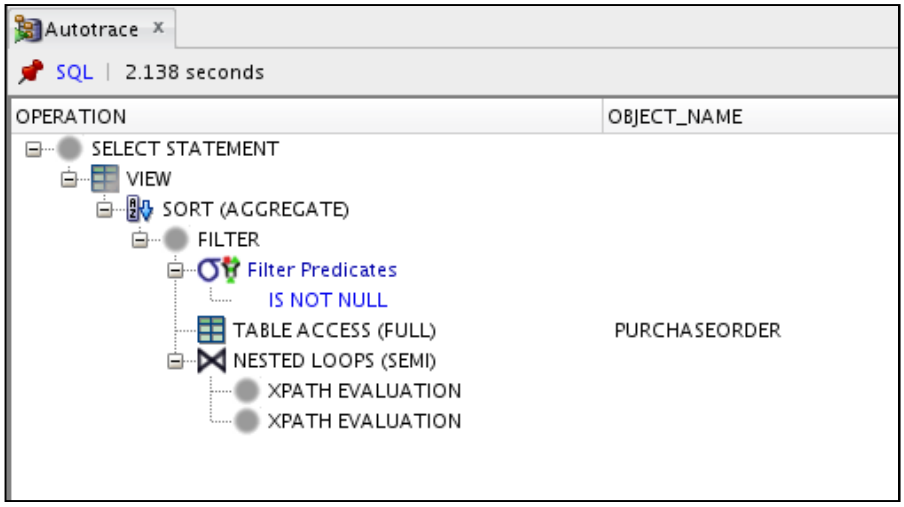
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Query 2:

Query 3:





Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Query 4:

2.13800001 seconds

Autotrace... (F6)

```
/
select *
  from XMLTABLE
       (
          'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[LineItems/LineItem[Part/text() = $UPC and Quantity > $QUANTITY]])'
           passing '707729113751' as "UPC", 8 as "QUANTITY"
       )
/
```

Autotrace

SQL | 2.314 seconds

OPERATION	OBJECT_NAME
SELECT STATEMENT	
VIEW	
SORT (AGGREGATE)	
FILTER	
Filter Predicates	
IS NOT NULL	
TABLE ACCESS (FULL)	PURCHASEORDER
NESTED LOOPS (SEMI)	
NESTED LOOPS	
XPATH EVALUATION	
XPATH EVALUATION	
XPATH EVALUATION	

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-7: Updating XML Content by Using XQuery-Update

Overview

XQuery-Update is an extension to W3C XQuery standard that makes it possible to update the content of XML documents. XQuery update operations can modify the values of existing nodes, replace a fragment of XML with another fragment of XML and insert and remove nodes from the document. Support for XQuery-Update is available in Oracle Database 11.2.0.3.0 and later. In Oracle XML DB XQuery operations are executed using the `XMLQuery` Operator. This practice provides a brief introduction to the XQuery-Update standard and shows you how to use the `XMLQuery()` SQL/XML function to perform XQuery-Update operations on XML content stored in the Oracle database. While it appears that the result of an XQuery-Update operation is always a new document, in the background the XQuery-Update is rewritten into a series of modifications to the data stored on disk. This enables partial update of the XML document, leading to significant savings in re-indexing and undo and redo generation.

The general form of an XQuery-update is:

```
UPDATE tablename
SET XML = XMLQuery(XQuery-Update operation)
WHERE XMLExists()
```

The `XMLExists` operator is used to determine the documents to which the XQuery-Update operation is applied. Predicates supplied to the XQuery-Update operation will determine which nodes within the documents selected by the `XMLExists` operation are actually updated.

In this practice, you use the XQuery-Update feature.

Assumptions

Tasks

1. Open `the lab_18_07_01.sql` script. The query (`INITIAL_STATE`) generates a summary of the existing state of the document that is the target of the XQuery-Update operations. Replace **<TODO>** with the required `XMLExists` syntax, which uses a predicate on the Reference element to check if the value of the text node associated with the Reference element is '`AFRIPP-20120430212831873PDT`'.

```
select XMLQUERY(
         '<POSummary
lineItemCount="{count($XML/PurchaseOrder/LineItems/LineItem)}">{
            $XML/PurchaseOrder/User,
            $XML/PurchaseOrder/Requestor,
            $XML/PurchaseOrder/LineItems/LineItem[2]
          }
          </POSummary>'
         passing object_value as "XML"
         returning CONTENT
       ) INITIAL_STATE
  from PURCHASEORDER
 where XMLExists(
         '$XML/PurchaseOrder[Reference=$REF]'
          passing object_value as "XML",
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

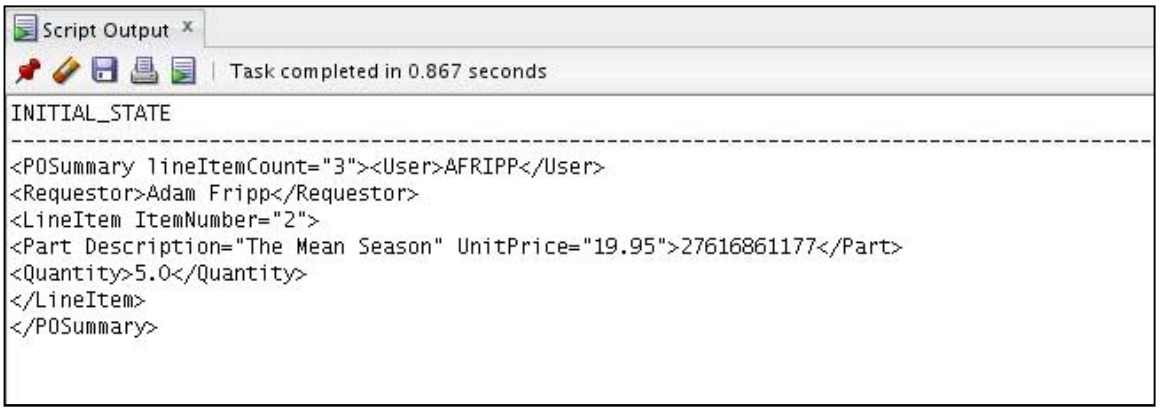
Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
                    'AFRIPP-20120430212831873PDT' as "REF"
        )
/
```

2. Run the script by using the `XDBOE` database connection.

   Select the `XDBOE` database connection from the Choose Db Connection drop-down list and then click the Run Script icon (or press the F5 key).



3. Open and run the `lab_18_07_03.sql` script. This XQuery-Update operation replaces the values of existing scalar nodes. Note how a predicate is used to determine which of the three description nodes in the document will be updated.
   a. Replace **<TODO1>** with the appropriate syntax to replace the `User` node with the value passed as `USERID`.
   b. Replace **<TODO2>** with the appropriate syntax to replace the `Requestor` node with the value passed as `FULLNAME`.
   c. Replace **<TODO3>** with the appropriate syntax to replace the `Description` attribute of `LineItems/LineItem/Part` that has a `Description` attribute value equal to the value passed as `OLDTITLE` with the value passed as `NEWTITLE`.

```
update PURCHASEORDER
   set object_value =
     XMLQuery('copy $NEWXML := $XML modify (
                for $PO in $NEWXML/PurchaseOrder return (
                replace value of node $PO/User with $USERID,
                replace value of node $PO/Requestor with $FULLNAME,
                replace value of node
$PO/LineItems/LineItem/Part[@Description=$OLDTITLE]/@Description
with $NEWTITLE )
                 )
                        return $NEWXML'
                        passing object_value as "XML",
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
                          'KCHUNG' as "USERID",
                          'Kelly Chung' as "FULLNAME",
                          'The Mean Season' as "OLDTITLE",
                          'The Wizard of Oz' as "NEWTITLE"
                          returning content
                        )
 where xmlExists(

'$XML/PurchaseOrder[Reference=$REF]/LineItems/LineItem/Part[@Descrip
tion=$OLDTITLE]'
          passing object_value as "XML",
                  'AFRIPP-20120430212831873PDT' as "REF",
                  'The Mean Season' as "OLDTITLE"
       )
/
```

```
1 rows updated.
```

4. Open and run the `lab_18_07_04.sql` script. This XQuery operation (`UPDATED_NODES`) generates a summary showing the state of the document after the XQuery operation has been completed.

```
select XMLQUERY(
         '<POSummary
lineItemCount="{count($XML/PurchaseOrder/LineItems/LineItem)}">{
            $XML/PurchaseOrder/User,
            $XML/PurchaseOrder/Requestor,
            $XML/PurchaseOrder/LineItems/LineItem[2]
          }
          </POSummary>'
         passing object_value as "XML"
         returning CONTENT
       ) UPDATED_NODES
  from PURCHASEORDER
 where xmlExists(
         '$XML/PurchaseOrder[Reference=$REF]'
          passing object_value as "XML",
                  'AFRIPP-20120430212831873PDT' as "REF"
       )
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
Script Output
Task completed in 0.023 seconds
UPDATED_NODES
--------------------------------------------------------------------------------
<POSummary lineItemCount="3"><User>KCHUNG</User>
<Requestor>Kelly Chung</Requestor>
<LineItem ItemNumber="2">
<Part Description="The Wizard of Oz" UnitPrice="19.95">27616861177</Part>
<Quantity>5.0</Quantity>
</LineItem>
</POSummary>
```

5. Open and run the `lab_18_07_05.sql` script by using the `XDBOE` database connection. Replace **<TODO>** with the appropriate syntax to delete the node where the `ItemNumber` attribute value of `/PurchaseOrder/LineItems/LineItem` equals to the value passed as `ITEMNO`. Use a predicate to determine which of the `LineItem` nodes in the document will be deleted.

```
update PURCHASEORDER
   set object_value = XMLQuery(
                        'copy $NEWXML := $XML modify (
                                                 delete nodes
$NEWXML/PurchaseOrder/LineItems/LineItem[@ItemNumber=$ITEMNO]
                                                 )
                         return $NEWXML'
                        passing object_value as "XML", 2 as ITEMNO
                        returning CONTENT
                      )
 where xmlExists(
         '$XML/PurchaseOrder[Reference=$REF]'
          passing object_value as "XML",
                  'AFRIPP-20120430212831873PDT' as "REF"
       )
/
```

```
1 rows updated.
```

6. Write the query to confirm that the document from the previous step is deleted.

```
select XMLQUERY(
         '<POSummary
lineItemCount="{count($XML/PurchaseOrder/LineItems/LineItem)}">{
            $XML/PurchaseOrder/LineItems/LineItem[2]
          }
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
          </POSummary>'
         passing object_value as "XML"
         returning CONTENT
       ) DELETED_NODE
  from PURCHASEORDER
 where xmlExists(
         '$XML/PurchaseOrder[Reference=$REF]'
          passing object_value as "XML",
                  'AFRIPP-20120430212831873PDT' as "REF"
       )
/
```

Before the delete operation:

```
Script Output
Task completed in 0.867 seconds
INITIAL_STATE
--------------------------------------------------------------------------------
<POSummary lineItemCount="3"><User>AFRIPP</User>
<Requestor>Adam Fripp</Requestor>
<LineItem ItemNumber="2">
<Part Description="The Mean Season" UnitPrice="19.95">27616861177</Part>
<Quantity>5.0</Quantity>
</LineItem>
</POSummary>
```

After the delete operation:

```
Script Output
Task completed in 0.039 seconds
DELETED_NODE
--------------------------------------------------------------------------------
<POSummary lineItemCount="2"><LineItem ItemNumber="3">
<Part Description="Irma La Douce" UnitPrice="19.95">27616865908</Part>
<Quantity>4.0</Quantity>
</LineItem>
</POSummary>
```

7. Open, update, and then run the `lab_18_07_07.sql` script by using the `XDBOE` database connection. Use XQuery-Update to insert a new `LineItem` node into the referenced document. Write an XQuery-Update operation to insert a new `LineItem` node into the document. Use a predicate to determine where the new `LineItem` node will be placed in the document. Insert the new `LineItem` node after `/PurchaseOrder/LineItems/LineItem` where the `ItemNumber` attribute for the `LineItem ="3"`.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
update PURCHASEORDER
   set object_value = XMLQuery(
     'copy $NEWXML := $XML modify (
     for $TARGET in
$NEWXML/PurchaseOrder/LineItems/LineItem[@ItemNumber="3"]
    return insert node $LINEITEM after $TARGET
)
    return $NEWXML'
    passing object_value as "XML",
    xmlType('<LineItem ItemNumber="4">
             <Part Description="Rififi"
              UnitPrice="29.95">37429155622
             </Part>
             <Quantity>2</Quantity>
             </LineItem>') as "LINEITEM"
             returning CONTENT
                     )
 where xmlExists(
         '$XML/PurchaseOrder[Reference=$REF]'
          passing object_value as "XML",
                  'AFRIPP-20120430212831873PDT' as "REF"
       )
/
```

```
1 rows updated.
```

8. Open and run the `lab_18_07_08.sql` script by using the `XDBOE` database connection. The XQuery operation (`INSERTED_NODE`) shows that the document has been updated. Because the new `LineItem` element, (which has an `ItemNumber` attribute of 4) was placed after the `LineItem` element with an `ItemNumber` attribute of `3`, the new `LineItem` is the third `LineItem` element.

```
select XMLQUERY(
         '<POSummary
lineItemCount="{count($XML/PurchaseOrder/LineItems/LineItem)}">{
            $XML/PurchaseOrder/LineItems/LineItem[3]
          }
          </POSummary>'
         passing object_value as "XML"
         returning CONTENT
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
        ) INSERTED_NODE
  from PURCHASEORDER
 where xmlExists(
         '$XML/PurchaseOrder[Reference=$REF]'
          passing object_value as "XML",
                  'AFRIPP-20120430212831873PDT' as "REF"
        )
/
```

```
Script Output
Task completed in 0.848 seconds
INSERTED_NODE
--------------------------------------------------------------------------------
<POSummary lineItemCount="3"><LineItem ItemNumber="4">
<Part Description="Rififi" UnitPrice="29.95">37429155622</Part>
<Quantity>2</Quantity>
</LineItem>
</POSummary>

1 rows updated.
```

9. Open and run the `lab_18_07_09.sql` script by using the `XDBOE` database connection. The XQuery-Update statement updates a fragment. It replaces the entire `LineItems` element with a new `LineItems` element. Revert the changes made to the `UserID` and `Requestor` elements. **Of course, because the Oracle Database provides complete transaction control as part of the SQL language, the entire setup of updates could also have been undone by issuing a rollback command**.

```
update PURCHASEORDER
   set object_value = XMLQuery(
                        ' copy $NEWXML := $XML modify (
                             for $PO in $NEWXML/PurchaseOrder return
(
                                  replace value of node $PO/User
with $USERID,
                                  replace value of node
$PO/Requestor with $FULLNAME,
                                  replace node $PO/LineItems with
$LINEITEMS
                             )
                         )
                         return $NEWXML'
                         passing object_value as "XML",
                                 'AFRIPP' as "USERID",
                                 'Adam Fripp' as "FULLNAME",
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
                                    xmlType(
                                      '<LineItems>
                                         <LineItem ItemNumber="1">
                                           <Part Description="The
Secret of Roan Inish" UnitPrice="19.95">43396509290</Part>
                                           <Quantity>7</Quantity>
                                         </LineItem>
                                         <LineItem ItemNumber="2">
                                           <Part Description="The Mean
Season" UnitPrice="19.95">27616861177</Part>
                                           <Quantity>5</Quantity>
                                         </LineItem>
                                         <LineItem ItemNumber="3">
                                           <Part Description="Irma La
Douce" UnitPrice="19.95">27616865908</Part>
                                           <Quantity>4</Quantity>
                                         </LineItem>
                                       </LineItems>'
                                    ) as "LINEITEMS"
                                    returning content
                      )
 where xmlExists(
         '$XML/PurchaseOrder[Reference=$REF]'
          passing object_value as "XML",
                  'AFRIPP-20120430212831873PDT' as "REF"
       )
/
```

```
1 rows updated.
```

10. Write a query, (`FINAL_STATE`), to confirm that the document is back in its original state.

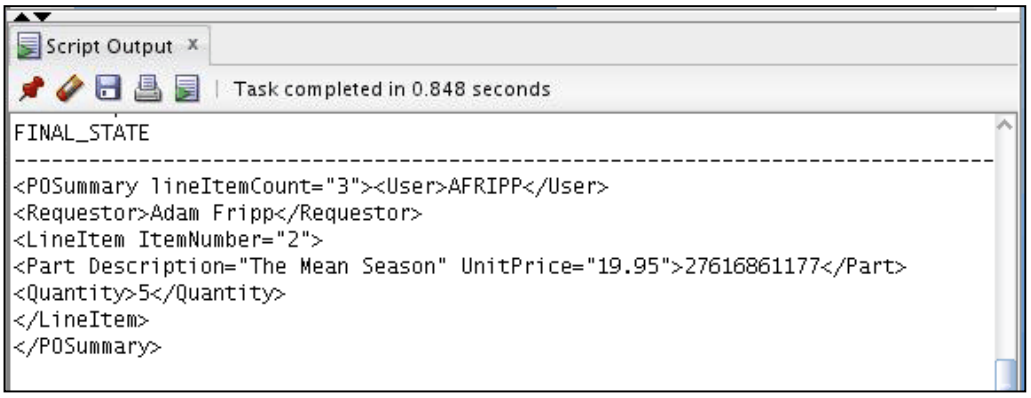
```
select XMLQUERY(
         '<POSummary
lineItemCount="{count($XML/PurchaseOrder/LineItems/LineItem)}">{
            $XML/PurchaseOrder/User,
            $XML/PurchaseOrder/Requestor,
            $XML/PurchaseOrder/LineItems/LineItem[2]
          }
          </POSummary>'
         passing object_value as "XML"
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
          returning CONTENT
        ) FINAL_STATE
  from PURCHASEORDER
 where xmlExists(
          '$XML/PurchaseOrder[Reference=$REF]'
           passing object_value as "XML",
                   'AFRIPP-20120430212831873PDT' as "REF"
        )
/
```



You can also see the content of the entire document by replacing `LineItem[2]` with `LineItem`.

```
FINAL_STATE
-----------------------------------------------------------------------------------------
<POSummary lineItemCount="3"><User>AFRIPP</User>
<Requestor>Adam Fripp</Requestor>
<LineItem ItemNumber="1">
<Part Description="The Secret of Roan Inish" UnitPrice="19.95">43396509290</Part>
<Quantity>7</Quantity>
</LineItem>
<LineItem ItemNumber="2">
<Part Description="The Mean Season" UnitPrice="19.95">27616861177</Part>
<Quantity>5</Quantity>
</LineItem>
<LineItem ItemNumber="3">
<Part Description="Irma La Douce" UnitPrice="19.95">27616865908</Part>
<Quantity>4</Quantity>
</LineItem>
</POSummary>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-8: Searching XML Content by Using XQuery Full Text

Overview

XQuery Full Text is an extension to W3C XQuery standard that makes it perform complex full-text style search operations on the content of XML documents. Support for XQuery Full Text is available in Oracle Database 12.1.0.1.0 and later. The original XQuery language included an operator called `CONTAINS` that allow pattern-matching-based searches on XML content. The `CONTAINS` operator provides a case-sensitive substring-style search capability; that is, it returns `TRUE` if the source string contains exactly the set of characters contained in the target string. The XQuery Full Text specification adds the ability to perform word-based searches of XML content, with all of the common features of a text-retrieval system. XQuery Full Text includes support for word match, windowing (word must appear within *n* words of word), and stemming (automatically recognize related words). In order to provide an efficient implementation of XQuery Full Text, Oracle Database 12*c* makes use of Oracle's Text indexing technology. In order to make use of XQuery Full Text it is necessary to create an XML-aware full-text index on the documents that are to be searched.

In this practice, you create an XML-aware full-text index.

Assumptions

Tasks

1. Create an XML-aware full-text index by using the information in the following two tables. The first step is to define the section group and storage preferences that will be used by the index. These items are managed by using methods provided by the package `CTX_DLL`. In order to use this package, the `XBBOE` user must have been granted the role `CTXAPP`. The second step is to create a `CTXSYS.CONTEXT` index based on the section group and storage preferences.

Procedure	Parameters
`CREATE_SECTION_GROUP`	`XQFT`, `PATH_SECTION_GROUP`
`SET_SEC_GRP_ATTR`	`XQFT`, `XML_ENABLE`, `TRUE`
`create_preference`	`STORAGE_PREFS`, `BASIC_STORAGE`
`set_attribute`	`STORAGE_PREFS, D_TABLE_CLAUSE,` `LOB(DOC) STORE AS SECUREFILE (COMPRESS MEDIUM CACHE)'`
`set_attribute`	`STORAGE_PREFS, I_TABLE_CLAUSE,` `LOB(TOKEN_INFO) STORE AS SECUREFILE (NOCOMPRESS CACHE)`

Index Name	`PURCHASEORDER_XQFT_IDX`
Index Type	`CTXSYS.CONTEXT`
Index Parameters	`storage STORAGE_PREFS` `section group XQFT`

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

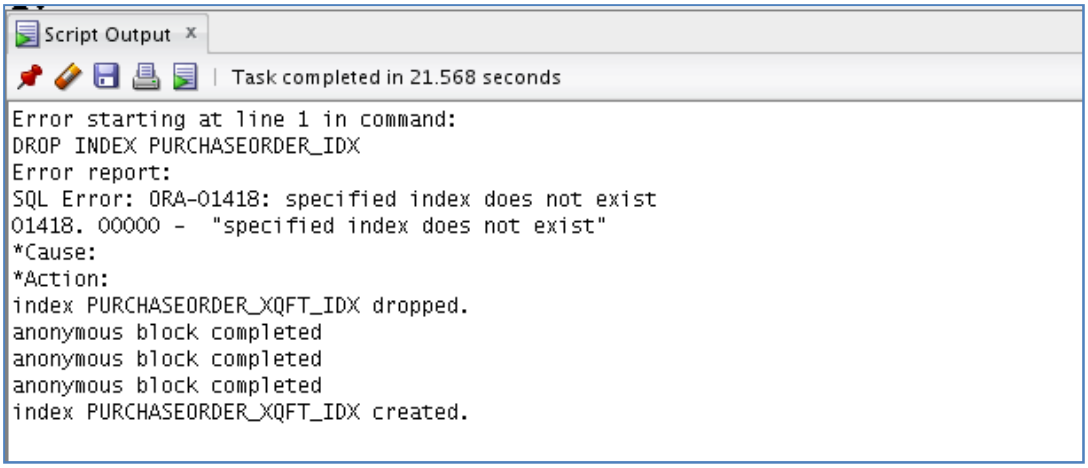
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
DROP INDEX PURCHASEORDER_IDX
/
begin
  CTX_DDL.DROP_SECTION_GROUP('XQFT');
end;
/
begin
  CTX_DDL.DROP_PREFERENCE('STORAGE_PREFS');
end;
/
begin
  CTX_DDL.CREATE_SECTION_GROUP('XQFT','PATH_SECTION_GROUP');
  CTX_DDL.SET_SEC_GRP_ATTR('XQFT','XML_ENABLE','T');

  CTX_DDL.create_preference('STORAGE_PREFS', 'BASIC_STORAGE');
  CTX_DDL.set_attribute(
     'STORAGE_PREFS',
     'D_TABLE_CLAUSE',
     'LOB(DOC) STORE AS SECUREFILE (COMPRESS MEDIUM CACHE)'
  );
  CTX_DDL.set_attribute(
     'STORAGE_PREFS',
     'I_TABLE_CLAUSE',
     'LOB(TOKEN_INFO) STORE AS SECUREFILE (NOCOMPRESS CACHE)'
  );
END;
/
DROP INDEX PURCHASEORDER_XQFT_IDX
/
CREATE INDEX PURCHASEORDER_XQFT_IDX
    ON PURCHASEORDER(OBJECT_VALUE)
       INDEXTYPE IS CTXSYS.CONTEXT
       PARAMETERS(
         'storage STORAGE_PREFS
         section group XQFT'
       )
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



When the index has been created, it can be used to optimize XQuery Full Text operations. Currently XQuery Full Text searches must always be performed via the `XMLExists` operator, to ensure that the optimizer uses the Full-Text index to locate which documents match the specified search conditions. As per the XQuery standard, by default all XQuery Full Text operations in Oracle Database 12*c* are case-insensitive.

2. Open, examine, and run the `lab_18_08_02.sql` script by using the `XDBOE` database connection. This script performs XQuery Full Text searches on the XML content that is stored in Oracle XML DB.

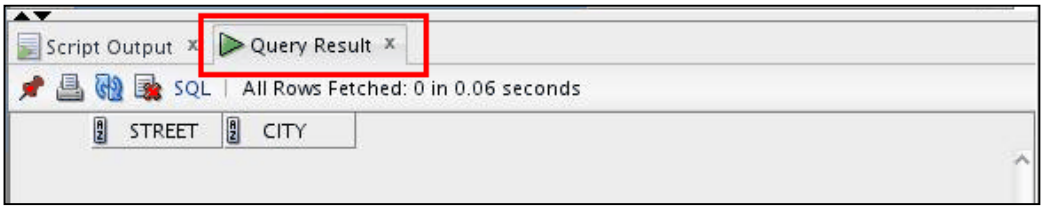
   Using the Files tab, right-click the `lab_18_08_02.sql` script, and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon to execute the query. The output is displayed in the Query Result tab. This query performs a search for an exact match on a phrase. The index cannot be used because the comparison is case-sensitive. No results are returned because the source is in mixed case and the target is in uppercase.

```
select distinct STREET, CITY
  from PURCHASEORDER,
       XMLTABLE(
         '$P/PurchaseOrder/ShippingInstructions/Address'
         passing OBJECT_VALUE as "P"
         COLUMNS
         STREET VARCHAR2(64) PATH 'street',
         CITY   VARCHAR2(32) PATH 'city'
       )
 where XMLExists(

'$P/PurchaseOrder/ShippingInstructions/Address[city=$PHRASE]'
         passing OBJECT_VALUE as "P",
                 'OXFORD' as "PHRASE"
       )
/
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

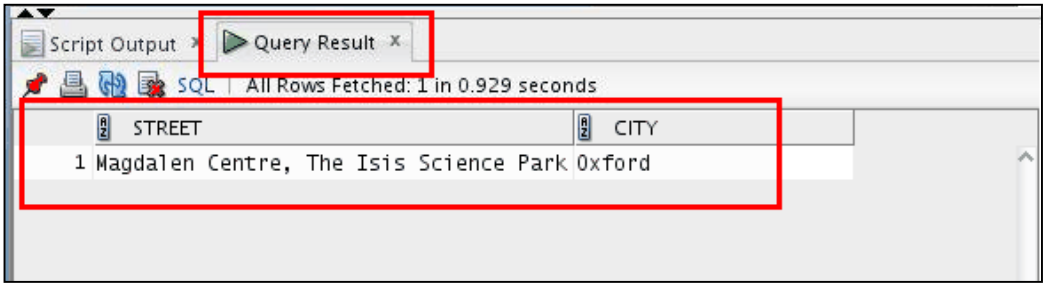


3. Open, examine, and run the `lab_18_08_03.sql` script by using the `XDBOE` database connection. This query searches for an exact match on a phrase.

   Using the Files tab, right-click the `lab_18_08_03.sql` script and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Statement icon to execute the query. The output is displayed in the Query Result tab.

   This query searches for an exact match on a phrase. The index cannot be used because the comparison is also case-sensitive; However, results are returned because the source and target are an exact match.

```
select distinct STREET, CITY
  from PURCHASEORDER,
       XMLTABLE(
         '$P/PurchaseOrder/ShippingInstructions/Address'
         passing OBJECT_VALUE as "P"
         COLUMNS
         STREET VARCHAR2(64) PATH 'street',
         CITY   VARCHAR2(32) PATH 'city'
       )
 where XMLExists(

'$P/PurchaseOrder/ShippingInstructions/Address[city=$PHRASE]'
         passing OBJECT_VALUE as "P",
                 'Oxford' as "PHRASE"
       )
/
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Open, examine, and run the `lab_18_08_04.sql` script by using the `XDBOE` database connection. The query uses the XQuery `contains()` operator to search for an exact match on a phrase.

   Using the Files tab, right-click the `lab_18_08_04.sql` script, and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon to execute the query. The output is displayed in the Query Result tab. The query uses the XQuery `contains()` operator to search for an exact match on a phrase. The index is not used because the comparison is case-sensitive, and `contains` performs a substring-type match, searching for the target string anywhere in the specified source. For word searches, this leads to false positives when a word in the source contains the target string. Case sensitivity also leads to false negatives where the target appears in the specified source but in a different case from the case used to define the target.

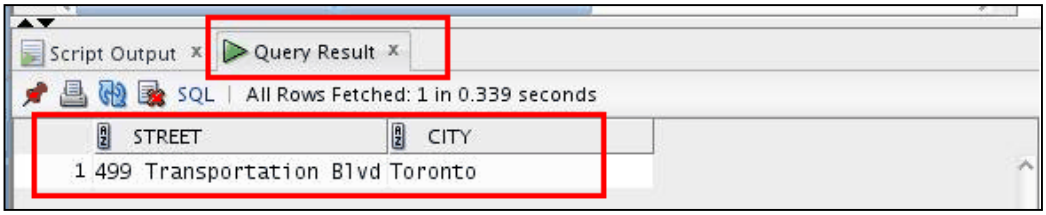
   There are a number of limitations to this kind of search. The `contains()` operator performs a substring-style of match, searching for the target string anywhere in the specified source. This kind of operation cannot easily be optimized by using any index. When thinking in text-retrieval terms, this kind of search often leads to false positives when a word in the source contains the target string. The case-sensitive nature of the `contains()` operator also leads to false negatives unless the programmer takes care to ensure that comparison of source and target is performed in a non-case-sensitive manner. As can be seen in the results, a search for the word '`sport`' has returned documents that contain the term '`transportation`' and omitted documents that contain the word '`Sporting`'.

```
select distinct STREET, CITY
  from PURCHASEORDER,
       XMLTABLE(
         '$P/PurchaseOrder/ShippingInstructions/Address'
         passing OBJECT_VALUE as "P"
         COLUMNS
         STREET VARCHAR2(64) PATH 'street',
         CITY   VARCHAR2(32) PATH 'city'
       )
 where XMLExists(

'$P/PurchaseOrder/ShippingInstructions/Address/street[contains(.,$PH
RASE)]'
         passing OBJECT_VALUE as "P",
                 'sport' as "PHRASE"
       )
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



5. Open, examine, and run the `lab_18_08_05.sql` script by using the `XDBOE` database connection. This query uses the XQuery Full Text `contains text` operation instead of the `contains()` operator. The `contains text` operation searches for the target as a word in the source.

   Using the Files tab, right-click the `lab_18_08_05.sql` script, and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon to execute the query. The output is displayed in the Query Result tab.

   This query uses the XQuery Full Text `contains text` operation instead of the `contains()` operator. The `contains text` operation searches for the target as a word in the source. Because the comparison is now based on words, rather than strings, a text index can be used to optimize the search. In this case, there are no results because neither `transportation` nor `Sporting` are an exact match for the phrase `sport`.

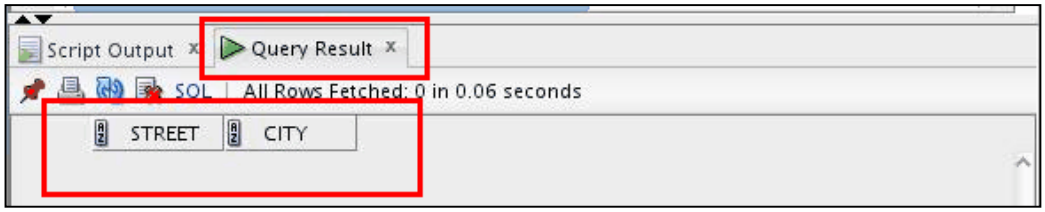
   The index is used because `contains text` comparisons are non-case-sensitive. However, because `contains text` is an exact word search, no results are found.

```
select distinct STREET, CITY
  from PURCHASEORDER,
       XMLTABLE(
         '$P/PurchaseOrder/ShippingInstructions/Address'
         passing OBJECT_VALUE as "P"
         COLUMNS
         STREET VARCHAR2(64) PATH 'street',
         CITY   VARCHAR2(32) PATH 'city'
       )
 where XMLExists(
         '$P/PurchaseOrder/ShippingInstructions/Address/street[.
contains text {$PHRASE}]'
         passing OBJECT_VALUE as "P",
                 'sport' as "PHRASE"
       )
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

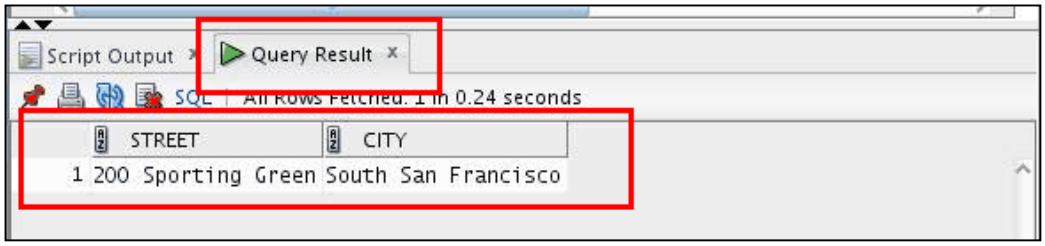


6. Open, examine, and run the `lab_18_08_06.sql` script by using the `XDBOE` database connection. This query uses the XQuery Full Text `using stemming` operation. The `using stemming` operation searches for any word related to the target as a word in the source.

   Using the Files tab, right-click the `lab_18_08_06.sql` script, and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon to execute the query. The output is displayed in the Query Result tab.

   This query uses the XQuery Full Text `using stemming` operation. The `using stemming` operation searches for any word related to the target as a word in the source. Because the comparison is now based on related words rather than exact words, the query returns documents that contain the word `Sporting` because `sport` is treated as a stem of `Sporting`.

```
select distinct STREET, CITY
  from PURCHASEORDER,
       XMLTABLE(
         '$P/PurchaseOrder/ShippingInstructions/Address'
         passing OBJECT_VALUE as "P"
         COLUMNS
         STREET VARCHAR2(64) PATH 'street',
         CITY   VARCHAR2(32) PATH 'city'
       )
 where XMLExists(
         '$P/PurchaseOrder/ShippingInstructions/Address/street[.
contains text {$PHRASE} using stemming]'
         passing OBJECT_VALUE as "P",
                 'sport' as "PHRASE"
       )
/
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

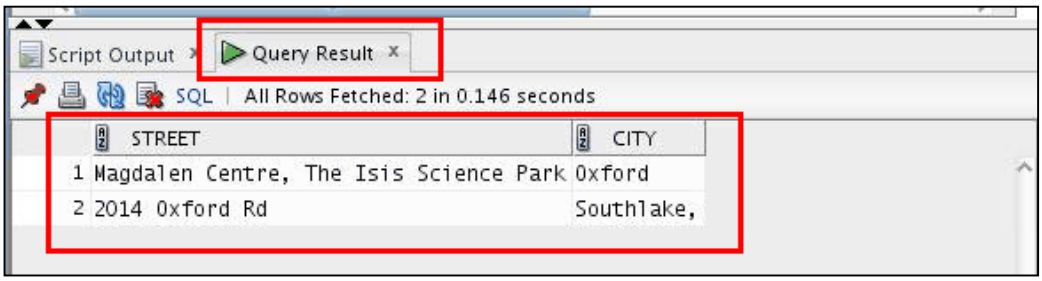
An XQuery Full Text `contains text` searches on a phrase with stemming. In this example, the index is used because `contains text` comparisons are non-case-sensitive. Results are returned because stemming identified that `sport` is a stem for `sporting`.

7. Open, examine, and run the `lab_18_08_07.sql` script by using the `XDBOE` database connection. This query demonstrates how you can use XQuery Full Text to search fragments as well as leaf-level nodes. In this query, the complex element `Address` is being searched for the word `Oxford`.

   Using the Files tab, right-click the `lab_18_08_07.sql` script, and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon to execute the query. The output is displayed in the Query Result tab.

   This query demonstrates how you can use XQuery Full Text to search fragments, as well as leaf-level nodes. In this query the complex element `Address` is being searched for the word `Oxford`. The query returns documents where any child of `Address` contains the target word. In this example the elements `street` and `city` contain the word `Oxford`.

```
select distinct STREET, CITY
  from PURCHASEORDER,
       XMLTABLE(
         '$P/PurchaseOrder/ShippingInstructions/Address'
         passing OBJECT_VALUE as "P"
         COLUMNS
         STREET VARCHAR2(64) PATH 'street',
         CITY   VARCHAR2(32) PATH 'city'
       )
 where XMLExists(
         '$P/PurchaseOrder/ShippingInstructions/Address[. contains
text {$PHRASE}]'
         passing OBJECT_VALUE as "P",
                 'Oxford' as "PHRASE"
       )
/
```



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

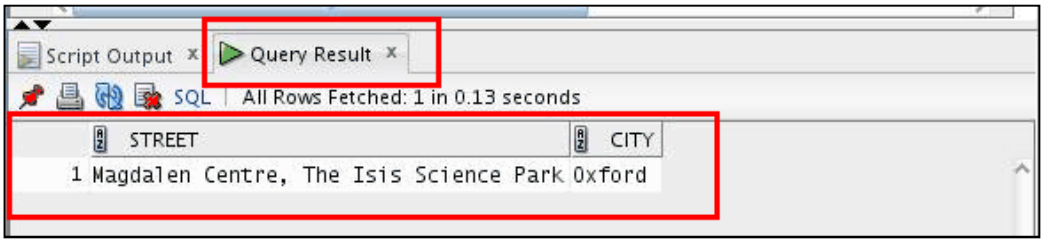
The index is used in this example because `contains text` comparisons are non-case-sensitive.

8. Open, examine, and run the `lab_18_08_08.sql` script by using the `XDBOE` database connection. This query demonstrates the use of the `ftand` operator to search for the presence of two words. The words do not need to be adjacent and do not need to be in any particular order.

   Using the Files tab, right-click the `lab_18_08_08.sql` script, and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon to execute the query. The output is displayed in the Query Result tab.

   This query demonstrates the use of the `ftand` operator to search for the presence of two words. The words do not need to be adjacent and do not need to be in any particular order.

```
select distinct STREET, CITY
  from PURCHASEORDER,
       XMLTABLE(
         '$P/PurchaseOrder/ShippingInstructions/Address'
         passing OBJECT_VALUE as "P"
         COLUMNS
         STREET VARCHAR2(64) PATH 'street',
         CITY   VARCHAR2(32) PATH 'city'
       )
 where XMLExists(
         '$P/PurchaseOrder/ShippingInstructions/Address[. contains
text {$PHRASE1} ftand {$PHRASE2} using stemming]'
         passing OBJECT_VALUE as "P",
                 'Oxford' as "PHRASE1",
                 'Center' as "PHRASE2"
       )
/
```



The index is used because `contains text` comparisons are non-case-sensitive. Results are returned because stemming identified `centre` as a stem for `center`.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

9. Open, examine, and run the `lab_18_08_09.sql` script by using the `XDBOE` database connection. This query demonstrates how to add a `window` to the `ftand` operator. A `window` makes it possible to control how many words are allowed to exist between the two terms that are being searched on.

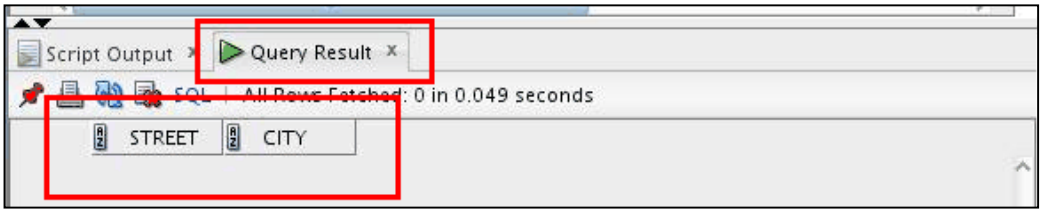
   Using the Files tab, right-click the `lab_18_08_09.sql` script, and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon to execute the query. The output is displayed in the Query Result tab.

   This query demonstrates how to add a `window` to the `ftand` operator. A `window` makes it possible to control how many words are allowed to exist between the two terms that are being searched on. In this case the `window` is restricted to `2` words and consequently no documents are returned.

```
select distinct STREET, CITY
  from PURCHASEORDER,
       XMLTABLE(
         '$P/PurchaseOrder/ShippingInstructions/Address'
         passing OBJECT_VALUE as "P"
         COLUMNS
         STREET VARCHAR2(64) PATH 'street',
         CITY   VARCHAR2(32) PATH 'city'
       )
 where XMLExists(
         '$P/PurchaseOrder/ShippingInstructions/Address[. contains
text {$PHRASE1} ftand {$PHRASE2} using stemming window 2 words]'
         passing OBJECT_VALUE as "P",
                 'Science' as "PHRASE1",
                 'Magdalen' as "PHRASE2"
       )
/
```

   The index is used in this example because the `contains text` comparison is non-case-sensitive.

   The `Window` clause specifies that the words must appear within `2` words of each other. No results are returned because the window is too small.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

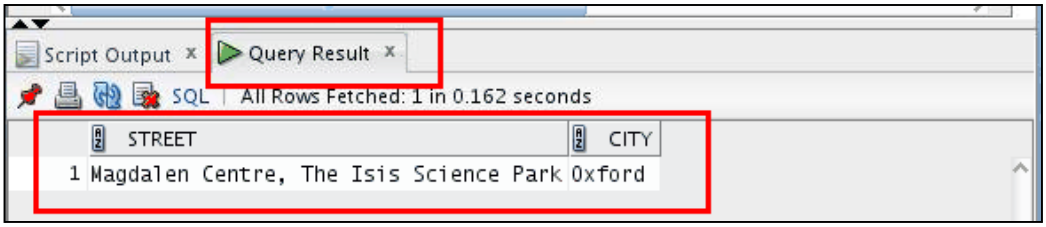
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

10. Open, examine, and run the `lab_18_08_10.sql` script by using the `XDBOE` database connection. This query demonstrates how expanding the size of the `window` allows the search to return documents that were eliminated from the result set by the narrow window used in the previous query.

    Using the Files tab, right-click the `lab_18_08_10.sql` script, and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon to execute the query. The output is displayed in the Query Result tab.

    This query demonstrates how expanding the size of the `window` allows the search to return documents that were eliminated from the result set by the narrow window used in the previous query.

```
select distinct STREET, CITY
  from PURCHASEORDER,
       XMLTABLE(
         '$P/PurchaseOrder/ShippingInstructions/Address'
         passing OBJECT_VALUE as "P"
         COLUMNS
         STREET VARCHAR2(64) PATH 'street',
         CITY   VARCHAR2(32) PATH 'city'
       )
 where XMLExists(
         '$P/PurchaseOrder/ShippingInstructions/Address[. contains
text {$PHRASE1} ftand {$PHRASE2} using stemming window 6 words]'
         passing OBJECT_VALUE as "P",
                 'Science' as "PHRASE1",
                 'Magdalen' as "PHRASE2"
       )
/
```



The index is used because the `contains text` comparison is non-case-sensitive. The `window` clause specifies that the words must appear within `6` words of each other. Results are returned since the window is large enough.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-9: Optimizing XML Storage and Processing with XML Schema

Overview

XML schemas are used by many industries to define the XML structures used for information exchange. XML documents conforming to these XML schemas are highly structured. In this practice, you use an XML schema to optimize storage, index, query, and manage structured XML data.

Assumptions

Tasks

Viewing the `PurchaseOrder` XML Schema Document Using a Web Browser

Note: User credentials are case-sensitive. For example, for a user name to be authenticated, it must match exactly the name as it was created. The default is all uppercase letters. In this example, we enter the username as `XDBOE` and the password as `oracle`.

1. Launch your Firefox web browser. Click the browser icon.

2. Enter the following URL: http://localhost:8080/home/XDBOE

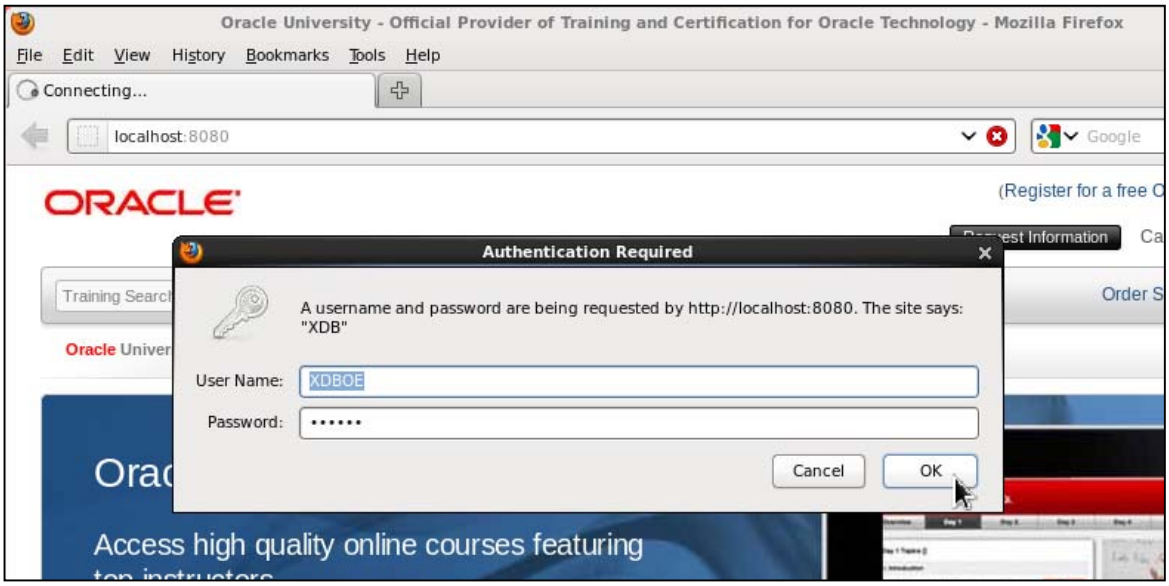
   Content in the XML DB repository is protected by Access Control Lists (ACLs). In order to access the content, it is necessary to authenticate with the appropriate username and password. When the browser opens the Authentication Required dialog box, enter `XDBOE` for the username and `oracle` for the password, and then click OK.

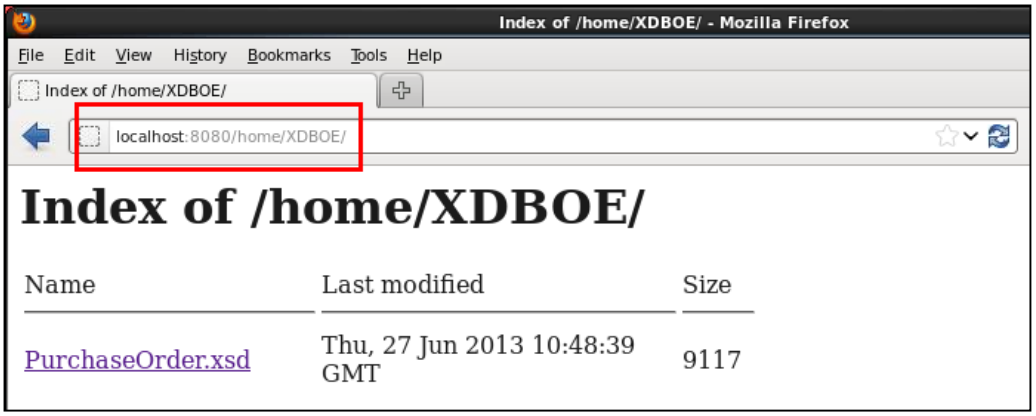
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

The browser displays the contents of the folder in question.



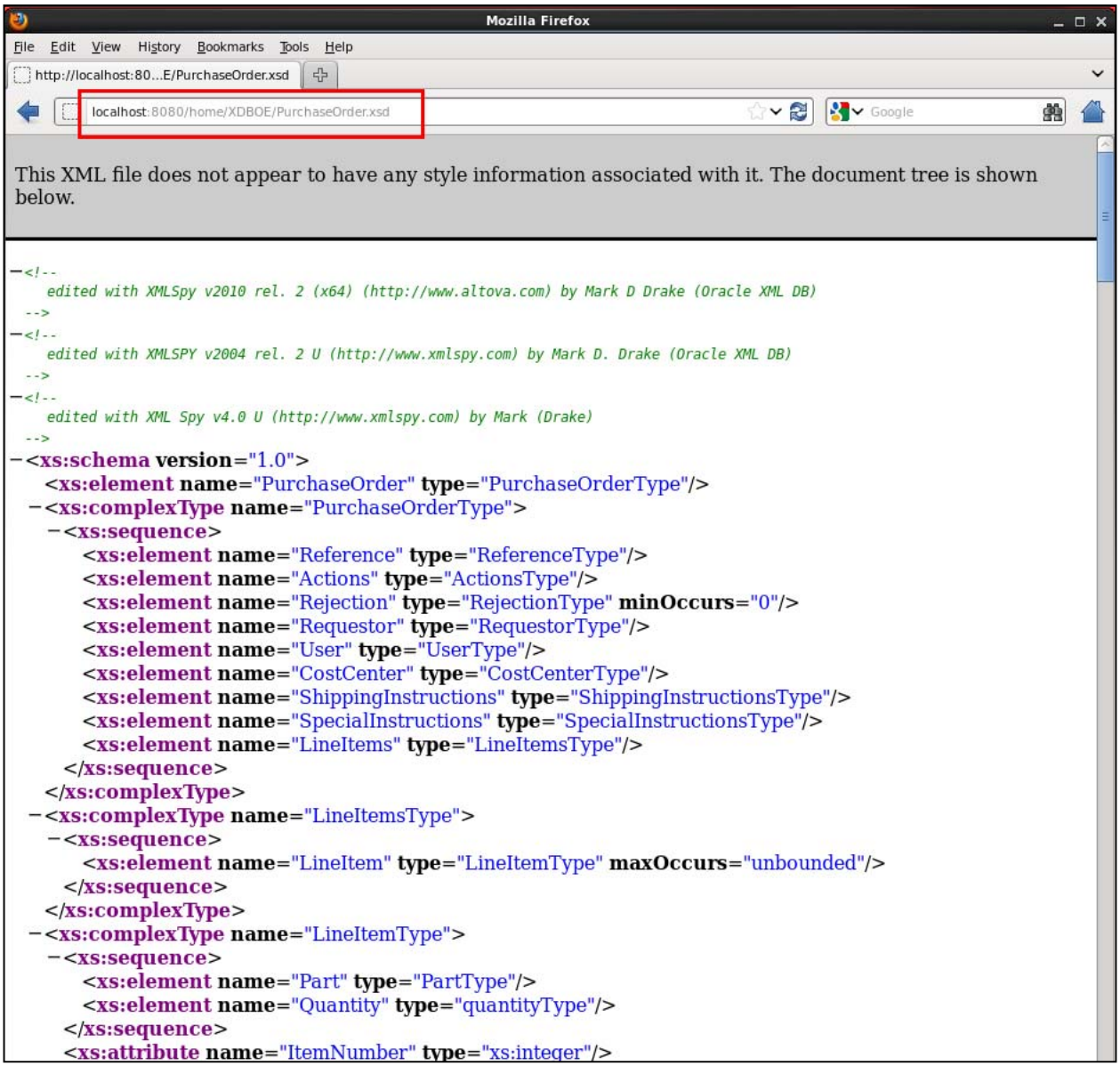


Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. Click the `PurchaseOrder.xsd` document link to display the `PurchaseOrder` XML schema.



You can see that the XML schema is an XML document, compliant with an XML Schema called the Schema For Schemas. This Schema defines the structure of a `PurchaseOrder` document.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

XML Schema Registration and Object-Relational Storage

Before an XML schema can be used with Oracle XML DB, it must be registered with the database. XML schemas can be registered for use with either binary XML storage or object-relational XML Storage. The decision on whether to use binary XML or object-relational storage will be driven by a number of factors, including:

- The complexity of the object-model defined by the XML schema
- The access patterns for instance documents
- The types of update operations that will take place
- The nature and frequency of changes to the XML schema

In general, if the XML schema describes a hierarchical object model with limited amounts of recursion and variability, then the XML schema is probably a candidate for object-relational storage. On the other hand, if the XML schema defines a complex, recursive structure with large degrees of variability and significant numbers of “any” elements, then binary XML storage is probably a better option.

Object-relational storage provides highly optimized leaf-level access and update, while the binary XML storage is more efficient for document-level operations. Consequently in ETL scenarios, where the objective is to get SQL-based access to XML content or populate existing relational tables with values extracted from the XML, object-relational storage is probably more effective. Content-centric scenarios, where document-level operations are more common, are probably better suited for binary XML, because it avoids the overhead associated with a complete decomposition and reconstruction of the XML.

Another factor to consider is XML schema evolution. If the XML schema changes in ways that do not invalidate the existing corpus of instance documents, then object-relational storage’s in-place schema evolution will probably be able to manage the schema changes. On the other hand, if the XML schema changes in ways that invalidate the existing corpus of documents on a regular basis, then schema-based binary XML—or even non-schema based binary XML—is probably more appropriate.

Object-relational storage works by analyzing the object model defined by the XML schema and deriving an equivalent SQL object model. It then creates the set of tables that allow the SQL object model to be persisted in the database. Collections in the XML model are mapped to nested tables in the SQL model. A complete, lossless bi-directional mapping between the XML and SQL models ensures that XML documents are stored in the database with no loss of fidelity. The XML abstraction, based on `XMLType`, allows developers to manipulate XML by using XQuery. XQuery operations on object-relational `XMLType` are compiled into the same query algebra as SQL, allowing the Oracle Database to optimize XQuery in the same way that it optimizes SQL. In this practice, you register an XML schema for use with object-relational storage and create indexes on the underlying storage model.

4. Open, examine, and run the `lab_18_09_04.sql` script by using the `XDBOE` database connection. This script annotates and registers the XML schema. It then describes some of the generated objects, and finally renames the nested tables used to manage the collections of `Action` and `LineItem` elements.

   Using the Files tab, right-click the lab_18_09_04.sql script, and then select Open to display it in the SQL Worksheet area. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon to execute the query. The output is displayed on the Query Result tab.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
declare
  cursor getTable
  is
  select TABLE_NAME
    from USER_XML_TABLES
   where TABLE_NAME in ( 'PURCHASEORDER');
begin
  for t in getTable() loop
    execute immediate 'DROP TABLE "' || t.TABLE_NAME || '" PURGE';
  end loop;
end;
/
declare
  V_XML_SCHEMA xmlType :=
xdburitype('/home/XDBOE/PurchaseOrder.xsd').getXML();
begin
  DBMS_XMLSCHEMA_ANNOTATE.disableMaintainDOM(V_XML_SCHEMA);

DBMS_XMLSCHEMA_ANNOTATE.setDefaultTable(V_XML_SCHEMA,'PurchaseOrder'
,'PURCHASEORDER');

DBMS_XMLSCHEMA_ANNOTATE.setSQLType(V_XML_SCHEMA,'PurchaseOrderType',
'PURCHASEORDER_T');

DBMS_XMLSCHEMA_ANNOTATE.setSQLType(V_XML_SCHEMA,'LineItemType','LINE
ITEM_T');

DBMS_XMLSCHEMA_ANNOTATE.setSQLCollType(V_XML_SCHEMA,'LineItem','LINE
ITEM_V');

DBMS_XMLSCHEMA_ANNOTATE.setSQLType(V_XML_SCHEMA,DBMS_XDB_CONSTANTS.X
SD_COMPLEX_TYPE,'ActionsType',DBMS_XDB_CONSTANTS.XSD_ELEMENT,'Action
','ACTION_T');

DBMS_XMLSCHEMA_ANNOTATE.setSQLCollType(V_XML_SCHEMA,'Action','ACTION
_V');

DBMS_XMLSCHEMA_ANNOTATE.setSqlType(V_XML_SCHEMA,DBMS_XDB_CONSTANTS.X
SD_COMPLEX_TYPE,'PurchaseOrderType',DBMS_XDB_CONSTANTS.XSD_ELEMENT,'
CostCenter','VARCHAR2');
  DBMS_XMLSCHEMA.registerSchema
  (
    SCHEMAURL        =>
'http://localhost:8080/source/schemas/poSource/xsd/purchaseOrder.xsd
',
    SCHEMADOC        => V_XML_SCHEMA,
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

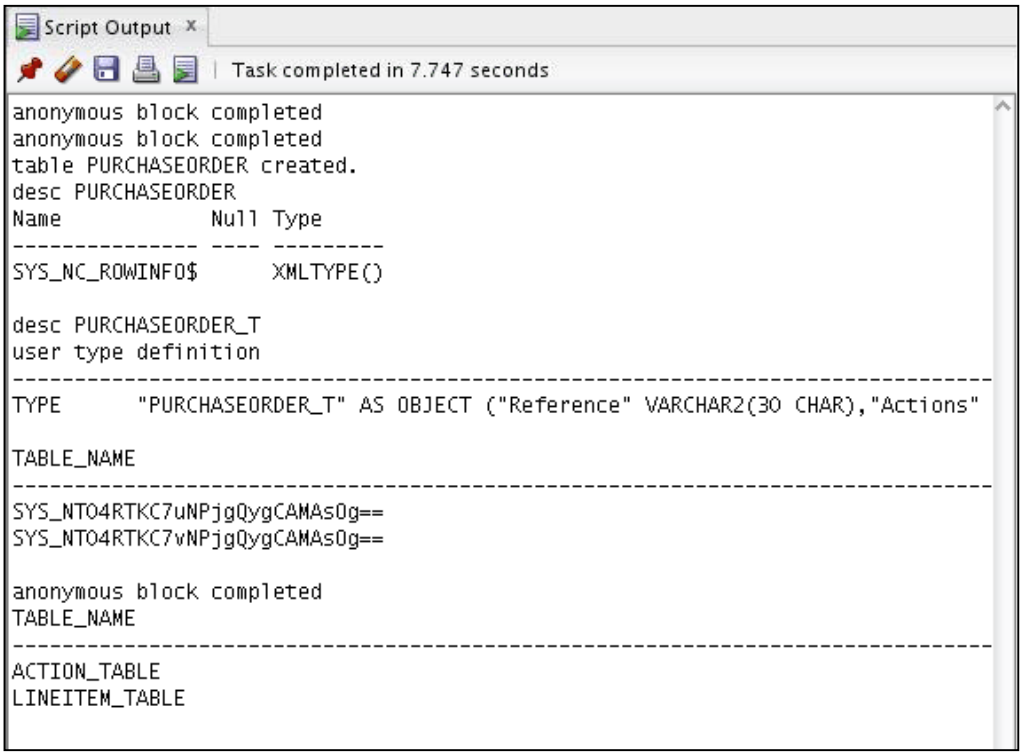
```
    LOCAL            => TRUE,
    GENBEAN          => FALSE,
    GENTYPES         => TRUE,
    GENTABLES        => TRUE,
    ENABLEHIERARCHY  => DBMS_XMLSCHEMA.ENABLE_HIERARCHY_NONE
  );
end;
/
desc PURCHASEORDER
--
desc PURCHASEORDER_T
--
select table_name
  from USER_NESTED_TABLES
 where PARENT_TABLE_NAME = 'PURCHASEORDER'
/
-- Rename the Nested Tables used to manage the collections of Action
and LineItem elements
--
begin

DBMS_XMLSTORAGE_MANAGE.renameCollectionTable(USER,'PURCHASEORDER',nu
ll,'/PurchaseOrder/Actions/Action/User','ACTION_TABLE',null);

DBMS_XMLSTORAGE_MANAGE.renameCollectionTable(USER,'PURCHASEORDER',nu
ll,'/PurchaseOrder/LineItems/LineItem/Part','LINEITEM_TABLE',null);
end;
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

5. Click the Run Script icon to execute the script. The output is displayed on the Script Output tab.



The following is the complete output of the `DESC PURCHASE_T` statement:

```
user type definition
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------
TYPE          "PURCHASEORDER_T" AS OBJECT ("Reference" VARCHAR2(30
CHAR),"Actions" "ActionsType648_T","Rejection"
"RejectionType647_T","Requestor" VARCHAR2(128 CHAR),"User"
VARCHAR2(10 CHAR),"CostCenter" VARCHAR2(4000
CHAR),"ShippingInstructions"
"ShippingInstruction641_T","SpecialInstructions" VARCHAR2(2048
CHAR),"LineItems" "LineItemsType650_T")NOT FINAL INSTANTIABLE
```

First the XML schema is loaded into memory from the XML DB repository by using the `XDBURITYPE` operator.

Next, the package `DBMS_XMLSCHEMA_ANNOTATE` is used to annotate the XML schema. As you learned earlier in this course, XML schema annotations provide a mechanism

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

that allows a DBA some control over the object model and tables created by the schema registration. This example does the following:

- **Disables DOM Fidelity**
- **Sets the name of the table that will be used to store instance documents in the database**
- **Defines explicit names for some of the SQL objects that will be created by the schema registration process**

Next the package `DBMS_XMLSCHEMA` is used to register the XML schema with the database. The `SchemaURL` is nothing more than a unique identifier for the XML Schema. `LOCAL` is set to `TRUE`, meaning that the XML schema is registered as a local XML schema. A local XML schema can be used only from the database schema that was used to register the XML schema. Registering an XML schema as a global schema allows the XML schema to be shared across multiple database schemas. `GENTypes` is set to `TRUE`, forcing the creation of the SQL objects that will be used to manage the XML. A SQL object will be created for each `complexType` defined by the XML schema. `GENTables` is also set to `TRUE`, forcing the creation the tables required to persist the SQL objects in the database.

In this case, the `XMLType` table `PURCHASEORDER` is created. `PURCHASEORDER` uses object-relational storage. Because the `PurchaseOrder` schema defines two repeating elements, two SQL collection types are generated, and two nested tables are created to manage the collections. The nested tables are given system-generated names. These names can be difficult to work with, so `DBMS_XMLSTORAGE_MANAGE` is used to supply user-friendly names.

Bulk Loading XML Files by Using SQL Loader

Oracle's SQL Loader utility provides a convenient way of loading large volumes of XML documents into the database. The following example shows a SQL Loader control file that will load a set of documents into an `XMLType` table. SQL Loader can be used with binary and object-relational storage models, and works with both `XMLType` tables and `XMLType` columns.

```
options (direct=true)
load data
infile '2012.dat'
append
into table PURCHASEORDER
xmltype(XMLDATA) (
filename filler char(120),
XMLDATA lobfile(filename) terminated by eof)
```

In this example, direct-path loading is used to improve performance. The files will be loaded into the `PURCHASEORDER` table. The names of the files to be loaded are contained in the file `2012.dat`. Each file name is a maximum of `120` characters long.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

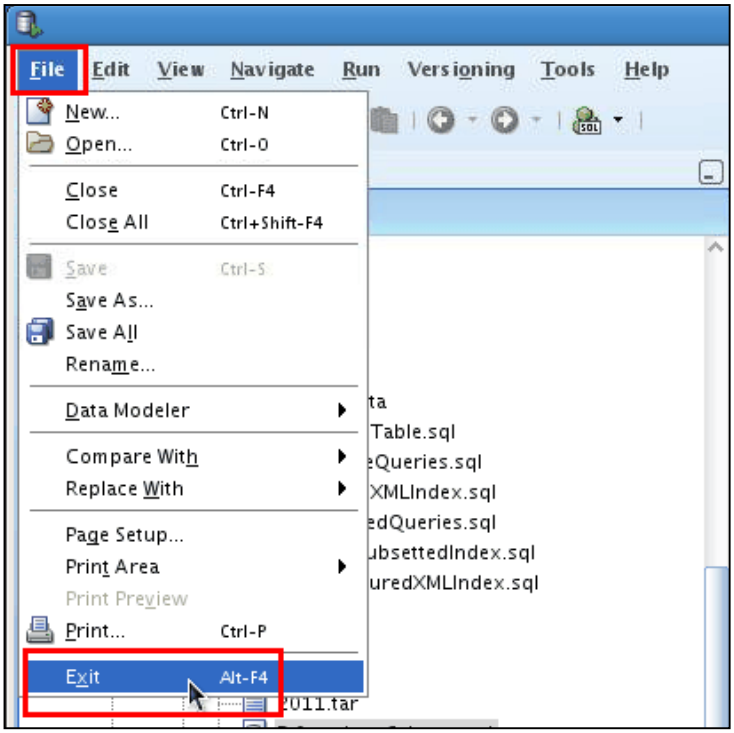
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

The file `2012.dat` contains `10,000` entries in the following format.

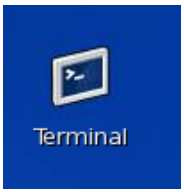
```
sampleData/2012/Apr/ABANDA-20120407201438314PDT.xml
sampleData/2012/Apr/ABANDA-20120417140257543PDT.xml
sampleData/2012/Apr/ABANDA-20120427125050959PDT.xml
. . .
```

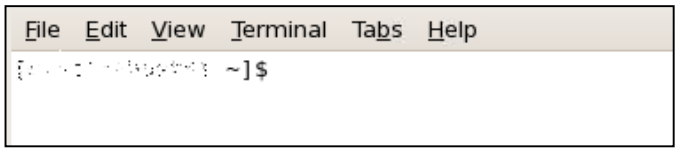
In this practice, you use SQL Loader to load the sample documents into the database.

1. Close SQL Developer. Select File > Exit from the menu bar.



2. Open a command-line terminal window. Click the Terminal icon on the desktop.





Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. Execute the following commands:

```
export XDB_HOL="/home/oracle/Desktop/DatabaseTrack/XMLDB"
cd $XDB_HOL/sqlLoader
```

The `sqlLoader` folder contains the `2012.ctl` and `2012.dat` files that you will need in this practice.

```
[oracle ~]$ export XDB_HOL="/home/oracle/Desktop/DatabaseTrack/XMLDB"
[oracle ~]$ cd $XDB_HOL/sqlLoader
(reverse-i-search)`': 
```

4. Execute the following command so that SQL Loader will direct path load `10,000` documents into the `PURCHASEORDER` table.

```
sqlldr -userid=XDBOE/oracle -control=2012.ctl -rows=1000
```

```
[oracle@EDT3R17P0 sqlLoader]$ sqlldr -userid=XDBOE/oracle -control=2012.ctl -rows=1000

SQL*Loader: Release 12.1.0.1.0 - Production on Wed Sep 18 09:21:03 2013

Copyright (c) 1982, 2013, Oracle and/or its affiliates.  All rights reserved.

Path used:      Conventional
Commit point reached - logical record count 1000
Commit point reached - logical record count 2000
Commit point reached - logical record count 3000
Commit point reached - logical record count 4000
Commit point reached - logical record count 5000
Commit point reached - logical record count 6000
Commit point reached - logical record count 7000
Commit point reached - logical record count 8000
Commit point reached - logical record count 9000
Commit point reached - logical record count 10000

Table PURCHASEORDER:
  10000 Rows successfully loaded.

Check the log file:
  2012.log
for more information about the load.
[oracle@EDT3R17P0 sqlLoader]$ 
```

5. After SQL Loader completes successfully, close the terminal window.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-10: Using XQuery with Object-Relational Storage

Overview

One of the key features of Oracle XML DB is that applications are developed independently of the way in which the XML is stored. This means that the choice of storage model does not affect the way in which applications are written. In the previous practice, the binary XML version of the `PURCHASEORDER` table was replaced with one based on object-relational storage. In this practice, you observe how no changes are required to execute the XQuery expressions used with the binary XML table on the object-relational table.

Assumptions

Tasks

1. Start SQL Developer. Open the `lab_18_10_01.sql` script.

   Note: This script contains the same individual queries that you created and ran in Practice 18-03.

```
set long 4096
--
-- 1. Use XQuery and fn:collection to count the documents in the
PURCHASEORDER
-- table.
--
select *
  from XMLTABLE(
          'count(fn:collection("oradb:/XDBOE/PURCHASEORDER"))'
       )
/
--
-- 2. Use XQuery to select a single document from the PURCHASEORDER
table
--
select *
  from XMLTABLE(
          'for $i in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[Reference/
text()=$REFERENCE]
           return $i'
           passing 'AFRIPP-2012060818343243PDT' as "REFERENCE"
       )
/
--
-- 3. XQuery with multiple predicates, returning a single node
(Reference)
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
--
select *
  from XMLTABLE(
          'for $i in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[CostCenter
=$CC and Requestor=$REQUESTOR and count(LineItems/LineItem) >
$QUANTITY]/Reference
           return $i'
           passing 'A60' as "CC", 'Diana Lorentz' as "REQUESTOR", 5
as "QUANTITY"
       )
/
--
-- 4. XQuery constructing a new summary document from the documents
that match the specified predicates
-- Also demonstrates the use of nested FOR loops, one for the set of
PurchaseOrder documents, and one for
-- the LineItem elements
--
select *
  from XMLTable(
         '<Summary UPC="{$UPC}">
          {
           for $p in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder
            for $l in $p/LineItems/LineItem[Quantity > $Quantity and
Part/text() =$UPC]
            order by $p/Reference
              return
                <PurchaseOrder reference="{$p/Reference/text()}"
lineItem="{fn:data($l/@ItemNumber)}" Quantity="{$l/Quantity}"/>
          }
          </Summary>'
          passing  '707729113751' as UPC, 3 as "Quantity"
       )
/
--
-- 5. Use XMLSerialize to format the XMLType and serialize it as a
CLOB.
--    Allows result to be viewed in products that do not support
XMLType.
--    XMLSerialize allows control over the layout of the serialized
--    XML.
--
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
select XMLSERIALIZE(CONTENT COLUMN_VALUE AS CLOB INDENT SIZE=2)
  from XMLTable(
         '<Summary UPC="{$UPC}">
          {
           for $p in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder
            for $l in $p/LineItems/LineItem[Quantity > $Quantity and
Part/text() =$UPC]
            order by $p/Reference
              return
                <PurchaseOrder reference="{$p/Reference/text()}"
lineItem="{fn:data($l/@ItemNumber)}" Quantity="{$l/Quantity}"/>
          }
          </Summary>'
          passing  '707729113751' as UPC, 3 as "Quantity"
       )
/
--
-- 6. Using XMLTable to create an in-line relational view from the
documents that match the XQuery expression.
--
select *
  from xmlTable(
          'for $p in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder
            for $l in $p/LineItems/LineItem[Quantity > 3 and
Part/text() = "707729113751"]
              return
              <Result ItemNumber="{fn:data($l/@ItemNumber)}">
                {
                  $p/Reference,
                  $p/Requestor,
                  $p/User,
                  $p/CostCenter,
                  $l/Quantity
                }

<Description>{fn:data($l/Part/@Description)}</Description>
                <UnitPrice>{fn:data($l/Part/@UnitPrice)}</UnitPrice>
                <PartNumber>{$l/Part/text()}</PartNumber>
             </Result>'
             columns
             SEQUENCE      for ordinality,
             ITEM_NUMBER       NUMBER(3) path '@ItemNumber',
```

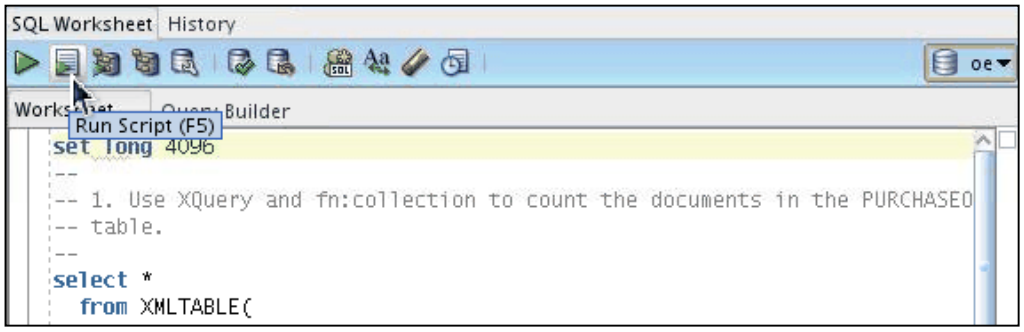
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
              REFERENCE     VARCHAR2( 30) path 'Reference',
              REQUESTOR     VARCHAR2(128) path 'Requestor',
              USERID        VARCHAR2( 10) path 'User',
              COSTCENTER    VARCHAR2(  4) path 'CostCenter',
              DESCRIPTION   VARCHAR2(256) path 'Description',
              PARTNO        VARCHAR2( 14) path 'PartNumber',
              QUANTITY       NUMBER(12,4) path 'Quantity',
              UNITPRICE      NUMBER(14,2) path 'UnitPrice'
       )
/
--
-- 7. Joining relational and XML tables using XQuery.
--
select REQUESTOR, DEPARTMENT_NAME
  from HR.EMPLOYEES e, HR.DEPARTMENTS d,
       XMLTABLE(
         'for $p in
fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder
            where $p/User=$EMAIL and $p/Reference=$REFERENCE
            return $p'
          passing 'AFRIPP-2012060818343243PDT' as "REFERENCE",
e.EMAIL as "EMAIL"
          COLUMNS
          REQUESTOR path 'Requestor/text()',
          USERNAME  path 'User'
       )
 where e.DEPARTMENT_ID = d.DEPARTMENT_ID
/
```

2. Click the Run Script icon to execute the script. The output is displayed in the Script Output tab.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Script Output

Task completed in 2.475 seconds

```
COLUMN_VALUE
--------------------------------------------------------------------------------
10000

COLUMN_VALUE
--------------------------------------------------------------------------------
<PurchaseOrder>
  <Reference>AFRIPP-2012060818343243PDT</Reference>
  <Actions>
    <Action>
      <User>MHARTSTE</User>
    </Action>
  </Actions>
  <Rejection/>
  <Requestor>Adam Fripp</Requestor>
  <User>AFRIPP</User>
  <CostCenter>A50</CostCenter>
  <ShippingInstructions>
    <name>Adam Fripp</name>
    <Address xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="address_US">
      <street>200 Sporting Green</street>
      <city>South San Francisco</city>
      <state>CA</state>
      <zipCode>99236</zipCode>
      <country>United States of America</country>
    </Address>
    <telephone>983-814-9068</telephone>
  </ShippingInstructions>
  <SpecialInstructions>COD</SpecialInstructions>
  <LineItems>
    <country>United States of America</country>
    </Address>
    <telephone>983-814-9068</telephone>
  </ShippingInstructions>
  <SpecialInstructions>COD</SpecialInstructions>
  <LineItems>
    <LineItem ItemNumber="1">
      <Part Description="Star Trek III: the Search for Spock" UnitPrice="19.95">97360162141</Part>
      <Quantity>6</Quantity>
    </LineItem>
    <LineItem ItemNumber="2">
      <Part Description="Agatha Christie&apos;s Miss Marple 1" UnitPrice="32.95">733961702576</Part>
      <Quantity>7</Quantity>
    </LineItem>
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
COLUMN_VALUE
--------------------------------------------------------------------------------
<Reference>DLORENTZ-20120411133920295PDT</Reference>

<Reference>DLORENTZ-2012042922031647PDT</Reference>

<Reference>DLORENTZ-20121219144418724PST</Reference>

<Reference>DLORENTZ-20120710194550792PDT</Reference>

<Reference>DLORENTZ-20120723165626922PDT</Reference>

<Reference>DLORENTZ-20120728151113368PDT</Reference>

<Reference>DLORENTZ-20120206180918983PST</Reference>

<Reference>DLORENTZ-20120318211347802PST</Reference>

<Reference>DLORENTZ-20120320185010571PST</Reference>
```

```
<Reference>DLORENTZ-20121103141148589PST</Reference>

<Reference>DLORENTZ-2012112521245631PST</Reference>


 15 rows selected

COLUMN_VALUE
--------------------------------------------------------------------------------
<Summary UPC="707729113751"><PurchaseOrder reference="DBERNSTE-20120427202452121PDT" lineItem="5" Quantity="6"></Purchas

XMLSERIALIZE(CONTENTCOLUMN_VALUEASCLOBINDENTSIZE=2)
--------------------------------------------------------------------------------
<Summary UPC="707729113751">
  <PurchaseOrder reference="DBERNSTE-20120427202452121PDT" lineItem="5" Quantity="6"/>
  <PurchaseOrder reference="LBISSOT-20120815132845576PDT" lineItem="5" Quantity="4"/>
  <PurchaseOrder reference="LDORAN-20120408225231939PDT" lineItem="3" Quantity="4"/>
  <PurchaseOrder reference="PVARGAS-20121030162426359PST" lineItem="2" Quantity="4"/>
  <PurchaseOrder reference="RLADWIG-20120406190600968PDT" lineItem="4" Quantity="9"/>
  <PurchaseOrder reference="WSMITH-20120224224555785PST" lineItem="1" Quantity="5"/>
</Summary>
```

```
  SEQUENCE ITEM_NUMBER REFERENCE                      REQUESTOR
---------- ----------- ------------------------------ ------------------------------
         1           5 LBISSOT-20120815132845576PDT   Laura Bissot
         2           2 PVARGAS-20121030162426359PST   Peter Vargas
         3           4 RLADWIG-20120406190600968PDT   Renske Ladwig
         4           3 LDORAN-20120408225231939PDT    Louise Doran
         5           1 WSMITH-20120224224555785PST    William Smith
         6           5 DBERNSTE-20120427202452121PDT  David Bernstein

 6 rows selected

REQUESTOR
--------------------------------------------------------------------------------
Adam Fripp
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Confirm that all of the queries executed successfully. No changes were required to the XQuery expressions. As expected, the results of running the queries on object-relational storage are identical to the results of running the queries on binary XML storage.

3. Open the `lab_18_10_03.sql` script.

```
-- A set of simple queries to demonstrate how indexing can optimize
-- XQuery operations.

select *
  from XMLTABLE
       (

'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[Ref
erence/text()=$REFERENCE])'
           passing 'AFRIPP-2010060818343243PDT' as "REFERENCE"
       )
/

select *
  from XMLTABLE
       (

'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[Use
r/text()=$USER])'
           passing 'AFRIPP' as "USER"
       )
/
select *
  from XMLTABLE
       (

'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[Lin
eItems/LineItem[Part/text()=$UPC]])'
           passing '707729113751' as "UPC"
       )
/
select *
  from XMLTABLE
       (
```

Oracle University and Error : You are not a Valid Partner use only

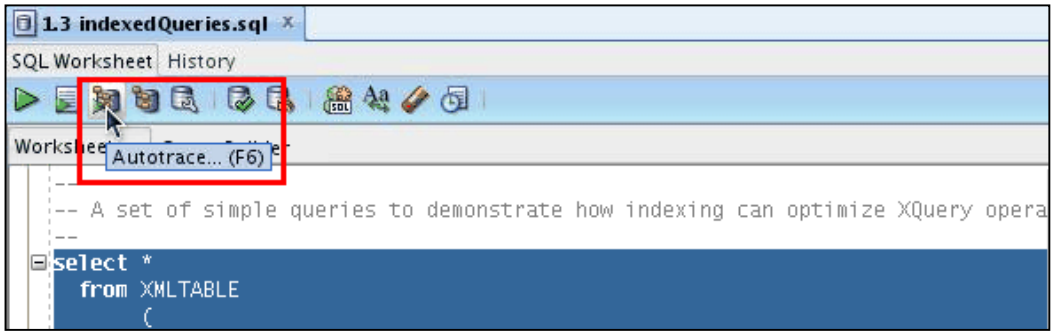
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

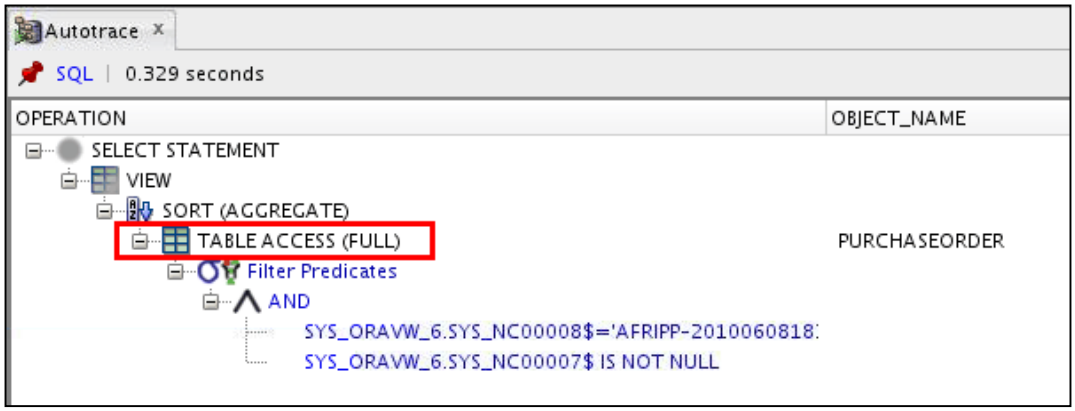
```
'count(fn:collection("oradb:/XDBOE/PURCHASEORDER")/PurchaseOrder[Lin
eItems/LineItem[Part/text() = $UPC and Quantity > $QUANTITY]])'
            passing '707729113751' as "UPC", 8 as "QUANTITY"
       )
/
```

4. Use Autotrace to generate the execution plan for each of the queries. The Autotrace output shows that full table scans are being used to execute the queries. This is expected because, currently, there are no indexes that can be used to optimize the queries.

   Highlight query 1, and then click the Autotrace icon on the SQL Worksheet toolbar. The output is displayed on the Autotrace tab.

Query 1:





Highlight query 2, and then click the Autotrace icon on the SQL Worksheet toolbar. The output is displayed in the Autotrace tab.

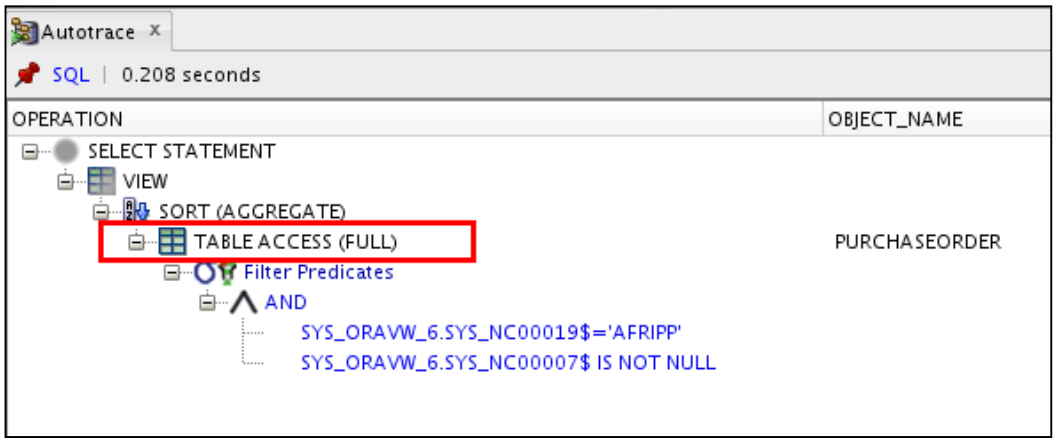
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

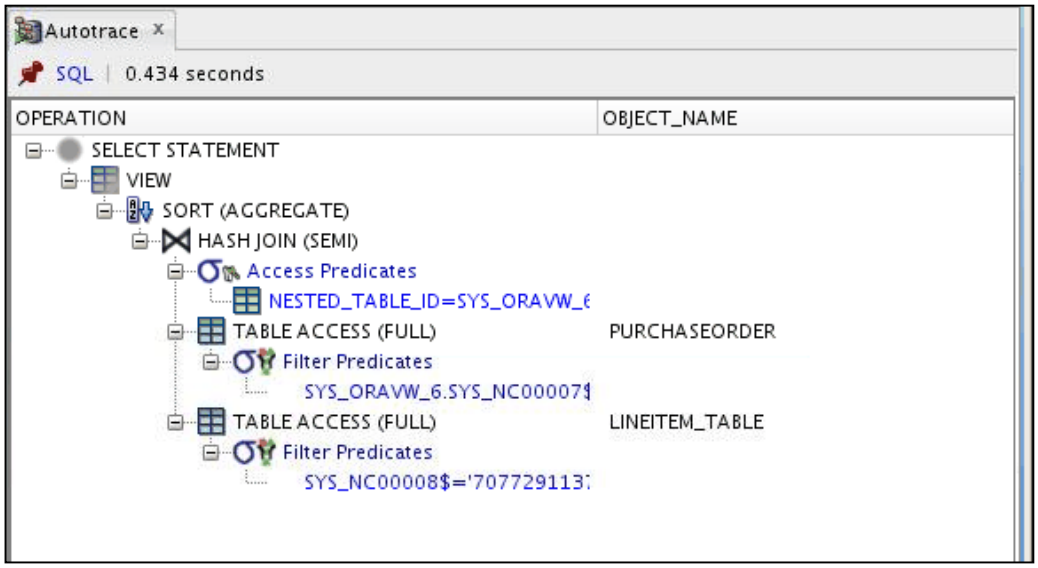
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Query 2:



Highlight query 3, and then click the Autotrace icon on the SQL Worksheet toolbar. The output is displayed on the Autotrace tab.

Query 3:



Highlight query 4, and then click the Autotrace icon on the SQL Worksheet toolbar. The output is displayed in the Autotrace tab.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Query 4:

Autotrace

SQL | 0.213 seconds

OPERATION	OBJECT_NAME
SELECT STATEMENT	
VIEW	
SORT (AGGREGATE)	
NESTED LOOPS	
NESTED LOOPS	
SORT (UNIQUE)	
TABLE ACCESS (FULL)	LINEITEM_TABLE
Filter Predicates	
AND	
SYS_NC00008$='707729113751'	
Quantity>8	
INDEX (UNIQUE SCAN)	LINEITEM_TABLE_MIDX
Access Predicates	
NESTED_TABLE_ID=SYS_ORAVW_6.SYS_NC000290	
TABLE ACCESS (BY INDEX ROWID)	PURCHASEORDER
Filter Predicates	
SYS_ORAVW_6.SYS_NC00007$ IS NOT NULL	

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-11: Indexing Object-Relational Storage to Optimize XQuery Operations

Overview

XQuery operations on object-relational storage are optimized by creating conventional B-Tree indexes on the tables that are used to persist the SQL objects. There are two ways of determining which tables and columns should be indexed.

- The first involves generating the execution plan for the query and then creating indexes based on the information in the plan. In this case the index will be created by using the system-generated table and column names shown in the execution plan.
- The second involves using the `XPATH2TABCOLMAPPING` method in the `DBMS_XMLSTORAGE_MANAGE` package to determine which table and column correspond to a particular XPath expression, and then using this information to generate a DDL statement that creates the appropriate index.

In this practice, you use the second approach to create indexes to optimize queries on the `Reference` and `User` nodes and a compound index on the repeating elements `Part Number` and `Quantity`.

Assumptions

Tasks

1. Open the `lab_18_11_01.sql` script.

```
-- Create a Unique Index on /PurchaseOrder/Reference

SELECT 'create unique index "REFERENCE_INDEX" on "' ||
i.TARGET_TABLE_NAME || '" ("' || i.TARGET_COLUMN_NAME || '");'
REFERENCE_INDEX
  from XMLTable(
         '/Result/Mapping'
         passing
DBMS_XMLSTORAGE_MANAGE.XPATH2TABCOLMAPPING(USER,'PURCHASEORDER',NULL
,'/PurchaseOrder/Reference','')
         COLUMNS
           TARGET_TABLE_NAME   VARCHAR2(30) path '@TableName',
           TARGET_COLUMN_NAME  VARCHAR2(30) path '@ColumnName'
        ) i
/

-- Create a Non-Unique Index on /PurchaseOrder/Requestor
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
SELECT 'create index "USER_INDEX" on "' || i.TARGET_TABLE_NAME || '"
("' || i.TARGET_COLUMN_NAME || '");' USER_INDEX
  from XMLTable(
          '/Result/Mapping'
           passing
DBMS_XMLSTORAGE_MANAGE.XPATH2TABCOLMAPPING(USER,'PURCHASEORDER',NULL
,'/PurchaseOrder/User','')
           COLUMNS
           TARGET_TABLE_NAME   VARCHAR2(30) path '@TableName',
           TARGET_COLUMN_NAME  VARCHAR2(30) path '@ColumnName'
        ) i
/
-- Create a Non-Unique Composite Index on
-- /PurchaseOrder/LineItems/LineItem/Part/text() and
-- /PurchaseOrder/LineItems/LineItem/Quantity

SELECT 'create index "PART_NUMBER_QUANTITY_INDEX" on "' ||
i.TARGET_TABLE_NAME || '" ("' || i.TARGET_COLUMN_NAME || '","' ||
j.TARGET_COLUMN_NAME || '");' PART_NUMBER_QUANTITY_INDEX
  from XMLTable(
         '/Result/Mapping'
         passing
DBMS_XMLSTORAGE_MANAGE.XPATH2TABCOLMAPPING(USER,'PURCHASEORDER',NULL
,'/PurchaseOrder/LineItems/LineItem/Part','')
         COLUMNS
         TARGET_TABLE_NAME   VARCHAR2(30) path '@TableName',
         TARGET_COLUMN_NAME  VARCHAR2(30) path '@ColumnName'
       ) i,
       XMLTable(
          '/Result/Mapping'
          passing
DBMS_XMLSTORAGE_MANAGE.XPATH2TABCOLMAPPING(USER,'PURCHASEORDER',NULL
,'/PurchaseOrder/LineItems/LineItem/Quantity','')
          COLUMNS
          TARGET_TABLE_NAME   VARCHAR2(30) path '@TableName',
          TARGET_COLUMN_NAME  VARCHAR2(30) path '@ColumnName'
       ) j
/
select 'call dbms_stats.gather_schema_stats(USER);' from dual
/
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

2. The script uses the `DBMS_XMLSTORAGE_MANAGE` package to determine which tables and columns need to be indexed. The output from running these queries is a set of DDL statements that will create the required indexes. Click the Run Script icon to execute the script. The output is displayed on the Script Output tab.

```
--
--  Create an Unique Index on /PurchaseOrder/Reference
--
SELECT 'create unique index "REFERENCE_INDEX" on "' || i.TARGET_TABLE_NAME || '" ("' || i.TARGET_COLUMN_NAM
  from XMLTable(
         '/Result/Mapping'
         passing DBMS_XMLSTORAGE_MANAGE.XPATH2TABCOLMAPPING(USER,'PURCHASEORDER',NULL,'/PurchaseOrder/Refer
```

```
Script Output
Task completed in 0.48 seconds
REFERENCE_INDEX
--------------------------------------------------------------------------------
create unique index "REFERENCE_INDEX" on "PURCHASEORDER" ("SYS_NC00008$");

USER_INDEX
--------------------------------------------------------------------------------
create index "USER_INDEX" on "PURCHASEORDER" ("SYS_NC00019$");

PART_NUMBER_QUANTITY_INDEX
--------------------------------------------------------------------------------
create index "PART_NUMBER_QUANTITY_INDEX" on "LINEITEM_TABLE" ("SYS_NC00008$","Quantity");

'CALLDBMS_STATS.GATHER_SCHEMA_STATS(USER);'
--------------------------------------------
call dbms_stats.gather_schema_stats(USER);
```

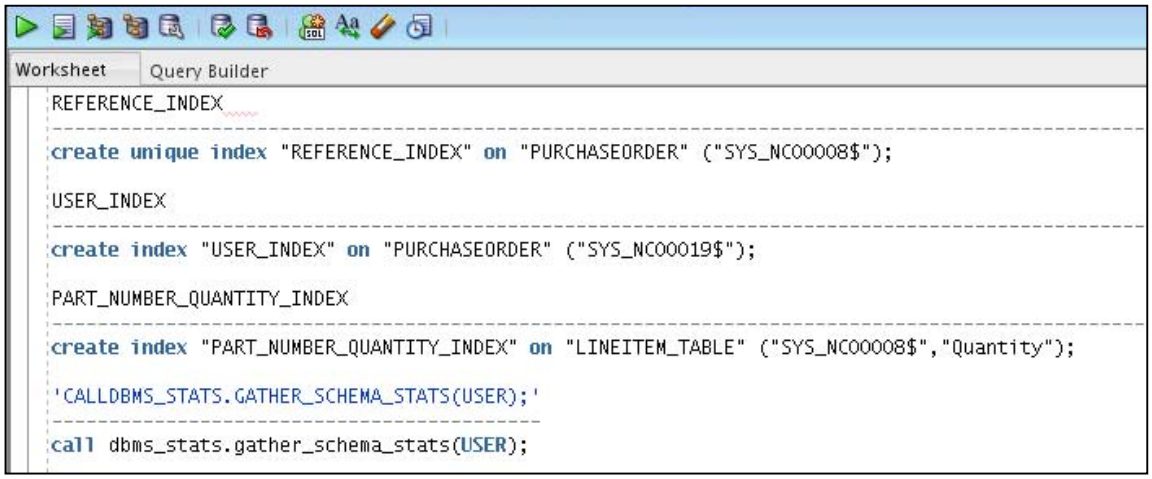
3. Click the **Run Script Output as Script** icon. This opens a new SQL Worksheet that contains the `create index` statements.

```
Script Output
Task completed in 0.48 seconds
REFERENCE_INDEX
Run Script Output as a Script (Ctrl+F5)
--------------------------------------------------------------------------------
create unique index "REFERENCE_INDEX" on "PURCHASEORDER" ("SYS_NC00008$");

USER_INDEX
--------------------------------------------------------------------------------
create index "USER_INDEX" on "PURCHASEORDER" ("SYS_NC00019$");

PART_NUMBER_QUANTITY_INDEX
--------------------------------------------------------------------------------
create index "PART_NUMBER_QUANTITY_INDEX" on "LINEITEM_TABLE" ("SYS_NC00008$","Quantity");

'CALLDBMS_STATS.GATHER_SCHEMA_STATS(USER);'
--------------------------------------------
call dbms_stats.gather_schema_stats(USER);
```

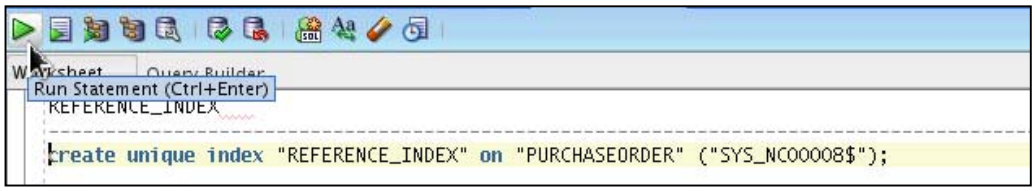
Oracle University and Error : You are not a Valid Partner use only

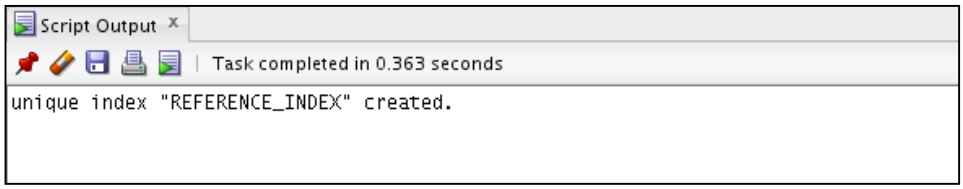
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



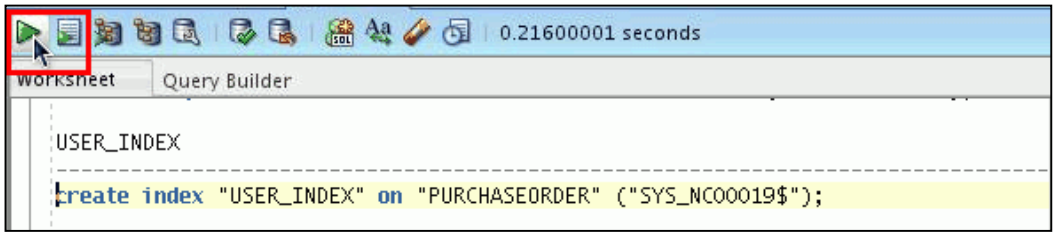
4. Position the cursor at the beginning of the first `create index` statement, and then click the Run Statement icon. This creates the index needed to optimize queries on the `Reference` element.





5. Repeat step 4 for each of the remaining `create index` statements as well as to call the `DBMS_STATS.GATHER_SCHEMA_STATS` package procedure.

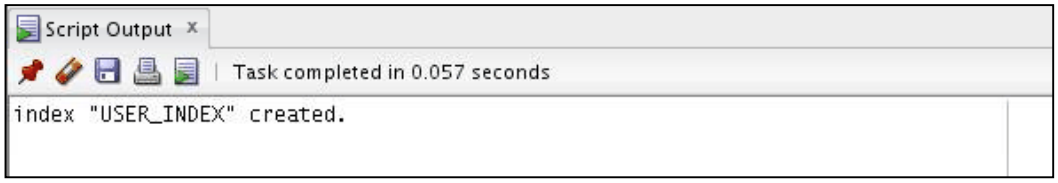
Statement 2:



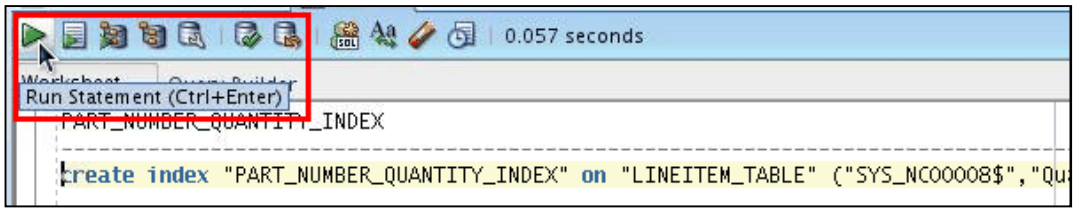
Oracle University and Error : You are not a Valid Partner use only

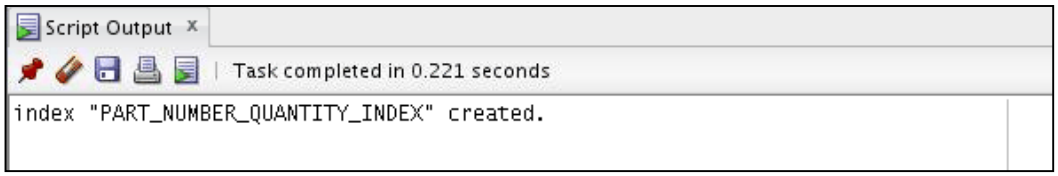
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

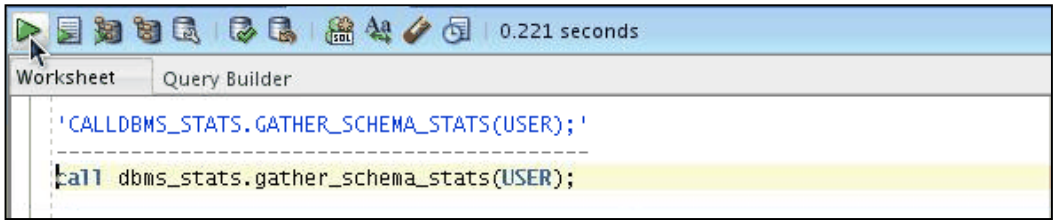


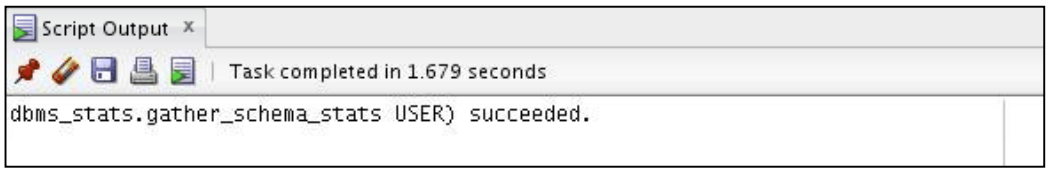
Statement 3:





Statement 4:





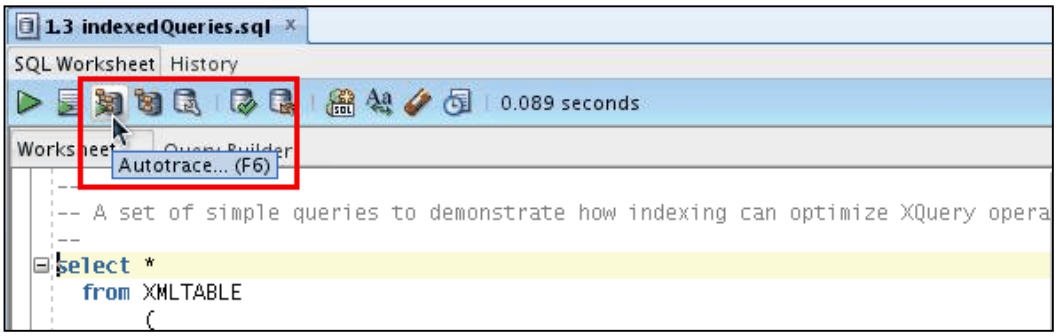
6. Return to the SQL Worksheet containing the `lab_18_10_03.sql` script or re-open the `lab_18_10_03.sql` script.
7. Use Autotrace to generate the execution plan for each of the four queries. The Autotrace output now shows that the indexes `REFERENCE_INDEX, USER_INDEX and PART_NUMBER_QUANTITY_INDEX` indexes are used to optimize the XQuery operations.
8. Highlight query 1 or place the cursor at the start of the query, and then click the Autotrace icon on the SQL Worksheet tab. Repeat this step for the remaining queries in this script.

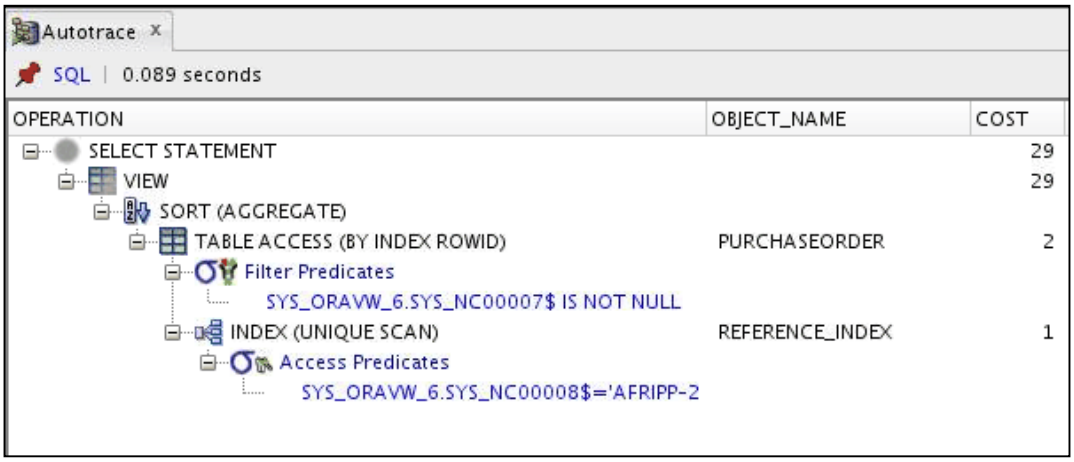
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

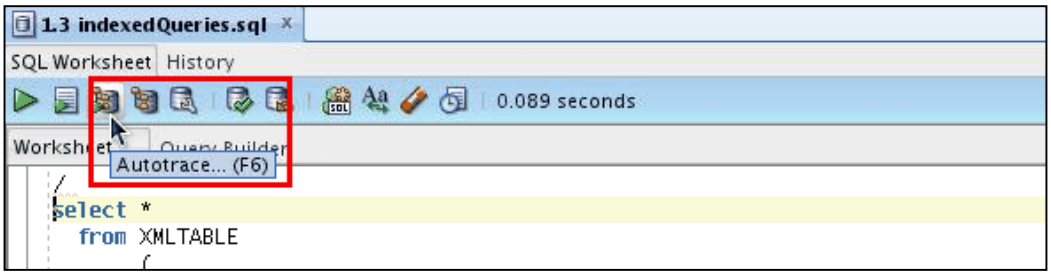
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Query 1:





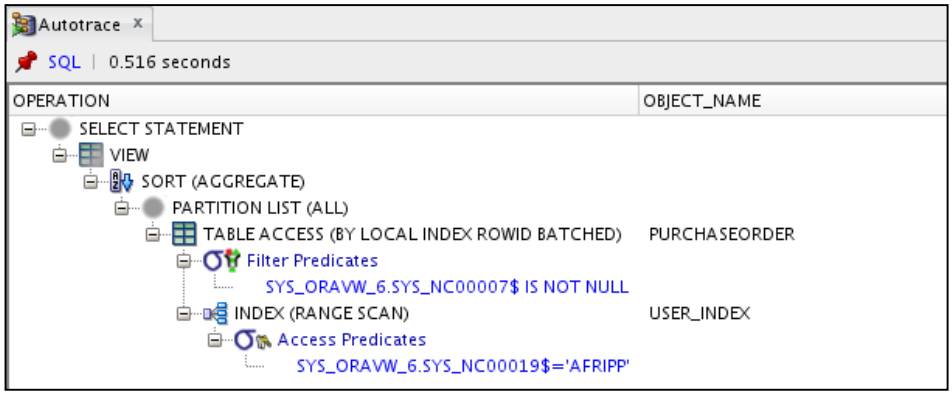
Query 2:



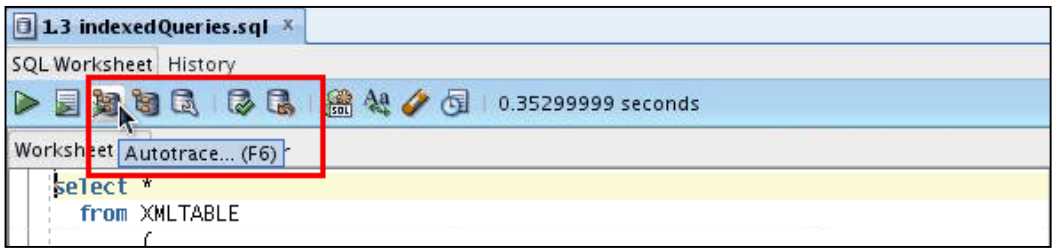
Oracle University and Error : You are not a Valid Partner use only

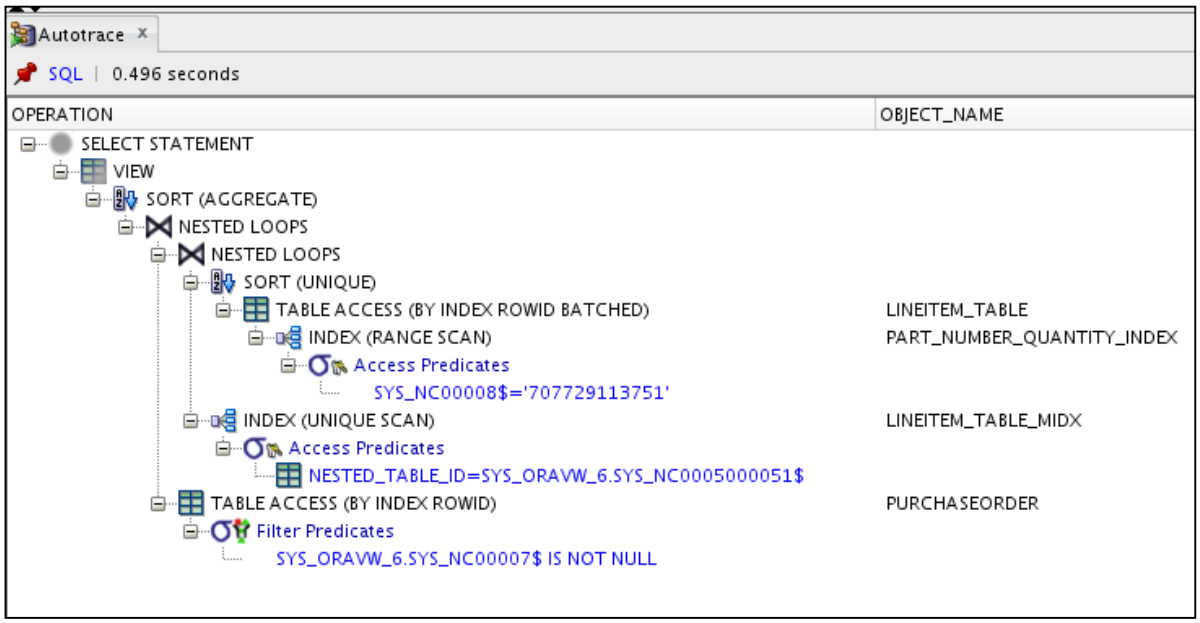
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED



Query 3:



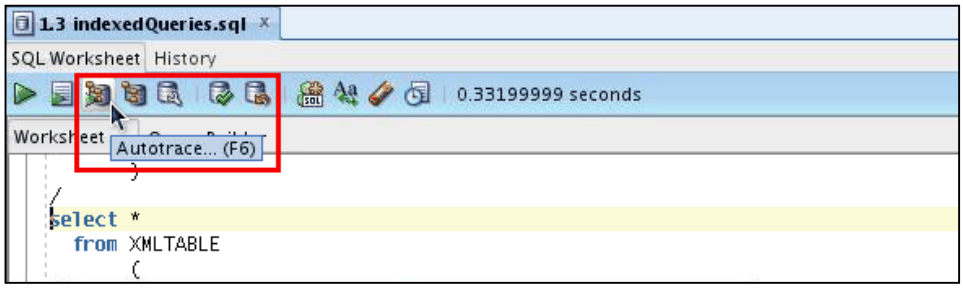


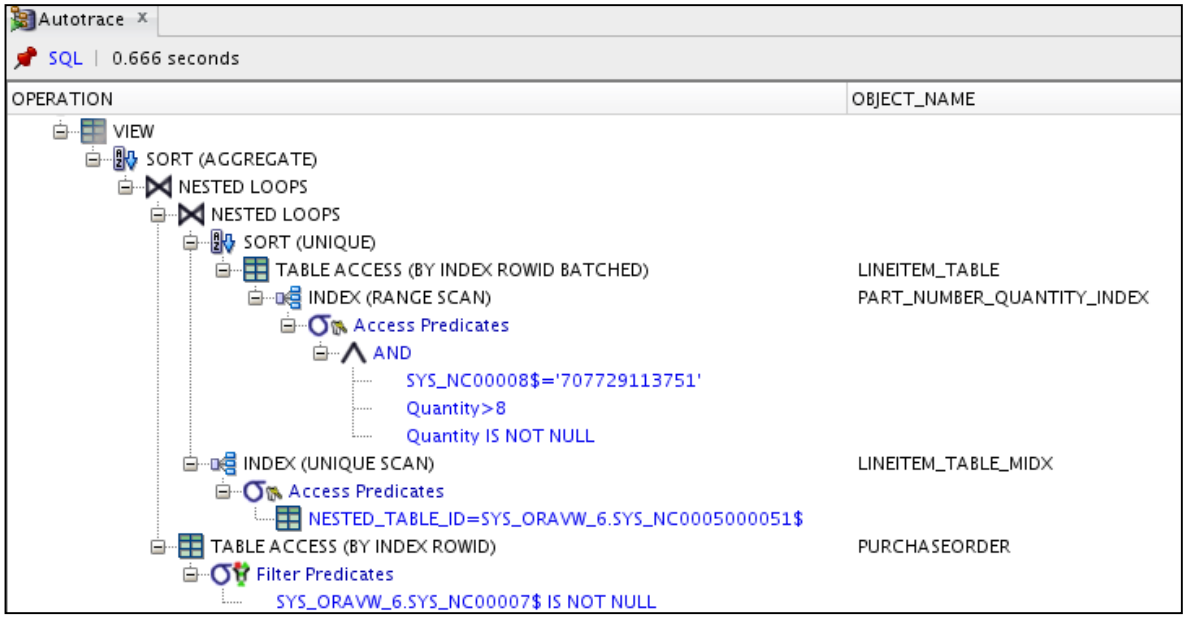
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Query 4:





Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-12: Equi-Partitioning of Object-Relational Storage

Overview

Partitioning helps simplify XML data lifecycle management and performance. Equi-partitioning for object-relational storage was introduced in Oracle Database 11*g* Release 2. Equi-partitioning ensures that all of the nested tables used to manage collections automatically follow the partitioning strategy defined for the parent `XMLType` table. This allows partitioning pruning and partition maintenance operations to operate on `XMLType` tables in the same way as they operate on simple relational tables.

In this practice, you create a partitioned object-relational `XMLType` table.

Assumptions

Tasks

1. Open, review, and run the `lab_18_12_01.sql` script by using the `XDBOE` database connection.

```
create table PURCHASEORDER_TEMP
as
select *
  from PURCHASEORDER
/
drop table PURCHASEORDER
/
create table PURCHASEORDER
             of XMLType
             XMLTYPE STORE as OBJECT RELATIONAL
             XMLSCHEMA
"http://localhost:8080/source/schemas/poSource/xsd/purchaseOrder.xsd
"
             Element "PurchaseOrder"
             PARTITION BY LIST (XMLDATA."CostCenter") (
               PARTITION P01 VALUES ('A0','A10','A20','A30'),
               PARTITION P02 VALUES ('A40','A50','A60','A70'),
               PARTITION P03 VALUES ('A80','A90','A100','A110')
             )
             PARALLEL 4
/

-- Rename the Nested Tables used to manage the collections of Action
-- and LineItem elements
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
begin

DBMS_XMLSTORAGE_MANAGE.renameCollectionTable(USER,'PURCHASEORDER',nu
ll,'/PurchaseOrder/Actions/Action/User','ACTION_TABLE',null);

DBMS_XMLSTORAGE_MANAGE.renameCollectionTable(USER,'PURCHASEORDER',nu
ll,'/PurchaseOrder/LineItems/LineItem/Part','LINEITEM_TABLE',null);
end;
/
insert into PURCHASEORDER
select *
  from PURCHASEORDER_TEMP
/
drop table PURCHASEORDER_TEMP
/
create unique index "REFERENCE_INDEX"
              on "PURCHASEORDER" ("SYS_NC00008$")
/
create index "USER_INDEX"
       on "PURCHASEORDER" ("SYS_NC00019$")
       Local
/
create index "PART_NUMBER_QUANTITY_INDEX"
       on "LINEITEM_TABLE" ("SYS_NC00008$","Quantity")
       Local
/
call Dbms_Stats.Gather_Schema_Stats(User)
/
```

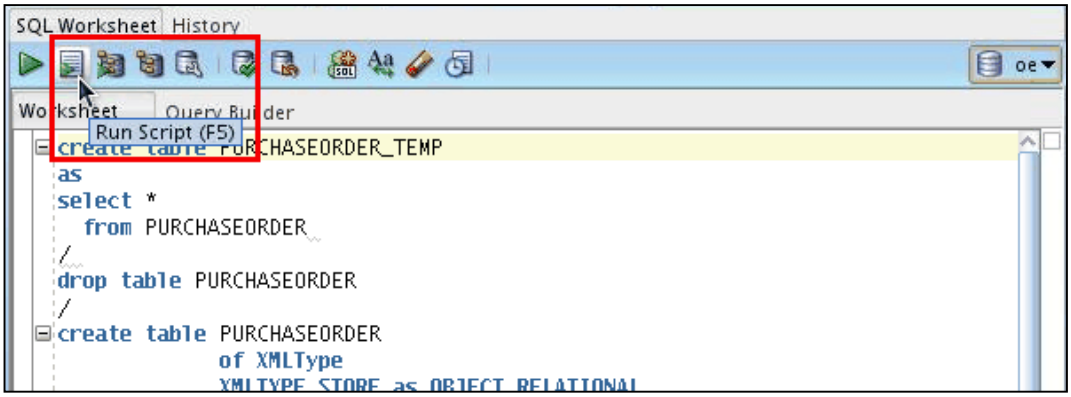
This script uses a `Create Table As Select (CTAS)` operation to make a copy of the content of the existing `PURCHASEORDER` table. It then drops the `PURCHASEORDER` table and re-creates it as a partitioned table. The table is partitioned based on the value of the `User` element. Note that the partition key has to be specified by using object-relational syntax. The script then reloads the documents into the `PURCHASEORDER` table and re-creates the indexes.

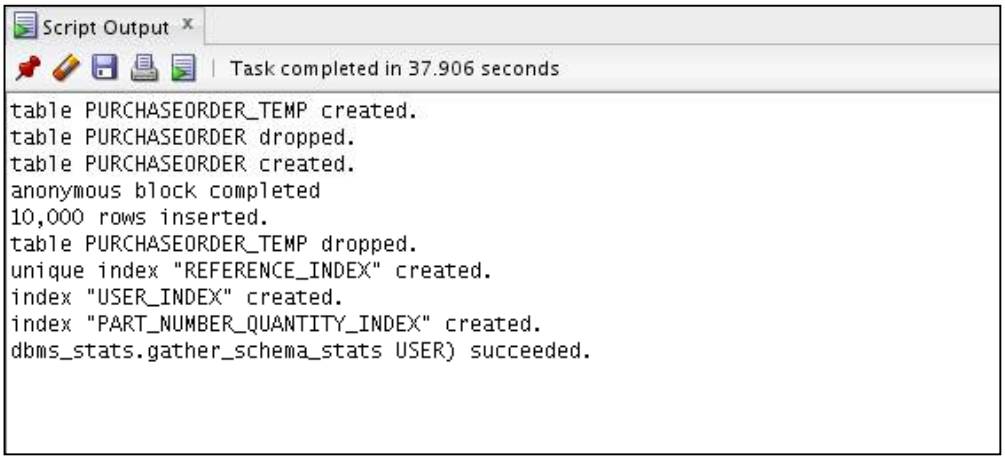
`REFERENCE_INDEX` is created as a global index, while indexes `USER_INDEX` and `PART_NUMBER_QUANTIY_INDEX` are created as local indexes.

Oracle University and Error : You are not a Valid Partner use only

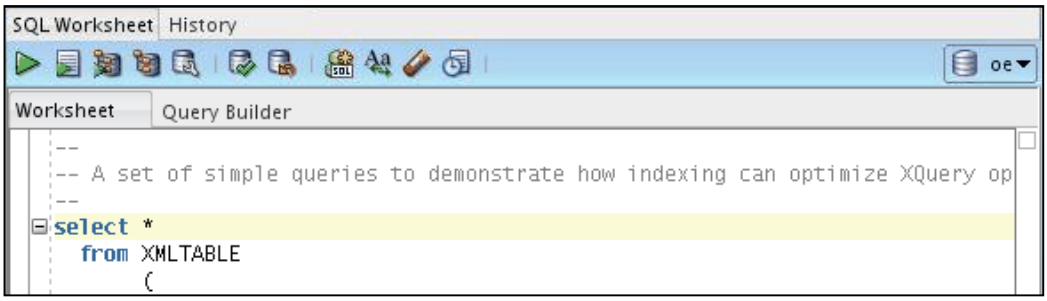
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED





2. Open the `lab_18_10_01.sql` script.



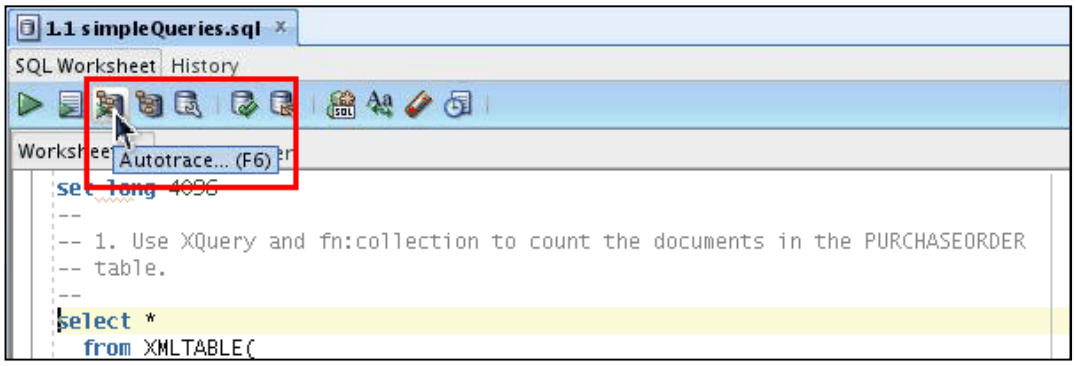
3. Use Autotrace to generate the execution plan for each of the seven queries. The Autotrace output shows that the table is now partitioned, parallel execution occurs where appropriate, and partition pruning takes place for XQuery operations that include a predicate on the partition key.

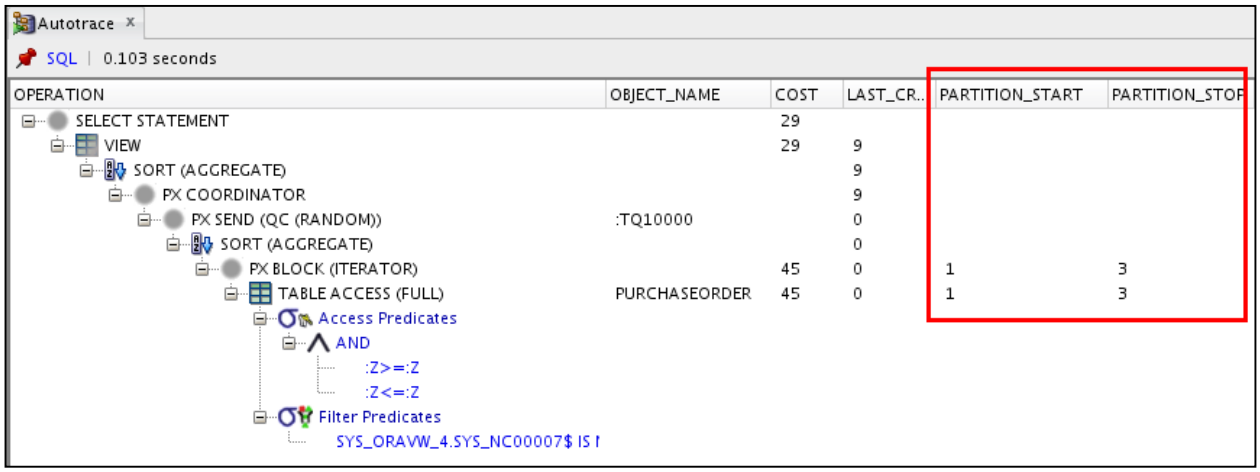
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

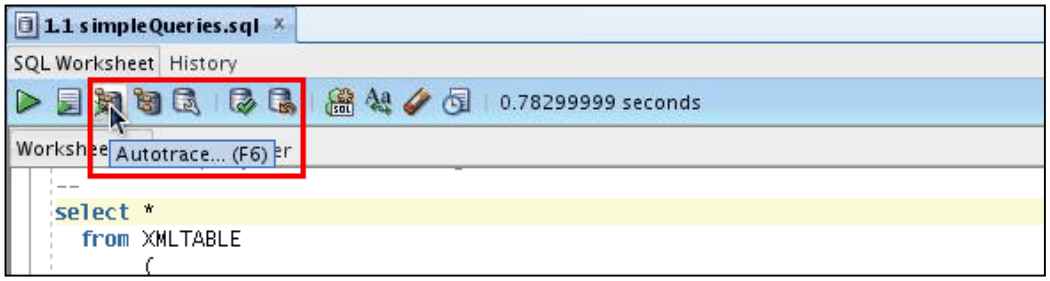
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

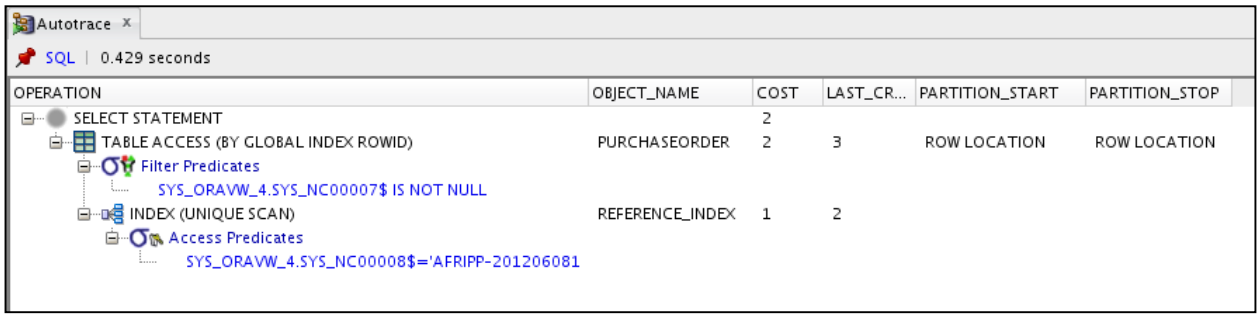
Query 1:





Query 2:



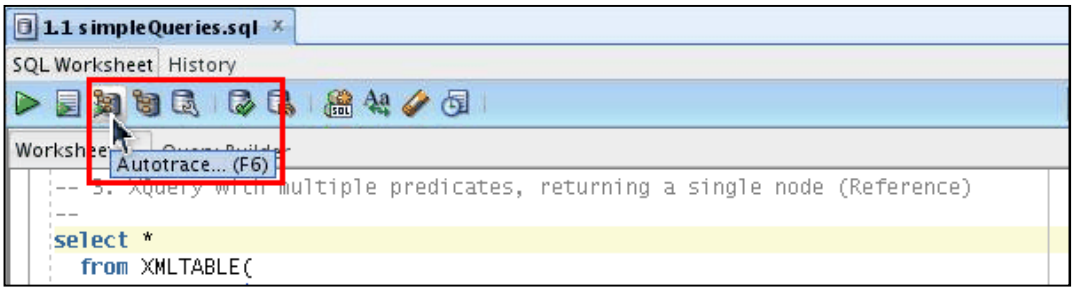


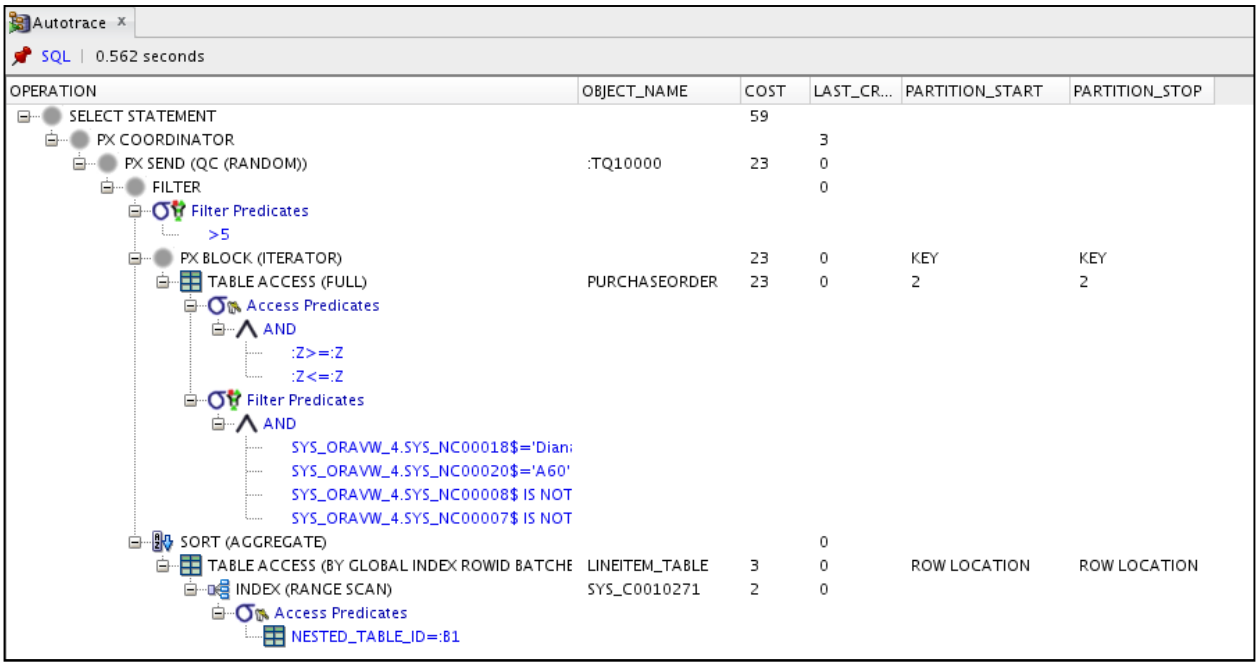
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

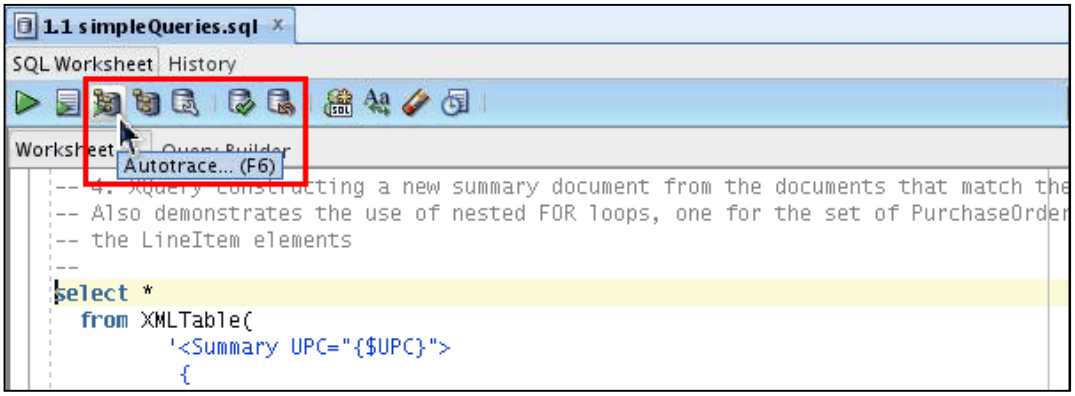
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Query 3:





Query 4:



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Autotrace

SQL | 0.392 seconds

OPERATION	OBJECT_NAME	COST	LAST_CR...	PARTITION_START	PARTITION_STOP
SELECT STATEMENT		19			
SORT (GROUP BY)			28		
NESTED LOOPS			28		
NESTED LOOPS		17	22		
PARTITION REFERENCE (ALL)		11	12	1	3
TABLE ACCESS (BY LOCAL INDEX ROWID BATCHED	LINEITEM_TABLE	11	12	1	3
INDEX (RANGE SCAN)	PART_NUMBER_Q...	4	6	1	3
Access Predicates					
AND					
SYS_NC00008$='707729113751'					
Quantity>3					
Quantity IS NOT NULL					
PARTITION HASH (ITERATOR)		0	10	KEY	KEY
INDEX (UNIQUE SCAN)	LINEITEM_TABLE_...	0	10	KEY	KEY
Access Predicates					
NESTED_TABLE_ID=SYS_ORAVW_4.SYS_N					
TABLE ACCESS (BY GLOBAL INDEX ROWID)	PURCHASEORDER	1	6	ROW LOCATION	ROW LOCATION
Filter Predicates					
SYS_ORAVW_4.SYS_NC00007$ IS NOT NULL					
FAST DUAL		2	0		

Query 5:

1.1 simpleQueries.sql

SQL Worksheet History

Worksheet Autotrace... (F6)

```
--
-- 5. Use XMLSerialize to format the XMLType and serialize it as a CLOB.
--    Allows result to be viewed in products that do not support XMLType.
--    XMLSerialize allows control over the layout of the serialized
--    XML.
--
select XMLSERIALIZE(CONTENT COLUMN_VALUE AS CLOB INDENT SIZE=2)
  from XMLTable(
         '<Summary UPC="{$UPC}">
          {
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Autotrace

SQL | 0.16 seconds

OPERATION	OBJECT_NAME	COST	LAST_CR...	PARTITION_START	PARTITION_STOP
SELECT STATEMENT		19			
SORT (GROUP BY)			28		
NESTED LOOPS			28		
NESTED LOOPS		17	22		
PARTITION REFERENCE (ALL)		11	12	1	3
TABLE ACCESS (BY LOCAL INDEX ROWID BATCHEI	LINEITEM_TABLE	11	12	1	3
INDEX (RANGE SCAN)	PART_NUMBER_Q...	4	6	1	3
Access Predicates					
AND					
SYS_NC00008$='707729113751'					
Quantity>3					
Quantity IS NOT NULL					
PARTITION HASH (ITERATOR)		0	10	KEY	KEY
INDEX (UNIQUE SCAN)	LINEITEM_TABLE_...	0	10	KEY	KEY
Access Predicates					
NESTED_TABLE_ID=SYS_ORAVW_4.SYS_N					
TABLE ACCESS (BY GLOBAL INDEX ROWID)	PURCHASEORDER	1	6	ROW LOCATION	ROW LOCATION
Filter Predicates					
SYS_ORAVW_4.SYS_NC00007$ IS NOT NULL					
FAST DUAL		2	0		

Query 6:

1.1 simpleQueries.sql

SQL Worksheet History

0.36300001 seconds

Worksheet Autotrace... (F6)

```
--
-- 6. Using XMLTable to create an in-line relational view from the documents that
--
select *
  from xmlTable(
```

Autotrace

SQL | 0.552 seconds

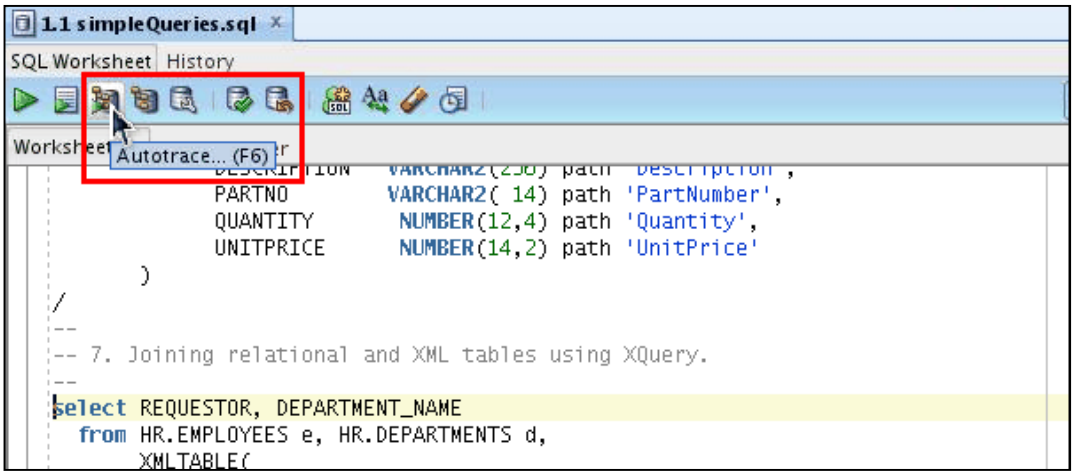
OPERATION	OBJECT_NAME	COST	LAST_CR...	PARTITION_START	PARTITION_STOP
SELECT STATEMENT		17			
VIEW		17	28		
COUNT			28		
NESTED LOOPS			28		
NESTED LOOPS		17	22		
PARTITION REFERENCE (ALL)		11	12	1	3
TABLE ACCESS (BY LOCAL INDEX ROWID BATC	LINEITEM_TABLE	11	12	1	3
INDEX (RANGE SCAN)	PART_NUMBER_Q...	4	6	1	3
Access Predicates					
AND					
SYS_NC00008$='7077291137					
Quantity>3					
Quantity IS NOT NULL					
PARTITION HASH (ITERATOR)		0	10	KEY	KEY
INDEX (UNIQUE SCAN)	LINEITEM_TABLE_...	0	10	KEY	KEY
Access Predicates					
NESTED_TABLE_ID=SYS_ORAVW_4.SY					
TABLE ACCESS (BY GLOBAL INDEX ROWID)	PURCHASEORDER	1	6	ROW LOCATION	ROW LOCATION
Filter Predicates					
SYS_ORAVW_4.SYS_NC00007$ IS NOT NULL					

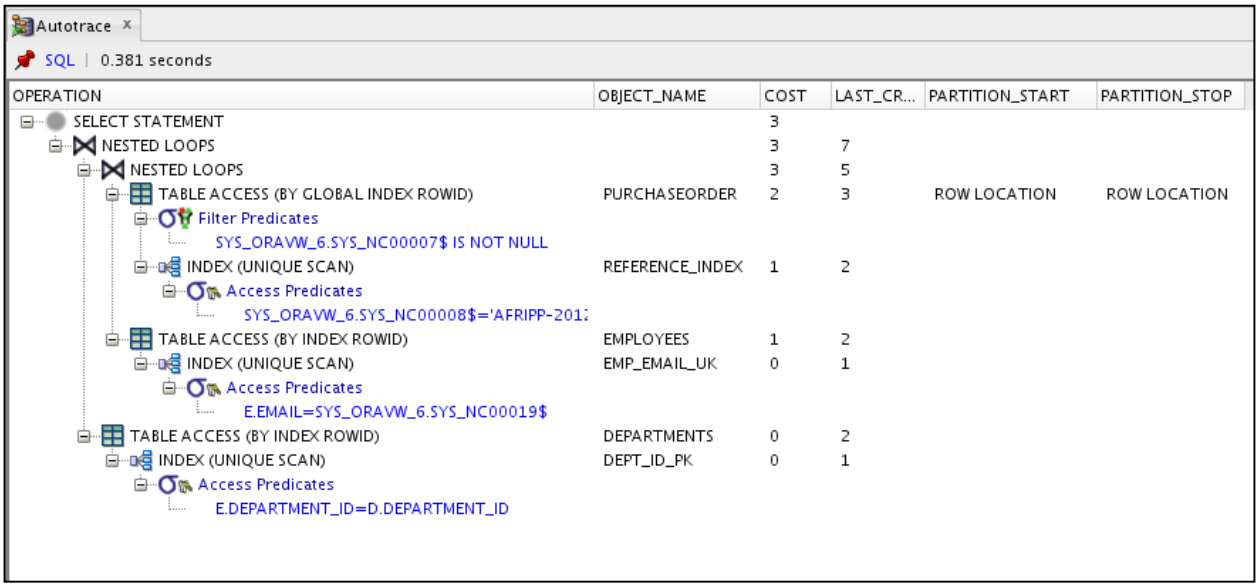
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Query 7:





Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-13: Relational Access to XML Content

Overview

Oracle's implementation of the SQL language provides many powerful features that are not yet supported by the XQuery standard. Also there are many tools (and developers) that are not yet able to work effectively with XML content and XQuery. One of the key benefits of XML DB is that it provides relational tools and programmers with the ability to use SQL to work directly with XML content, enabling all the features of SQL to be used with XML and preserving existing investments in relational technology and infrastructure.

In this practice, you use the combination of XQuery and `XMLTable` to enable efficient SQL operations on XML content.

Assumptions

Creating Relational Views of XML Content

Tasks

1. Open the `lab_18_13_01.sql` script.

```
-- Create a Master View, from elements that occur at most once per
-- document

create or replace view PURCHASEORDER_MASTER_VIEW
as
  select m.*
    from PURCHASEORDER p,
         xmlTable
         (
            '$p/PurchaseOrder'
            passing p.OBJECT_VALUE as "p"
            columns
            REFERENCE            path  'Reference/text()',
            REQUESTOR            path  'Requestor/text()',
            USERID               path  'User/text()',
            COSTCENTER           path  'CostCenter/text()',
            SHIP_TO_NAME         path
'ShippingInstructions/name/text()',
            SHIP_TO_ADDRESS      path
'ShippingInstructions/address/text()',
            SHIP_TO_PHONE        path
'ShippingInstructions/telephone/text()',
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

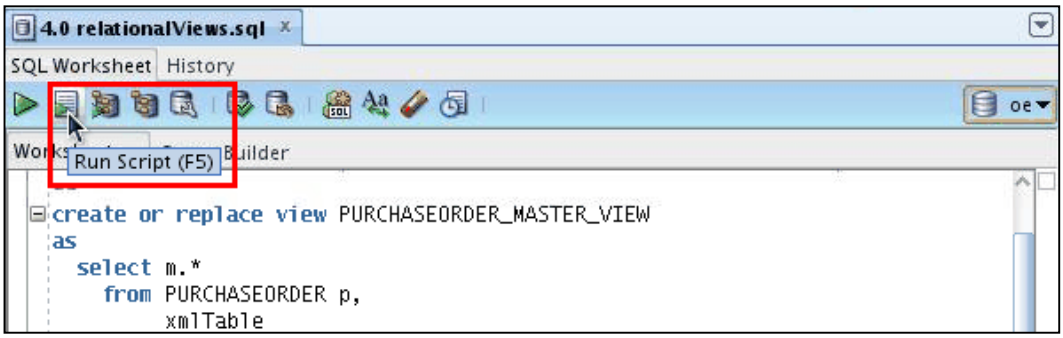
```
            INSTRUCTIONS         path  'SpecialInstructions/text()'
         ) m
/

-- Create a Detail View, from the contents of the LineItem
-- collection. LineItem can appear more than once is a document. The
-- rows in this view can be joined with the rows in the previous
-- view using REFERENCE, which is common to both views.

create or replace view PURCHASEORDER_DETAIL_VIEW
as
  select m.REFERENCE, l.*
    from PURCHASEORDER p,
         xmlTable
         (
            '$p/PurchaseOrder'
            passing p.OBJECT_VALUE as "p"
            columns
            REFERENCE            path  'Reference/text()',
            LINEITEMS    xmlType path  'LineItems/LineItem'
         ) m,
         xmlTable
         (
           '$l/LineItem'
           passing m.LINEITEMS as "l"
           columns
           ITEMNO         path '@ItemNumber',
           DESCRIPTION    path 'Part/@Description',
           PARTNO         path 'Part/text()',
           QUANTITY       path 'Quantity',
           UNITPRICE      path 'Part/@UnitPrice'
         ) l
/
```

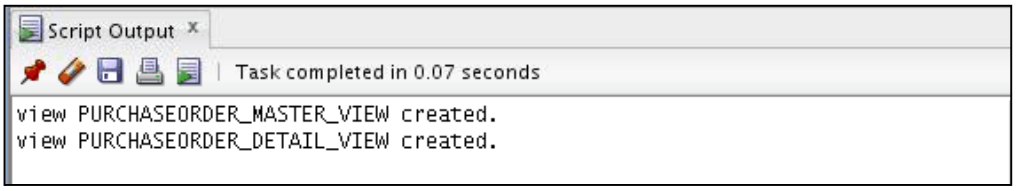
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

2. Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon (or press the F5 key) to execute the script. The output is displayed in the Script Output tab.





This script uses XQuery and `XMLTable` to create two views that provide relational access to XML content. These views provide a typical **MASTER-DETAIL** relationship that can be used to access the contents of the XML by using SQL. The views use XQuery to map scalar values from the XML into the columns defined by the view.

The first view provides access to values from elements and attributes that occur at most once per document. It can be considered the master table.

The second view provides access to values from the collection of `LineItem` elements that can occur multiple times per document. It can be considered the detail. This view is created by using nested `XMLTable` operators.

The first `XMLTable` operator projects out the values that form the primary key for each document and the set of repeating elements. The set of repeating elements forms the input to the second `XMLTable` operator, which generates one row for each of the repeating elements.

Since the second `XMLTable` processes the output of the first `XMLTable`, a correlated join takes place. This ensures that the rows produced by the secondary `XMLTable` are joined with the corresponding rows from the primary `XMLTable`.

Oracle University and Error : You are not a Valid Partner use only

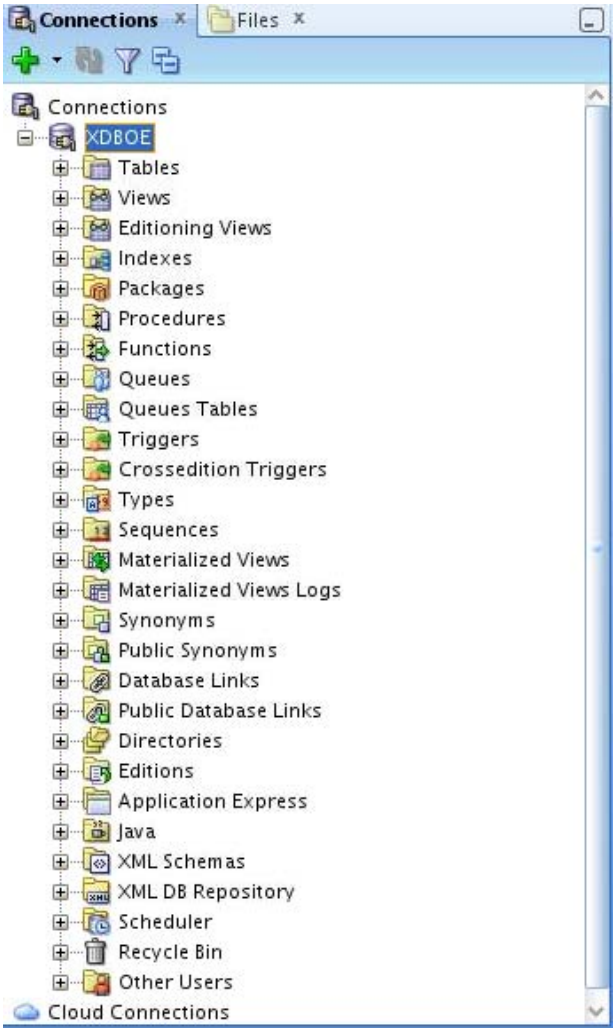
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Examine the Views in SQL Developer

Tasks

1. In this section, you examine the views that you created in the previous step, which you can use with any tool that supports accessing relational data via SQL views. Select the Connections tab, and then click the + icon to the left of the `XDBOE` connection to expand it.

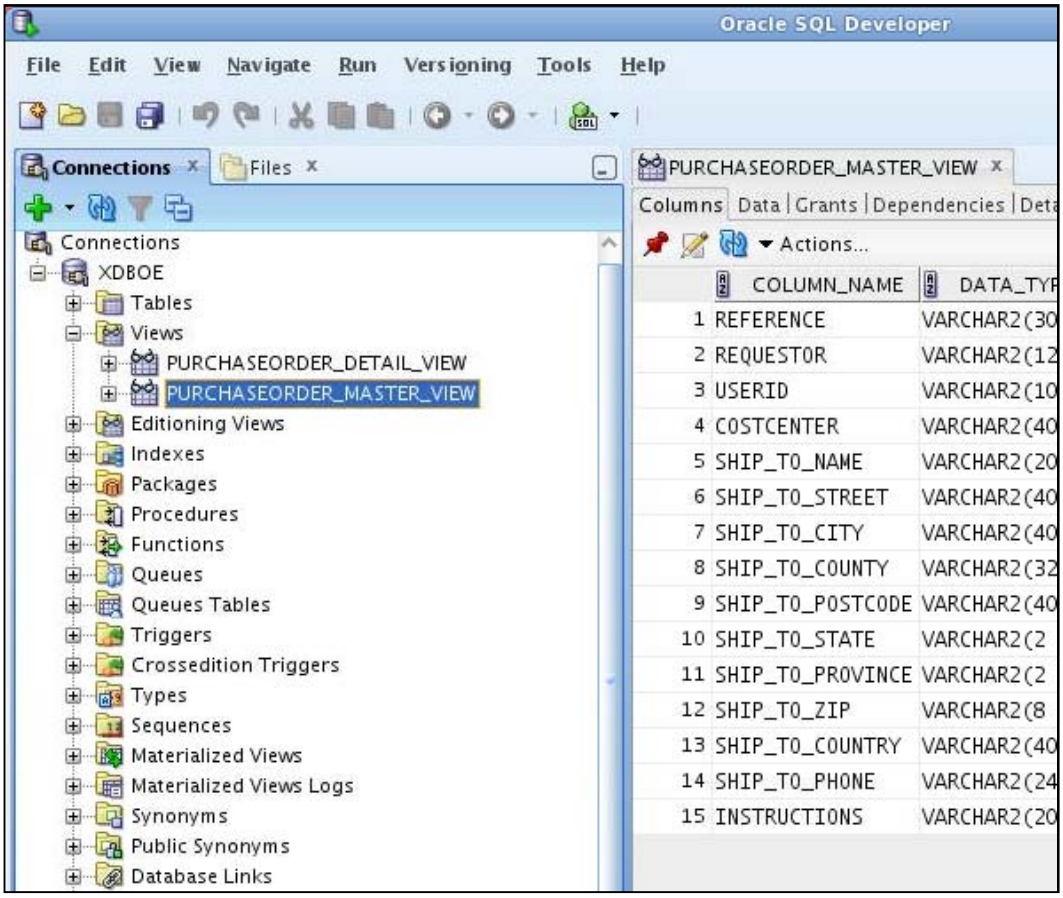


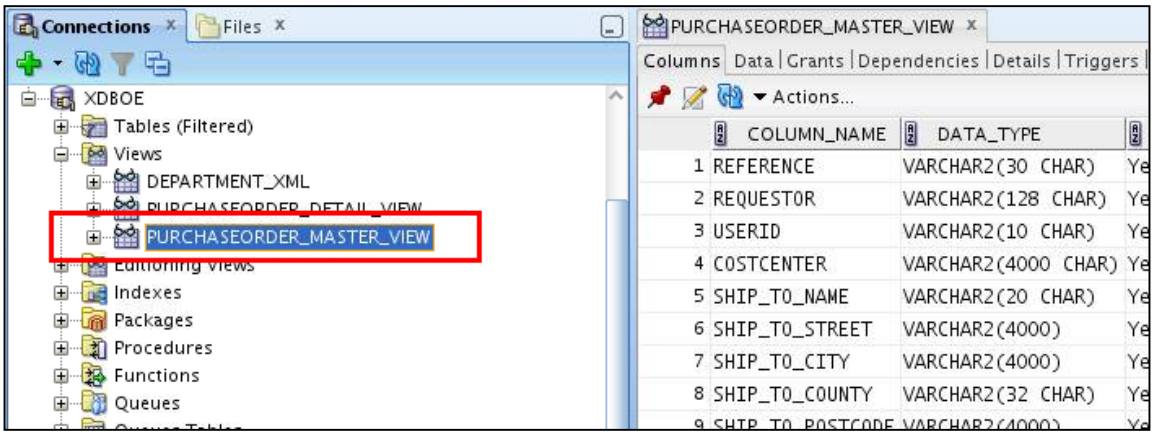
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

2. Expand Views, and then select `PURCHASEORDER_MASTER_VIEW`.





Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

3. In the right pane, click the Columns tab, if not already selected.
   Note: SQL Developer considers the view a standard relational view.

PURCHASEORDER_MASTER_VIEW

Columns | Data | Grants | Dependencies | Details | Triggers | SQL

Actions...

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS	INSERTABLE	UP
1	REFERENCE	VARCHAR2(30 CHAR)	Yes	(null)	1	(null)	NO	NO
2	REQUESTOR	VARCHAR2(128 CHAR)	Yes	(null)	2	(null)	NO	NO
3	USERID	VARCHAR2(10 CHAR)	Yes	(null)	3	(null)	NO	NO
4	COSTCENTER	VARCHAR2(4000 CHAR)	Yes	(null)	4	(null)	NO	NO
5	SHIP_TO_NAME	VARCHAR2(20 CHAR)	Yes	(null)	5	(null)	NO	NO
6	SHIP_TO_STREET	VARCHAR2(4000)	Yes	(null)	6	(null)	NO	NO
7	SHIP_TO_CITY	VARCHAR2(4000)	Yes	(null)	7	(null)	NO	NO
8	SHIP_TO_COUNTY	VARCHAR2(32 CHAR)	Yes	(null)	8	(null)	NO	NO
9	SHIP_TO_POSTCODE	VARCHAR2(4000)	Yes	(null)	9	(null)	NO	NO
10	SHIP_TO_STATE	VARCHAR2(2 CHAR)	Yes	(null)	10	(null)	NO	NO
11	SHIP_TO_PROVINCE	VARCHAR2(2 CHAR)	Yes	(null)	11	(null)	NO	NO
12	SHIP_TO_ZIP	VARCHAR2(8 CHAR)	Yes	(null)	12	(null)	NO	NO
13	SHIP_TO_COUNTRY	VARCHAR2(4000)	Yes	(null)	13	(null)	NO	NO
14	SHIP_TO_PHONE	VARCHAR2(24 CHAR)	Yes	(null)	14	(null)	NO	NO
15	INSTRUCTIONS	VARCHAR2(2048 CHAR)	Yes	(null)	15	(null)	NO	NO

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Click the Data tab.

   Note: SQL Developer is able to retrieve the data, even though the content is stored in the database as an `XMLType` table.

PURCHASEORDER_MASTER_VIEW

Columns | Data | Grants | Dependencies | Details | Triggers | SQL

Sort.. | Filter:

	REFERENCE	REQUESTOR	USERID	COSTCENTER	SHIP_TO_NAME	SHIP_TO_STREE
1	OTUVAULT-20120715145625544PDT	Oliver Tuvault	OTUVAULT	A80	Oliver Tuvault	Magdalen Centre
2	OTUVAULT-20120719160438877PDT	Oliver Tuvault	OTUVAULT	A80	Oliver Tuvault	Magdalen Centre
3	OTUVAULT-20120725172618964PDT	Oliver Tuvault	OTUVAULT	A80	Oliver Tuvault	Magdalen Centre
4	OTUVAULT-20120725205754510PDT	Oliver Tuvault	OTUVAULT	A80	Oliver Tuvault	Magdalen Centre
5	OTUVAULT-20120725211405150PDT	Oliver Tuvault	OTUVAULT	A80	Oliver Tuvault	Magdalen Centre
6	OTUVAULT-20120728222103851PDT	Oliver Tuvault	OTUVAULT	A80	Oliver Tuvault	Magdalen Centre
7	OTUVAULT-20120731210038802PDT	Oliver Tuvault	OTUVAULT	A80	Oliver Tuvault	Magdalen Centre
8	AHUTTON-20120605164727868PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
9	AHUTTON-20120606190957171PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
10	AHUTTON-20120607182930490PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
11	AHUTTON-20120608122925911PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
12	AHUTTON-20120610142133322PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
13	AHUTTON-2012061115520614PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
14	AHUTTON-2012061218120194PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
15	AHUTTON-20120612213218426PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
16	AHUTTON-2012062113585289PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
17	AHUTTON-20120625214204544PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
18	AHUTTON-20120629150142163PDT	Alyssa Hutton	AHUTTON	A80	Alyssa Hutton	Magdalen Centre
19	JRUSSEL-20120617174541470PDT	John Russell	JRUSSEL	A80	John Russell	Magdalen Centre
20	SSEWALL-20120729172641184PDT	Sarath Sewall	SSEWALL	A80	Sarath Sewall	Magdalen Centre
21	NCAMBRAU-20120614221711429PDT	Nanette Cambrault	NCAMBRAU	A80	Nanette Cambrault	Magdalen Centre
22	NCAMBRAU-20120616222230327PDT	Nanette Cambrault	NCAMBRAU	A80	Nanette Cambrault	Magdalen Centre
23	NCAMBRAU-20120627224925890PDT	Nanette Cambrault	NCAMBRAU	A80	Nanette Cambrault	Magdalen Centre

. . .

5. Repeat steps 2, 3, and 4 for the `PURCHASEORDER_DETAIL_VIEW`.

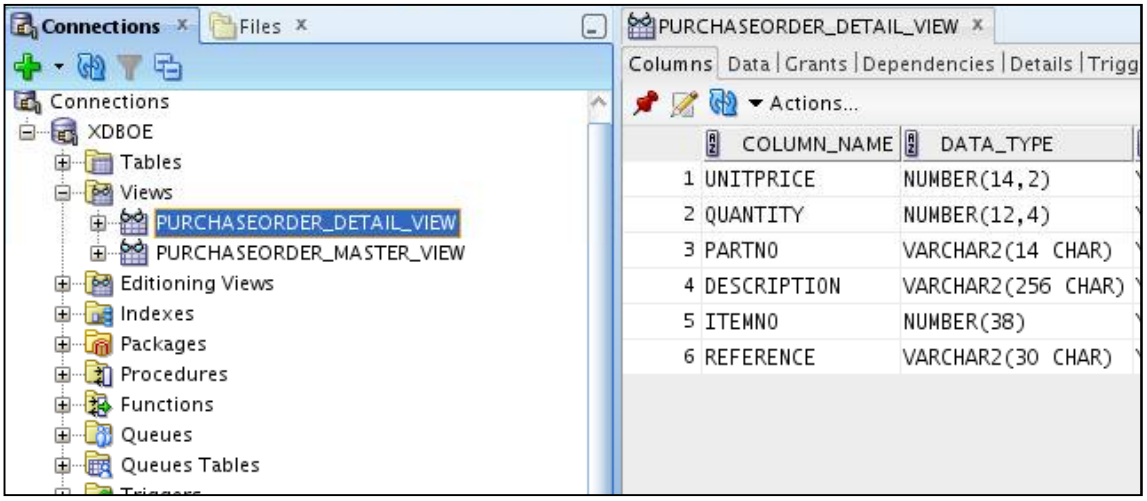
   This demonstrates that any ID, reporting tool, or Business Intelligence tool that understands the relational paradigm and SQL can use this technique to access content stored in `XMLType` tables within Oracle Database.

Oracle University and Error : You are not a Valid Partner use only

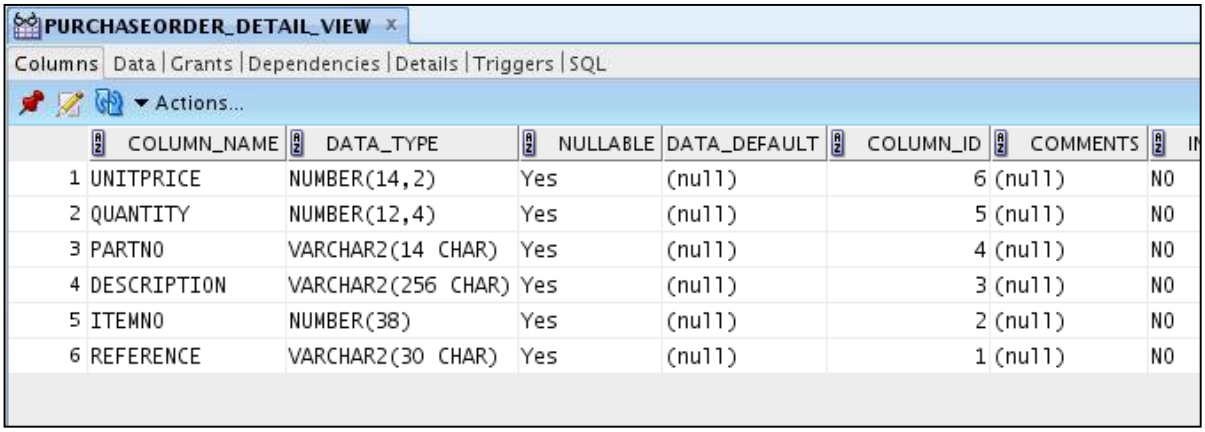
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Expand Views in the `XDBOE` connection, and then select `PURCHASEORDER_DETAIL_VIEW`.



6. In the right pane, click the Columns tab if not already selected. Note that SQL Developer sees the view as a standard relational view.

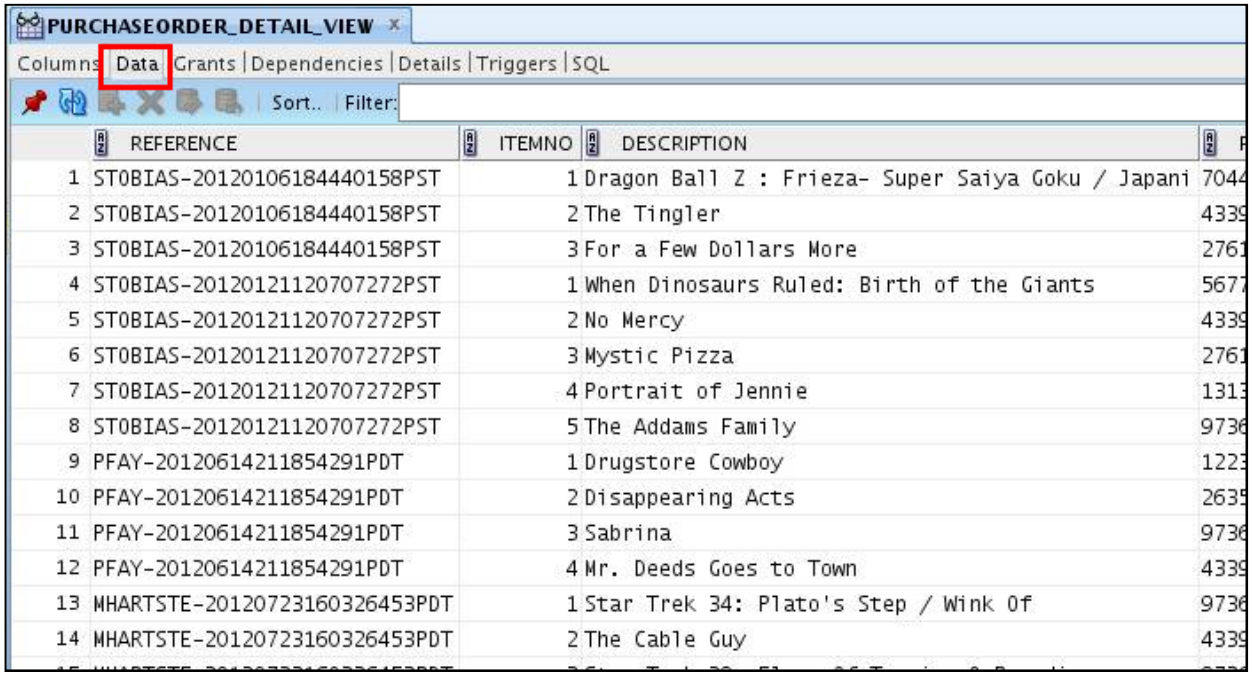


Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

7. Click the Data tab. Note again that SQL Developer is able to retrieve the data, even though the content is stored in the database as an `XMLType` table.



	REFERENCE	ITEMNO	DESCRIPTION	F
1	STOBIAS-20120106184440158PST	1	Dragon Ball Z : Frieza- Super Saiya Goku / Japani	7044
2	STOBIAS-20120106184440158PST	2	The Tingler	4339
3	STOBIAS-20120106184440158PST	3	For a Few Dollars More	2761
4	STOBIAS-20120121120707272PST	1	When Dinosaurs Ruled: Birth of the Giants	5677
5	STOBIAS-20120121120707272PST	2	No Mercy	4339
6	STOBIAS-20120121120707272PST	3	Mystic Pizza	2761
7	STOBIAS-20120121120707272PST	4	Portrait of Jennie	1313
8	STOBIAS-20120121120707272PST	5	The Addams Family	9736
9	PFAY-20120614211854291PDT	1	Drugstore Cowboy	1223
10	PFAY-20120614211854291PDT	2	Disappearing Acts	2635
11	PFAY-20120614211854291PDT	3	Sabrina	9736
12	PFAY-20120614211854291PDT	4	Mr. Deeds Goes to Town	4339
13	MHARTSTE-20120723160326453PDT	1	Star Trek 34: Plato's Step / Wink Of	9736
14	MHARTSTE-20120723160326453PDT	2	The Cable Guy	4339

. . .

Performing SQL Operations on XML Content

Tasks

8. In this section, you demonstrate how relational views of XML enable simple and advanced SQL statements to operate on XML content stored in the Oracle database. Open the `lab_18_13_08.sql` script.

```
-- Execute a simple SQL query over the relational view of XML
-- content showing the use of SQL Group By. XQuery 1.0 did not
-- support the concept of group by

select COSTCENTER, count(*)
  From Purchaseorder_Master_View
  group by COSTCENTER
/

-- Simple Query showing a join between the master and detail views
-- with relational predicates on both views.
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

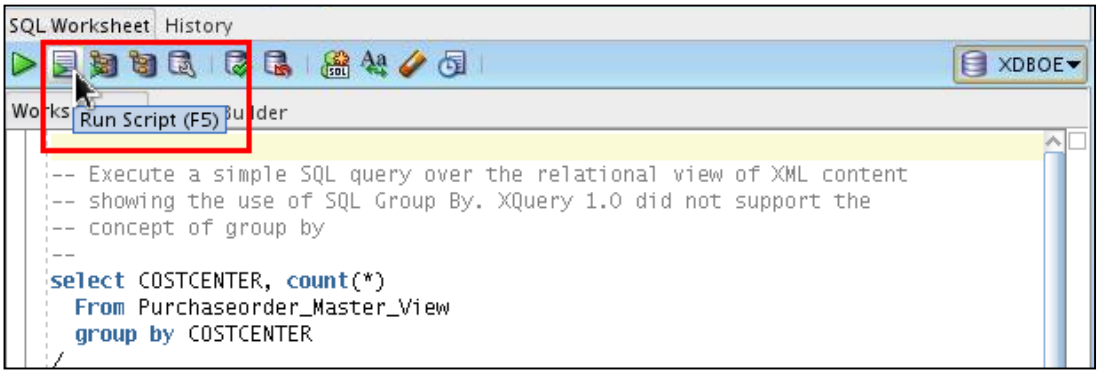
```
select m.REFERENCE, INSTRUCTIONS, ITEMNO, PARTNO, DESCRIPTION,
QUANTITY, UNITPRICE
  from PURCHASEORDER_MASTER_VIEW m, PURCHASEORDER_DETAIL_VIEW d
 where m.REFERENCE = d.REFERENCE
   and m.REQUESTOR = 'Steven King'
   and d.QUANTITY  > 7
   And D.Unitprice > 25.00
/

-- Simple Query showing a join between the master and detail views
-- with relational predicate on detail view.

Select M.Reference, L.Itemno, L.Partno, L.Description
  from PURCHASEORDER_MASTER_VIEW m, PURCHASEORDER_DETAIL_VIEW l
 Where M.Reference = L.Reference
   and l.PARTNO in ('717951010490', '43396713994', '12236123248')
/

-- SQL Query on detail view making use of SQL Analytics
-- functionality not provided by XQuery

Select Partno, Count(*) "Orders", Quantity "Copies"
  from PURCHASEORDER_DETAIL_VIEW
 where PARTNO in ('717951010490', '43396713994', '12236123248')
 group by rollup(PARTNO, QUANTITY)
/

-- SQL Query on detail view making use of SQL Analytics
-- functionality not provided by XQuery

Select Partno, Reference, Quantity, QUANTITY -
LAG(QUANTITY,1,QUANTITY) over (ORDER BY
SUBSTR(REFERENCE,INSTR(REFERENCE,'-') + 1)) as DIFFERENCE
  from PURCHASEORDER_DETAIL_VIEW
 Where Partno = '43396713994'
 order by  SUBSTR(REFERENCE,INSTR(REFERENCE,'-') + 1) DESC
/
```

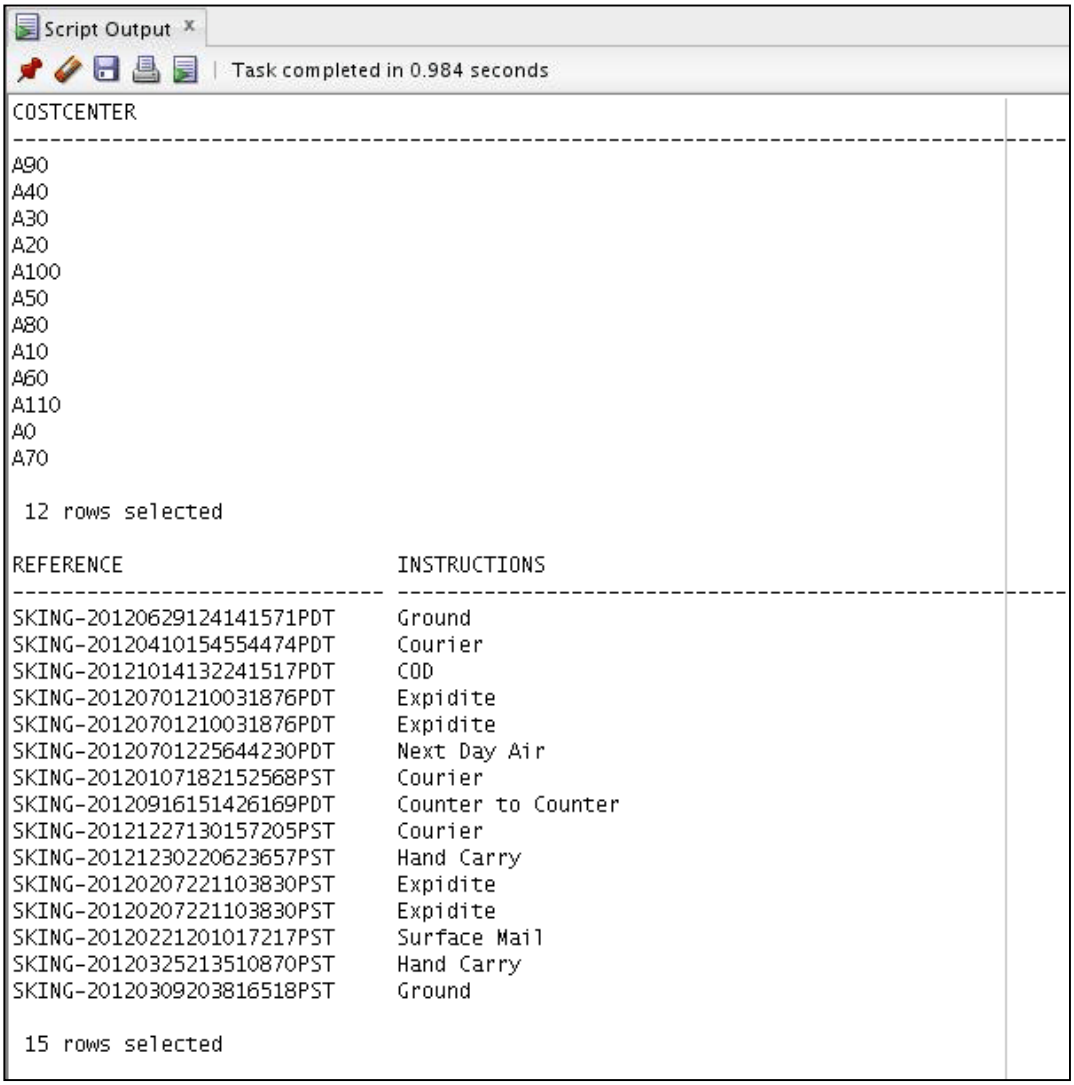
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

9. Run the script by using the `XDBOE` database connection. None of the queries contain a reference to XML constructs. Advanced SQL functionality like group by, group by (rollup()) and lag can be used to analyze XML content. Predicates can be supplied as part of the SQL queries. The output is displayed on the Script Output tab.





Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Script Output

Task completed in 0.984 seconds

```
REFERENCE                          ITEMNO PARTNO         DESCRIPTION
------------------------------ ---------- -------------- ------------------------------
STOBIAS-2012062817564628PDT             1 12236123248    Down from the Mountain
WTAYLOR-20121110211100813PST            2 12236123248    Down from the Mountain
JPATEL-20120211224743995PST             4 12236123248    Down from the Mountain
SKING-20120701210031876PDT              2 12236123248    Down from the Mountain
SKING-20120309203816518PST              4 43396713994    Manhattan Murder Mystery
SKING-20120701210031876PDT              1 717951010490   The Bodyguard from Beijing
HBLOOM-20120715130255162PDT             3 43396713994    Manhattan Murder Mystery
IMIKKILI-20120706161052639PDT           5 12236123248    Down from the Mountain
DOCONNEL-20120505144942493PDT           1 12236123248    Down from the Mountain
KCHUNG-20120725130042715PDT             4 43396713994    Manhattan Murder Mystery
KMOURGOS-20120315161325495PST           6 43396713994    Manhattan Murder Mystery
SMARKLE-2012020513310514PST             7 43396713994    Manhattan Murder Mystery
JAMRLOW-20120813202643842PDT            2 43396713994    Manhattan Murder Mystery
AWALSH-20121110151234770PST             4 43396713994    Manhattan Murder Mystery
KGEE-20120514150055950PDT               1 717951010490   The Bodyguard from Beijing
DAUSTIN-20120722150412667PDT            1 717951010490   The Bodyguard from Beijing
LBISSOT-20120718194758220PDT            3 717951010490   The Bodyguard from Beijing
LBISSOT-2012053013002005PDT             3 717951010490   The Bodyguard from Beijing
ABULL-20120331224817816PST              4 717951010490   The Bodyguard from Beijing
PKAUFLIN-20120626213117279PDT           5 717951010490   The Bodyguard from Beijing
PHALL-2012051222512837 9PDT             1 12236123248    Down from the Mountain
OTUVAULT-20120115185501127PST           1 12236123248    Down from the Mountain
EBATES-2012053119342862PDT              3 12236123248    Down from the Mountain
EABEL-20120706201113772PDT              1 43396713994    Manhattan Murder Mystery
JTAYLOR-20121218122235847PST            3 12236123248    Down from the Mountain

 25 rows selected
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Script Output

Task completed in 0.984 seconds

```
 25 rows selected

PARTNO             Orders     Copies
-------------- ---------- ----------
12236123248             2          2
12236123248             2          3
12236123248             1          5
12236123248             1          7
12236123248             3          8
12236123248             1          9
12236123248            10
43396713994             2          2
43396713994             1          5
43396713994             2          7
43396713994             1          8
43396713994             2          9
43396713994             8
717951010490            1          2
717951010490            2          5
717951010490            1          6
717951010490            2          7
717951010490            1          9
717951010490            7
                       25

 20 rows selected

PARTNO         REFERENCE                        QUANTITY DIFFERENCE
-------------- ------------------------------ ---------- ----------
43396713994    AWALSH-20121110151234770PST             9          1
43396713994    JAMRLOW-20120813202643842PDT            8          6
43396713994    KCHUNG-20120725130042715PDT             2         -5
43396713994    HBLOOM-20120715130255162PDT             7          2
43396713994    EABEL-20120706201113772PDT              5          3
43396713994    KMOURGOS-20120315161325495PST           2         -7
43396713994    SKING-20120309203816518PST              9          2
43396713994    SMARKLE-2012020513310514PST             7          0

 8 rows selected
```

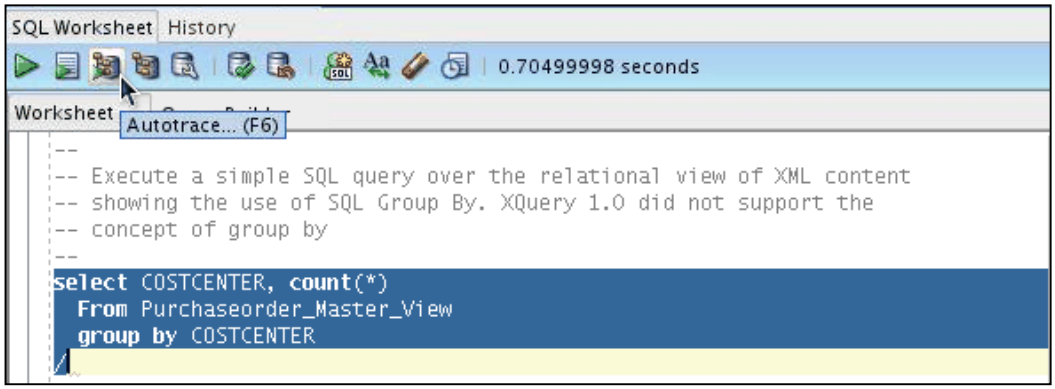
Oracle University and Error : You are not a Valid Partner use only

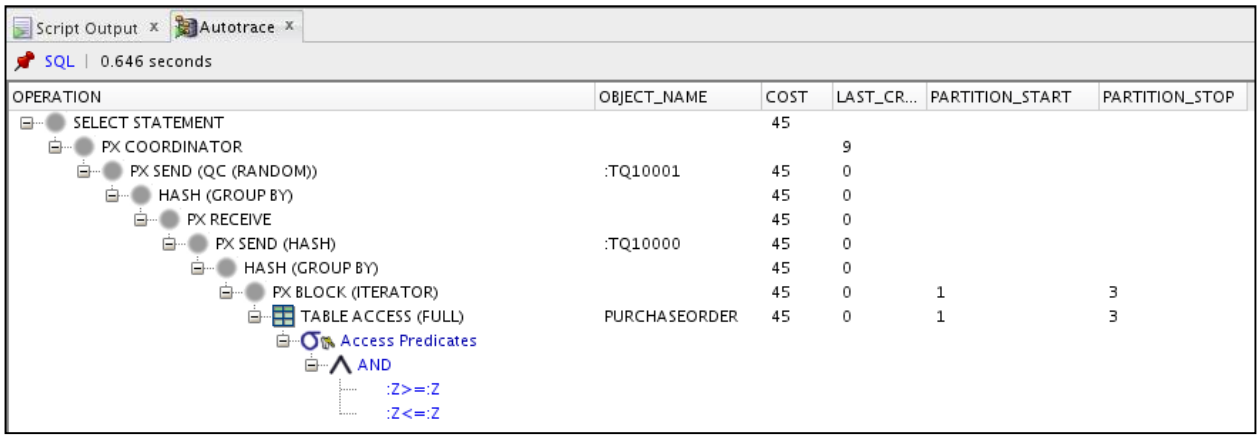
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

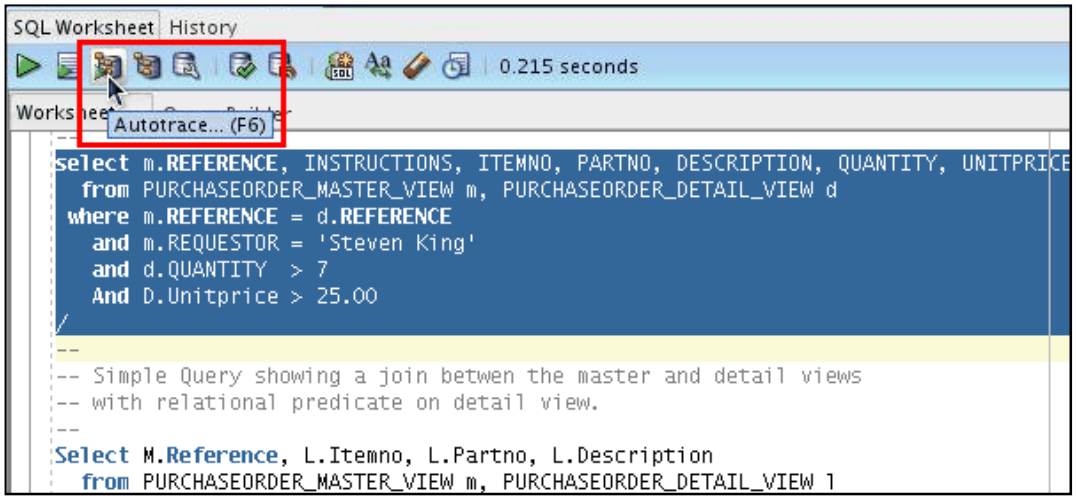
10. Use Autotrace to generate the execution plan for each of the queries. The execution plan shows that all of the queries are executed as SQL queries over the tables that manage the XML content. The XML abstraction has been completely bypassed in these cases.

Query 1:





Query 2:



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Script Output | Autotrace

SQL | 0.238 seconds

OPERATION	OBJECT_NAME	COST	LAST_CR...	PARTITION_START	PARTITION_STOP
SELECT STATEMENT		123			
PX COORDINATOR			9		
PX SEND (QC (RANDOM))	:TQ10000		0		
NESTED LOOPS			0		
NESTED LOOPS		123	0		
NESTED LOOPS		71	0		
PX BLOCK (ITERATOR)		45	0	1	3
TABLE ACCESS (FULL)	PURCHASEORDER	45	0	1	3
Access Predicates					
AND					
:Z>=:Z					
:Z<=:Z					
Filter Predicates					
P.SYS_NC00018$='Steven King'					
TABLE ACCESS (BY GLOBAL INDEX ROWID)	PURCHASEORDER	1	0	ROW LOCATION	ROW LOCATION
INDEX (UNIQUE SCAN)	REFERENCE_INDEX	0	0		
Access Predicates					
P.SYS_NC00008$=P.SYS_NC0000					
INDEX (RANGE SCAN)	SYS_C0010271	1	0		
Access Predicates					
SYS_ALIAS_5.NESTED_TABLE_ID=SYS_ALI					
TABLE ACCESS (BY GLOBAL INDEX ROWID)	LINEITEM_TABLE	2	0	ROW LOCATION	ROW LOCATION
Filter Predicates					
AND					
Quantity>7					
SYS_NC00010$>25					

Query 3:

SQL Worksheet | History

0.57800001 seconds

Worksheet | Query Builder

Autotrace... (F6)

```
-- showing a join between the master and detail views
-- with relational predicate on detail view.
--
Select M.Reference, L.Itemno, L.Partno, L.Description
  from PURCHASEORDER_MASTER_VIEW m, PURCHASEORDER_DETAIL_VIEW l
 where M.Reference = L.Reference
   and l.PARTNO in ('717951010490', '43396713994', '12236123248')
/
--
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Script Output | Autotrace

SQL | 0.186 seconds

OPERATION	OBJECT_NAME	COST	LAST_CR...	PARTITION_START	PARTITION_STOP
SELECT STATEMENT		58			
NESTED LOOPS		58	131		
NESTED LOOPS		58	104		
PARTITION REFERENCE (ALL)		32	38	1	3
INLIST ITERATOR			38		
TABLE ACCESS (BY LOCAL INDEX ROWID BATCHEI	LINEITEM_TABLE	32	38	1	3
INDEX (RANGE SCAN)	PART_NUMBER_Q...	6	13	1	3
Access Predicates					
OR					
SYS_NC00008$='12236123248'					
SYS_NC00008$='43396713994'					
SYS_NC00008$='717951010490'					
PARTITION HASH (ITERATOR)		1	66	KEY	KEY
TABLE ACCESS (BY GLOBAL INDEX ROWID)	PURCHASEORDER	1	66	ROW LOCATION	ROW LOCATION
INDEX (UNIQUE SCAN)	LINEITEM_TABLE_...	0	41	KEY	KEY
Access Predicates					
SYS_ALIAS_5.NESTED_TABLE_ID=SYS_ALI					
INDEX (UNIQUE SCAN)	REFERENCE_INDEX	0	27		
Access Predicates					
P.SYS_NC00008$=P.SYS_NC00008$					

Query 4:

SQL Worksheet | History

0.42199999 seconds

Worksheet | Query Builder

Autotrace... (F6)

```
-- functionality not provided by XQuery
--
Select Partno, Count(*) "Orders", Quantity "Copies"
  from PURCHASEORDER_DETAIL_VIEW
 where PARTNO in ('717951010490', '43396713994', '12236123248')
 group by rollup(PARTNO, QUANTITY)
/
```

Script Output | Autotrace

SQL | 0.209 seconds

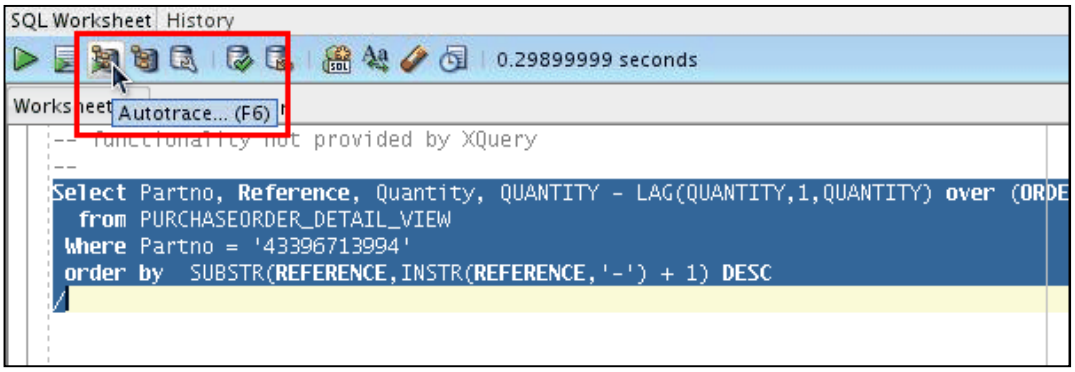
OPERATION	OBJECT_NAME	COST	LAST_CR...	PARTITION_START	PARTITION_STOP
SELECT STATEMENT		32			
SORT (GROUP BY ROLLUP)		32	79		
NESTED LOOPS		32	79		
PARTITION REFERENCE (ALL)		32	38	1	3
INLIST ITERATOR			38		
TABLE ACCESS (BY LOCAL INDEX ROWID BATCHEI	LINEITEM_TABLE	32	38	1	3
INDEX (RANGE SCAN)	PART_NUMBER_Q...	6	13	1	3
Access Predicates					
OR					
SYS_NC00008$='12236123248'					
SYS_NC00008$='43396713994'					
SYS_NC00008$='717951010490'					
PARTITION HASH (ITERATOR)		0	41	KEY	KEY
INDEX (UNIQUE SCAN)	LINEITEM_TABLE_...	0	41	KEY	KEY
Access Predicates					
SYS_ALIAS_3.NESTED_TABLE_ID=P.SYS_NC00					

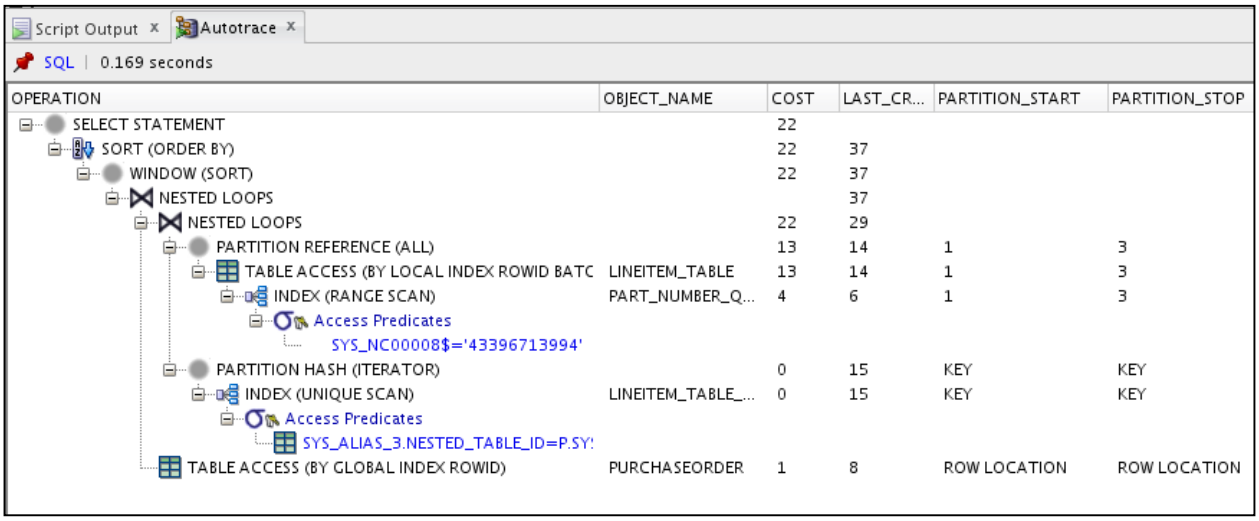
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Query 5:





OPERATION	OBJECT_NAME	COST	LAST_CR...	PARTITION_START	PARTITION_STOP
SELECT STATEMENT		22			
SORT (ORDER BY)		22	37		
WINDOW (SORT)		22	37		
NESTED LOOPS			37		
NESTED LOOPS		22	29		
PARTITION REFERENCE (ALL)		13	14	1	3
TABLE ACCESS (BY LOCAL INDEX ROWID BATC	LINEITEM_TABLE	13	14	1	3
INDEX (RANGE SCAN)	PART_NUMBER_Q...	4	6	1	3
Access Predicates					
SYS_NC00008$='43396713994'					
PARTITION HASH (ITERATOR)		0	15	KEY	KEY
INDEX (UNIQUE SCAN)	LINEITEM_TABLE_...	0	15	KEY	KEY
Access Predicates					
SYS_ALIAS_3.NESTED_TABLE_ID=P.SY!					
TABLE ACCESS (BY GLOBAL INDEX ROWID)	PURCHASEORDER	1	8	ROW LOCATION	ROW LOCATION

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-14: XML Access to Relational Content

Overview

The previous practice demonstrated how to access XML content from SQL. Oracle XML DB also allows XQuery to operate directly on relational data. Two distinct metaphors can be used for this, one is XML-centric, and the other is SQL-centric. Both techniques are based on the concept of creating an XML representation of the data in relational tables. There is no difference in terms of efficiency; the decision about which approach to adopt is a matter of personal preference.

- The first technique uses a canonical mapping to generate an XML representation of each row in a relational table. Non-canonical XML representations of the relational data can then be created by using XQuery to transform from the canonical representation into the required format.
- The second approach involves a SQL-centric approach, based on the SQL/XML publishing functions. The SQL/XML publishing functions allow a SQL statement to return one or more XML documents rather than a traditional tabular result set.

Both techniques can be used to create XML views that allow XQuery-based operations to be performed on relational data.

Assumptions

Using XQuery to Generate XML Documents from Relational Tables

Tasks

1. Open the `lab_18_14_01.sql` script. The first example uses XQuery and `fn:collection` to generate XML documents from relational tables.

```
-- Simple XQuery showing how fn:collection can be used to generate
-- a simple XML document from each row in a relational table.
set long 5000

select xmlserialize(Document Column_Value As Clob Indent Size=2)
  from XMLTable (
          'fn:collection("oradb:/HR/DEPARTMENTS")'
        )
 where rownum < 3
/
-- Using XQuery and fn:collection to create XML documents
-- from relational tables.

select *
  from XMLTable
        (
          'for $d in fn:collection("oradb:/HR/DEPARTMENTS")/ROW,
```

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
                    $l in fn:collection("oradb:/HR/LOCATIONS")/ROW,
                    $c in fn:collection("oradb:/HR/COUNTRIES")/ROW
                               where $d/LOCATION_ID = $l/LOCATION_ID
                                          and $l/COUNTRY_ID =
$c/COUNTRY_ID
                               return
                                    <Department DepartmentId=
"{$d/DEPARTMENT_ID/text()}" >

   <Name>{$d/DEPARTMENT_NAME/text()}</Name>
                                                  <Location>

   <Address>{$l/STREET_ADDRESS/text()}</Address>

   <City>{$l/CITY/text()}</City>

   <State>{$l/STATE_PROVINCE/text()}</State>

   <Zip>{$l/POSTAL_CODE/text()}</Zip>

   <Country>{$c/COUNTRY_NAME/text()}</Country>
                                                  </Location>
                                                  <EmployeeList>
                                                  {
                                                  for $e in
fn:collection("oradb:/HR/EMPLOYEES")/ROW,
                                                             $m in
fn:collection("oradb:/HR/EMPLOYEES")/ROW,
                                                                   $j in
fn:collection("oradb:/HR/JOBS")/ROW
                                                             where
$e/DEPARTMENT_ID = $d/DEPARTMENT_ID
                                                             and
$j/JOB_ID = $e/JOB_ID

   and $m/EMPLOYEE_ID = $e/MANAGER_ID

   return

   <Employee employeeNumber="{$e/EMPLOYEE_ID/text()}" >

         <FirstName>{$e/FIRST_NAME/text()}</FirstName>

         <LastName>{$e/LAST_NAME/text()}</LastName>

         <EmailAddress>{$e/EMAIL/text()}</EmailAddress>
```

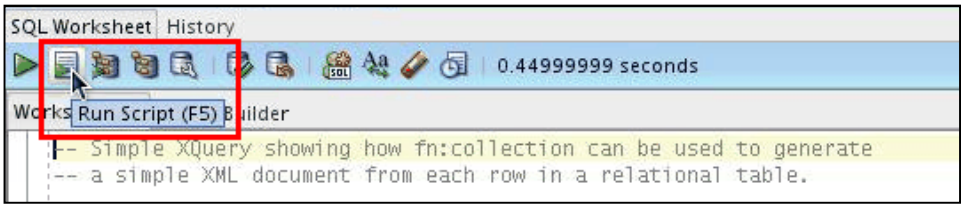
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
        <Telephone>{$e/PHONE_NUMBER/text()}</Telephone>

        <StartDate>{$e/HIRE_DATE/text()}</StartDate>

        <JobTitle>{$j/JOB_TITLE/text()}</JobTitle>

        <Salary>{$e/SALARY/text()}</Salary>

        <Manager>{$m/LAST_NAME/text(), ", ",
$m/FIRST_NAME/text()}</Manager>

   <Commission>{$e/COMMISSION_PCT/text()}</Commission>

   </Employee>
                                                    }
                                                    </EmployeeList>
                                 </Department>'
        ) D
  where rownum < 3
/
```

2. Run the `lab_18_14_01.sql` script by using the `XDBOE` database connection.
   Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon (or press the F5 key) to execute the script. The output is displayed in the Script Output tab.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

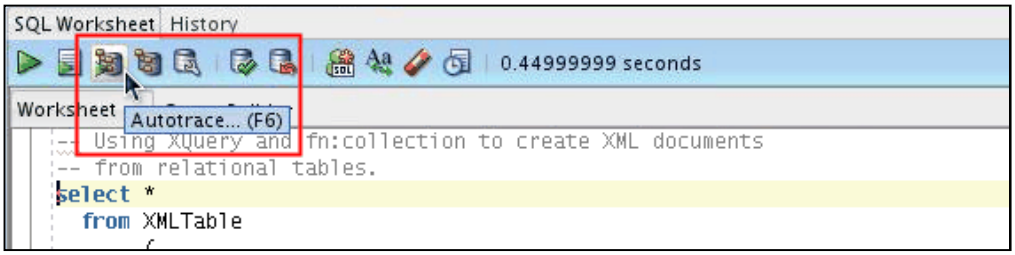
THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
Script Output
Task completed in 1.486 seconds
XMLSERIALIZE(DOCUMENTCOLUMN_VALUEASCLOBINDENTSIZE=2)
--------------------------------------------------------------------------------
<ROW>
  <DEPARTMENT_ID>10</DEPARTMENT_ID>
  <DEPARTMENT_NAME>Administration</DEPARTMENT_NAME>
  <MANAGER_ID>200</MANAGER_ID>
  <LOCATION_ID>1700</LOCATION_ID>
</ROW>


<ROW>
  <DEPARTMENT_ID>20</DEPARTMENT_ID>
  <DEPARTMENT_NAME>Marketing</DEPARTMENT_NAME>
  <MANAGER_ID>201</MANAGER_ID>
  <LOCATION_ID>1800</LOCATION_ID>
</ROW>



COLUMN_VALUE
--------------------------------------------------------------------------------
<Department DepartmentId="20"><Name>Marketing</Name><Location><Address xmlns:xsi="htt
<Department DepartmentId="70"><Name>Public Relations</Name><Location><Address xmlns:x
```

The first query shows the output of the simple canonical mapping used to generate XML documents from rows in a relational table. Each document has a root node called ROW. The root node contains one element for each column in the table. The element names will be generated from the column names; character escaping will be used when the column name contains characters that are not permitted as an XML name. XQuery pragmas can be used to determine how null values are handled.

The second query shows how XQuery can be used to transform the canonical mapping into a more useful XML format. This example also shows how this technique can be used to generate documents containing content that comes from multiple relational tables.

3. Use Autotrace to generate the execution plan for the second query.



Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Autotrace

SQL | 0.178 seconds

OPERATION	OBJECT_NAME	COST	LAST_CR_...
SELECT STATEMENT		13	
SORT (AGGREGATE)			26
HASH JOIN		4	26
Access Predicates			
EMPLOYEE_ID=MANAGER_ID			
NESTED LOOPS			10
NESTED LOOPS		2	7
TABLE ACCESS (BY INDEX ROWID BA	EMPLOYEES	2	4
INDEX (RANGE SCAN)	EMP_DEPARTME...	1	2
Access Predicates			
DEPARTMENT_ID=:B1			
INDEX (UNIQUE SCAN)	JOB_ID_PK	0	3
Access Predicates			
JOB_ID=JOB_ID			
TABLE ACCESS (BY INDEX ROWID)	JOBS	0	3
VIEW	index$_join$_0...	2	16
HASH JOIN			16
Access Predicates			
ROWID=ROWID			
INDEX (FAST FULL SCAN)	EMP_NAME_IX	1	8
INDEX (FAST FULL SCAN)	EMP_EMP_ID_PK	1	8
COUNT (STOPKEY)			9
Filter Predicates			
ROWNUM<3			
NESTED LOOPS			9
NESTED LOOPS		1	8
NESTED LOOPS		1	5
INDEX (FULL SCAN)	COUNTRY_C_ID...	1	1
TABLE ACCESS (BY INDEX ROWID BA	LOCATIONS	0	4
INDEX (RANGE SCAN)	LOC_COUNTRY...	0	3
Access Predicates			
COUNTRY_ID=COUNTR			
INDEX (RANGE SCAN)	DEPT_LOCATIO...	0	3
Access Predicates			
LOCATION_ID=LOCATION_ID			
TABLE ACCESS (BY INDEX ROWID)	DEPARTMENTS	0	1

The Autotrace output shows that a purely relational execution plan is used to generate the XML. As expected, the execution plan for the SQL/XML approach is just as efficient as the execution plan for the XQuery-based option.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Using SQL/XML Functions to Generate XML Documents from Relational Tables

Tasks

4. The second example in this section demonstrates how to use the SQL/XML functions to generate XML documents from relational tables. Open the `lab_18_14_04.sql` script.

```
select xmlElement
       (
         "Department",
         xmlAttributes( d.DEPARTMENT_ID as "DepartmentId"),
         xmlElement("Name", d.DEPARTMENT_NAME),
         xmlElement
         (
           "Location",
           xmlForest
           (
              STREET_ADDRESS as "Address", CITY as "City",
STATE_PROVINCE as "State",
              POSTAL_CODE as "Zip",COUNTRY_NAME as "Country"
           )
         ),
         xmlElement
         (
           "EmployeeList",
           (
             select xmlAgg
                    (
                      xmlElement
                      (
                        "Employee",
                        xmlAttributes ( e.EMPLOYEE_ID as
"employeeNumber" ),
                        xmlForest
                        (
                          e.FIRST_NAME as "FirstName", e.LAST_NAME
as "LastName", e.EMAIL as "EmailAddress",
                          e.PHONE_NUMBER as "Telephone", e.HIRE_DATE
as "StartDate", j.JOB_TITLE as "JobTitle",
                          e.SALARY as "Salary", m.FIRST_NAME || ' '
|| m.LAST_NAME as "Manager"
                        ),
```

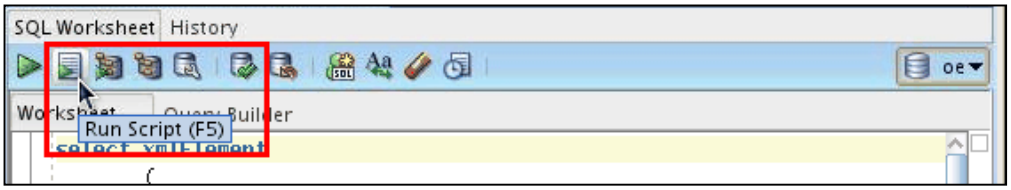
Oracle University and Error : You are not a Valid Partner use only

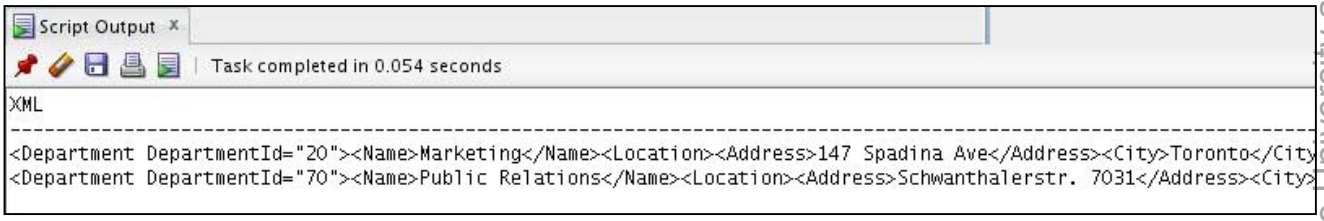
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
                        xmlElement ( "Commission", e.COMMISSION_PCT
)
                      )
                    )
                from HR.EMPLOYEES e, HR.EMPLOYEES m, HR.JOBS j
               where e.DEPARTMENT_ID = d.DEPARTMENT_ID
                 and j.JOB_ID = e.JOB_ID
                 and m.EMPLOYEE_ID = e.MANAGER_ID
            )
          )
        ) as XML
  from HR.DEPARTMENTS d, HR.COUNTRIES c, HR.LOCATIONS l
 where d.LOCATION_ID = l.LOCATION_ID
   And L.Country_Id  = C.Country_Id
   and rownum < 3
/
```

5. Run the script by using the `XDBOE` database connection.





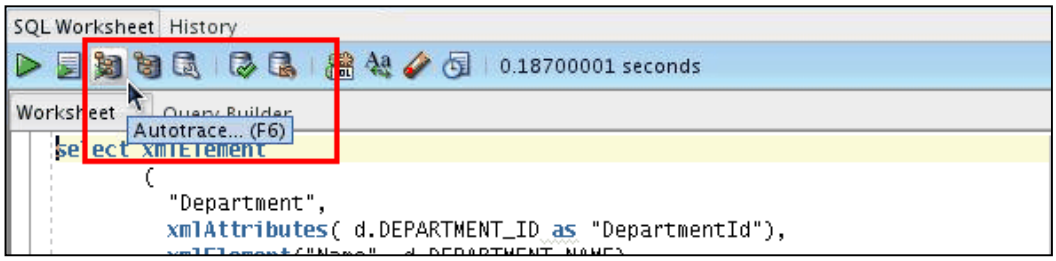
This query shows the SQL/XML equivalent of the second XQuery in the previous example. SQL/XML uses a template-based approach to generating XML from relational data. The nesting of the SQL/XML operators describes the shape of the document and how the column values are mapped into text nodes and attributes.

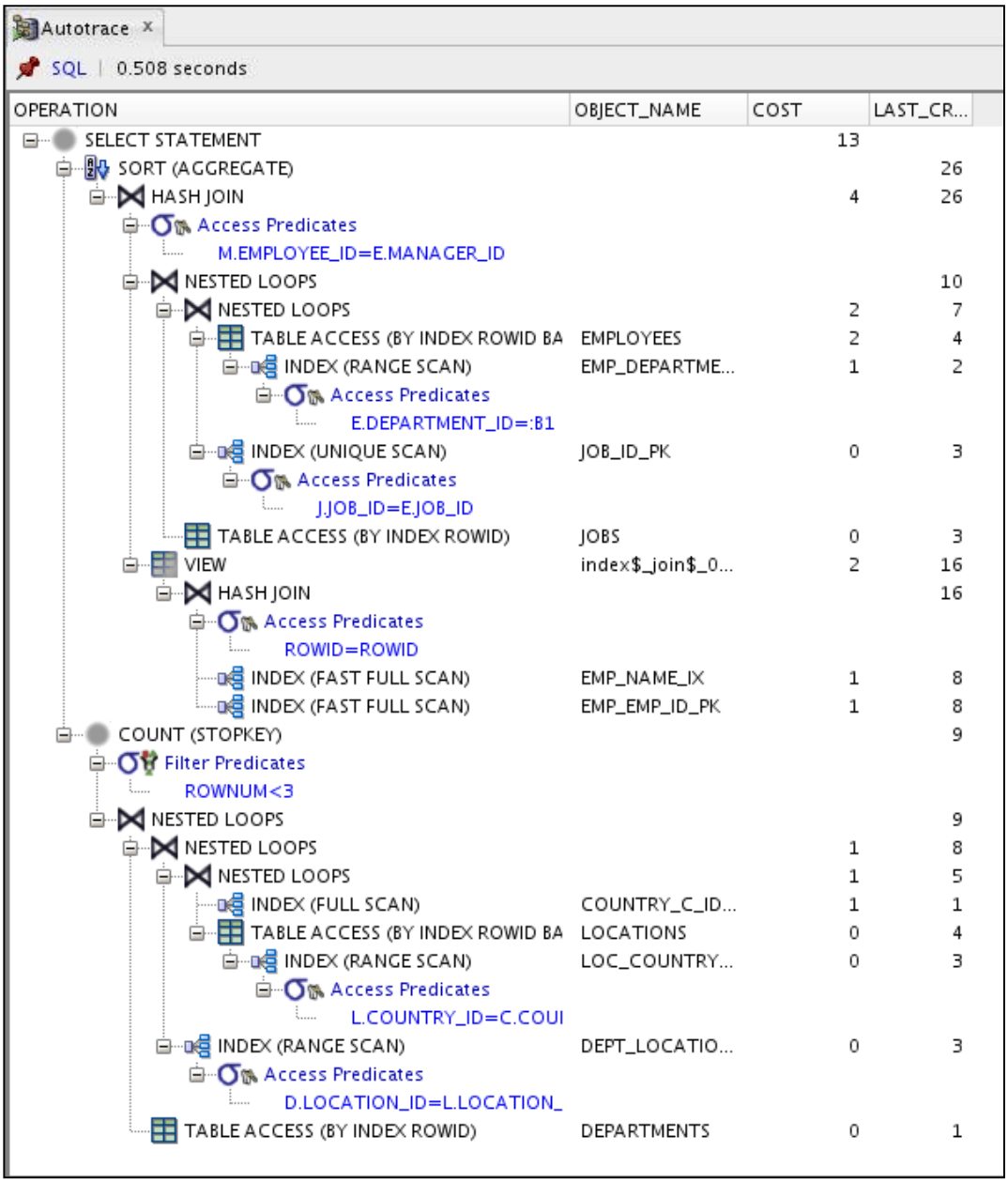
Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

6. Use Autotrace to generate the execution plan for the second query.





OPERATION	OBJECT_NAME	COST	LAST_CR...
SELECT STATEMENT		13	
SORT (AGGREGATE)			26
HASH JOIN		4	26
Access Predicates			
M.EMPLOYEE_ID=E.MANAGER_ID			
NESTED LOOPS			10
NESTED LOOPS		2	7
TABLE ACCESS (BY INDEX ROWID BA	EMPLOYEES	2	4
INDEX (RANGE SCAN)	EMP_DEPARTME...	1	2
Access Predicates			
E.DEPARTMENT_ID=:B1			
INDEX (UNIQUE SCAN)	JOB_ID_PK	0	3
Access Predicates			
J.JOB_ID=E.JOB_ID			
TABLE ACCESS (BY INDEX ROWID)	JOBS	0	3
VIEW	index$_join$_0...	2	16
HASH JOIN			16
Access Predicates			
ROWID=ROWID			
INDEX (FAST FULL SCAN)	EMP_NAME_IX	1	8
INDEX (FAST FULL SCAN)	EMP_EMP_ID_PK	1	8
COUNT (STOPKEY)			9
Filter Predicates			
ROWNUM<3			
NESTED LOOPS			9
NESTED LOOPS		1	8
NESTED LOOPS		1	5
INDEX (FULL SCAN)	COUNTRY_C_ID...	1	1
TABLE ACCESS (BY INDEX ROWID BA	LOCATIONS	0	4
INDEX (RANGE SCAN)	LOC_COUNTRY...	0	3
Access Predicates			
L.COUNTRY_ID=C.COUI			
INDEX (RANGE SCAN)	DEPT_LOCATIO...	0	3
Access Predicates			
D.LOCATION_ID=L.LOCATION_			
TABLE ACCESS (BY INDEX ROWID)	DEPARTMENTS	0	1

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Autotrace output shows that a purely relational execution plan is used to generate the XML. As expected, the execution plan for the SQL/XML approach is just as efficient as the execution plan for the XQuery-based option.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Practice 18-15: Creating a Persistent XML View of Relational Data

Overview

In this practice, you create an XML view of relational data. You use the same SQL/XML query as you did in the previous example; however, the XQuery expression from the first example in this practice could also be used for this purpose. The view is a view of `XMLType`, which means that the view definition must include a mechanism for generating a unique ID for each row in the view. In this example, the ID is supplied by the expression in the `with object id` clause of the `create view` statement. In this case, uniqueness is based on the contents of the `Name` element.

Assumptions

Tasks

1. Open the `lab_18_15_01.sql` script.

```
-- Use SQL/XML to create a persistent XML view of relational content

create or replace view DEPARTMENT_XML of xmltype
with object id
(
  XMLCAST(XMLQUERY('/Department/Name' passing OBJECT_VALUE returning
CONTENT) as VARCHAR2(30))
)
as
select xmlElement
       (
         "Department",
         xmlAttributes( d.DEPARTMENT_ID as "DepartmentId"),
         xmlElement("Name", d.DEPARTMENT_NAME),
         xmlElement
         (
           "Location",
           xmlForest
           (
              STREET_ADDRESS as "Address", CITY as "City",
STATE_PROVINCE as "State",
              POSTAL_CODE as "Zip",COUNTRY_NAME as "Country"
           )
         ),
         xmlElement
         (
           "EmployeeList",
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle University and Error : You are not a Valid Partner use only

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

```
            (
              select xmlAgg
                     (
                       xmlElement
                       (
                         "Employee",
                         xmlAttributes ( e.EMPLOYEE_ID as
"employeeNumber" ),
                         xmlForest
                         (
                           e.FIRST_NAME as "FirstName", e.LAST_NAME
as "LastName", e.EMAIL as "EmailAddress",
                           e.PHONE_NUMBER as "Telephone", e.HIRE_DATE
as "StartDate", j.JOB_TITLE as "JobTitle",
                           e.SALARY as "Salary", m.FIRST_NAME || ' '
|| m.LAST_NAME as "Manager"
                         ),
                         xmlElement ( "Commission", e.COMMISSION_PCT
)
                       )
                     )
                from HR.EMPLOYEES e, HR.EMPLOYEES m, HR.JOBS j
               where e.DEPARTMENT_ID = d.DEPARTMENT_ID
                 and j.JOB_ID = e.JOB_ID
                 and m.EMPLOYEE_ID = e.MANAGER_ID
            )
          )
        ) as XML
  from HR.DEPARTMENTS d, HR.COUNTRIES c, HR.LOCATIONS l
 where d.LOCATION_ID = l.LOCATION_ID
   and l.COUNTRY_ID  = c.COUNTRY_ID
/
```

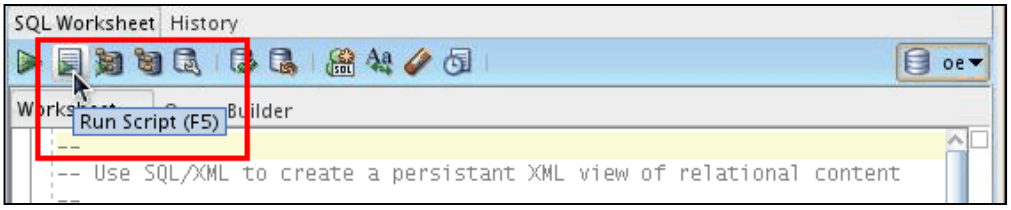
Oracle University and Error : You are not a Valid Partner use only

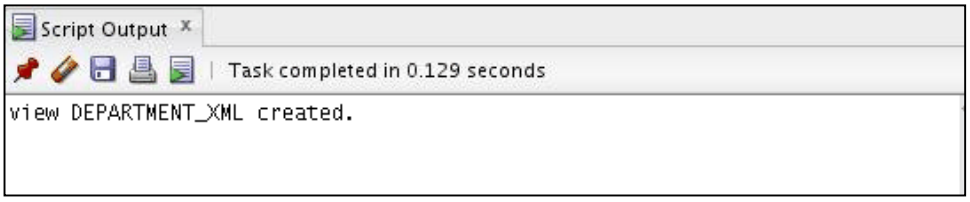
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

2. Run the script by using the `XDBOE` database connection.

   Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon (or press the F5 key) to execute the script. An `XMLType` view called `DEPARTMENT_XML` is created. The output is displayed in the Script Output tab.





3. Open the `lab_18_15_03.sql` script.

```
-- Using XML Views to execute XQuery expressions on relational
-- data

select *
  from XMLTable
       (
         'for $d in
fn:collection("oradb:/XDBOE/DEPARTMENT_XML")/Department[Name="Execut
ive"]
          return $d'
       )
/
select *
  from XMLTable
       (
         'for $d in
fn:collection("oradb:/XDBOE/DEPARTMENT_XML")/Department[EmployeeList
/Employee/LastName="Grant"]/Name
          return $d'
       )
/
```

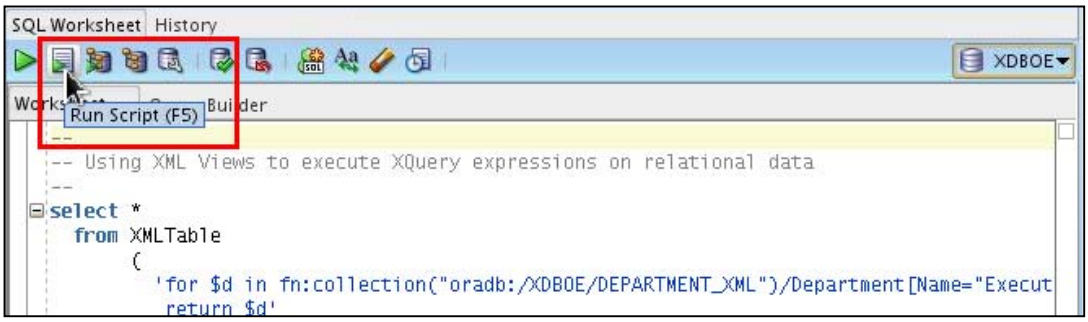
Oracle University and Error : You are not a Valid Partner use only

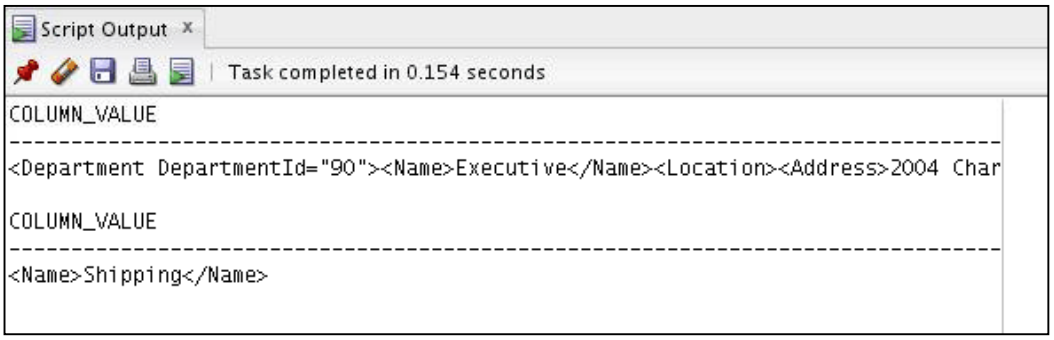
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

4. Run the script by using the `XDBOE` database connection.

   Select the `XDBOE` database connection from the Choose Db Connection drop-down list, and then click the Run Script icon (or press the F5 key) to execute the script. The output is displayed on the Script Output tab.





The `XMLType` view allows XQuery operations to be performed against relational data. In the same way that relational views of XML enable SQL-centric development on top of XML content, XML views of relational data enable XML-centric developers to work with relational data.

5. Use Autotrace to generate the execution plan for the second query.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Autotrace

SQL | 0.284 seconds

OPERATION	OBJECT_NAME	COST	LAST_CR...
SELECT STATEMENT		2	
HASH JOIN (SEMI)		2	9
Access Predicates			
E.DEPARTMENT_ID=D.DEPARTMENT_ID			
NESTED LOOPS			6
NESTED LOOPS		0	5
TABLE ACCESS (BY INDEX ROWID BATCH	LOCATIONS	0	2
INDEX (FULL SCAN)	LOC_COUNTRY...	0	1
Filter Predicates			
L.COUNTRY_ID IS NOT NULL			
INDEX (RANGE SCAN)	DEPT_LOCATIO...	0	3
Access Predicates			
D.LOCATION_ID=L.LOCATION_			
TABLE ACCESS (BY INDEX ROWID)	DEPARTMENTS	0	1
TABLE ACCESS (BY INDEX ROWID BATCHED)	EMPLOYEES	2	3
Filter Predicates			
E.MANAGER_ID IS NOT NULL			
INDEX (RANGE SCAN)	EMP_NAME_IX	1	1
Access Predicates			
E.LAST_NAME='Grant'			

The Autotrace output shows that the execution plan obtained when executing an XQuery on an `XMLType` view of relational data is a purely relational plan, demonstrating that XML DB is able to optimize XQuery operations on relational data in the same way that it can optimize relational operations on XML content.

Oracle University and Error : You are not a Valid Partner use only

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.