Lab 7 : SQL/XPath/XQuery

Download Oracle Developer

https://www.oracle.com/database/sqldeveloper/technologies/download/

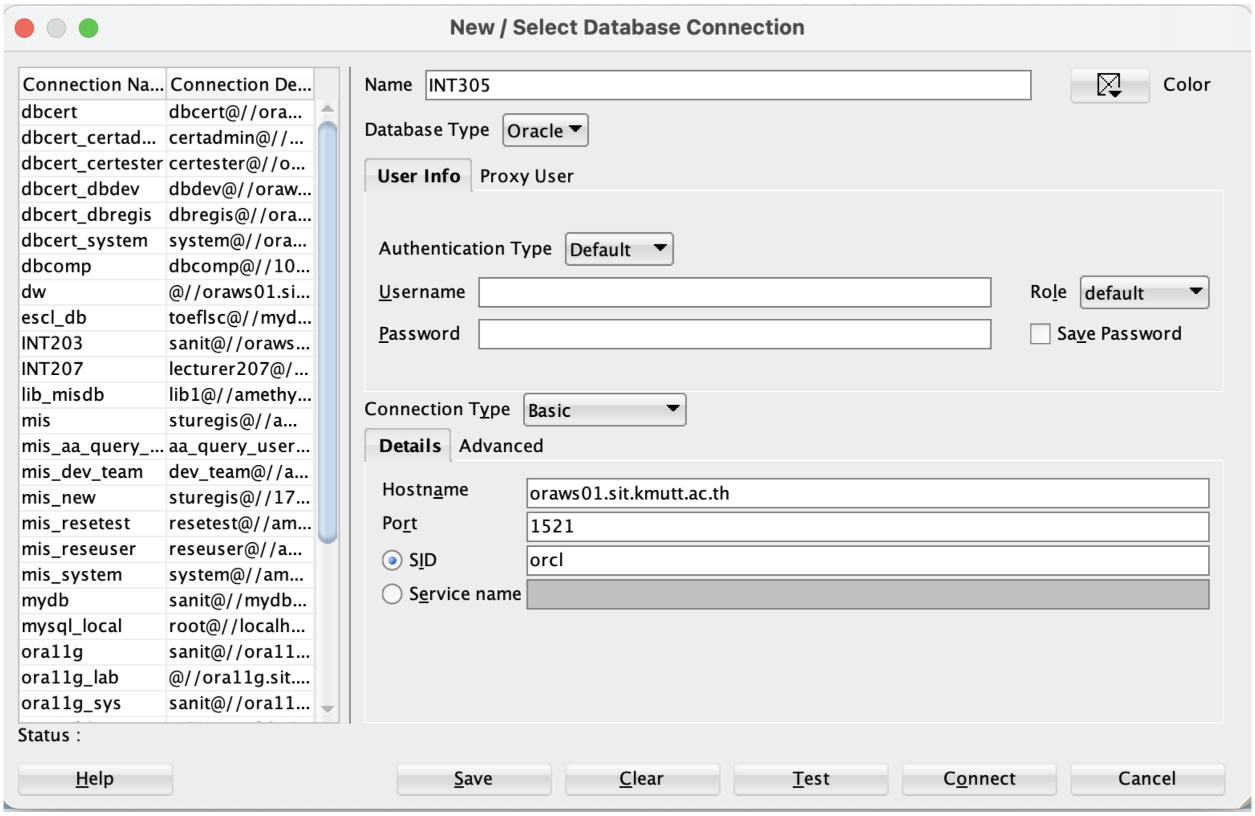
USERNAME : DBxxx (xxx 001-170)

PASSWORD : ORACLE

HOST: oraws01.sit.kmutt.ac.th

HOST (no vpn) : oraws01-4home.sit.kmutt.ac.th

SID : orcl



New / Select Database Connection

Connection Na...	Connection De...
dbcert	dbcert@//ora...
dbcert_certad...	certadmin@//...
dbcert_certester	certester@//o...
dbcert_dbdev	dbdev@//oraw...
dbcert_dbregis	dbregis@//ora...
dbcert_system	system@//ora...
dbcomp	dbcomp@//10...
dw	@//oraws01.si...
escl_db	toeflsc@//myd...
INT203	sanit@//oraws...
INT207	lecturer207@/...
lib_misdb	lib1@//amethy...
mis	sturegis@//a...
mis_aa_query_...	aa_query_user...
mis_dev_team	dev_team@//a...
mis_new	sturegis@//17...
mis_resetest	resetest@//am...
mis_reseuser	reseuser@//a...
mis_system	system@//am...
mydb	sanit@//mydb...
mysql_local	root@//localh...
ora11g	sanit@//ora11...
ora11g_lab	@//ora11g.sit....
ora11g_sys	sanit@//ora11...

Status :

Name: INT305

Database Type: Oracle

User Info | Proxy User

Authentication Type: Default

Username:

Role: default

Password:

☐ Save Password

Connection Type: Basic

Details | Advanced

Hostname: oraws01.sit.kmutt.ac.th

Port: 1521

● SID: orcl

○ Service name:

Download the script StaffXML.sql and run it.

```
<staffs>

   <staff staffno="SA9" branchno="B007">

       <name>

          <firstname>Mary</firstname>

          <lastname>Howe</lastname>

      </name>

      <position>Assistant</position>

      <sex>F</sex>

      <salary>9000</salary>

   </staff>

   <staff>

    …….

  </staff>

</staff>
```

XPath Expression

XPath uses a path expression to select nodes or list of node from a XML document.

Expression	Description
nodename	Selects all nodes with the name “nodename”
/	Selects from the root node
//	Selects nodes in the document from the current node that match the selection no matter where they are
@	Selects attributes
employee	Selects all node with the name “employee”
employees/employee	Selects all employee elements that are children of employees
//employee	Selects all employee elements no matter where they are in the document
/employees/employee[1]	Selects the first employee element that is the child of the employees element
/employees/employee[last()]	Selects the last employee element that is the child of the employees element
//employee[@type=’admin’]	Select all the employee elements that have an attribute named type with a value of ‘admin’

XMLQuery Syntax

```
        XMLQuery( '<xquery string>'

                        PASSING    <xmltype column>

                        RETURNING  CONTENT

                        NULL ON EMPTY  )
```

xquery string	XQuery expression including XPath and FLWOR expressions.
xmltype column	the column with XMLType where to get the XML for querying
RETURNING CONTENT	is mandatory (not support RETURNING SEQUENCE)
NULL ON EMPTY	is optional that tells Oracle to do if the XQuery expression returns to nothing

Example : Display the first name of the first employee

```
SELECT XMLCAST(  XMLQUERY('/staffs/staff[1]/name/firstname'

                          PASSING    data

                          RETURNING CONTENT  )  AS VARCHAR2(100))   T

FROM  staffs ;
```

Try to replace the xquery string with /staffs/staff[1]//firstname

XMLCAST syntax

```
        XMLCAST(  value_expresion    AS      datatype   )
```

XMLCAST casts value expression to the scalar SQL datatype (NUMBER, VARCHAR2)

XMLQuery with FLWOR expression

FLWOR is the most general expression syntax in XQuery. FLWOR (pronounced “flower”) stands for FOR, LET, WHERE, ORDER BY and RETURN.

Syntax of FLWOR expression:

```
        FOR       forVar IN  inExpression

        LET       letVar := letExpression

        [WHERE      filterExpression]

        [ORDER BY  orderSpec  [ascending | descending]

        RETURN   expression
```

Example : List the first name of all staffs who work in the branch no B003. The result should be ordered by first name.

```
SET LONG 1000 -- set column long to avoid truncating the column

SELECT  XMLQUERY('for  $i  in  /staffs/staff
                    where $i/@branchno="B003"
                    order by $i/name/firstname
                    return <row> {$i/name/firstname}</row>'
                    PASSING  data
                    RETURNING CONTENT)  T
FROM staffs ;
```

```
SELECT  XMLQUERY('for  $i  in  /staffs/staff[@branchno="B003"]
                    order by $i//firstname
                    return <row> {$i//firstname}</row>'
                    PASSING  data
                    RETURNING CONTENT)  T
FROM staffs ;
```

```
SELECT  XMLQUERY('for  $i  in  /staffs/staff[@branchno="B003"]
                    order by $i//firstname
                    return <Name>{$i//firstname/text()} {" "} {$i//lastname/text()}</Name>'
                    PASSING  data
                    RETURNING CONTENT)  T
FROM staffs ;
```

XMLTABLE Syntax

```
XMLTABLE  ( '<xquery string>'
                    PASSING   <xml column>
                    COLUMNS  <column name>  <column type>     PATH  <xquery path>,
                             <column name>  <column type>     PATH  <xquery path> [, …]  )
```

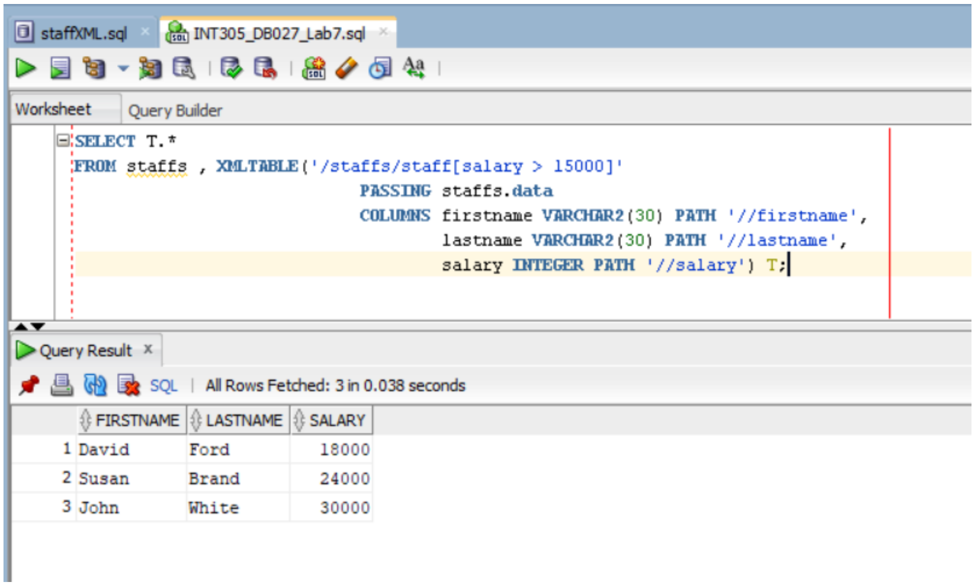
XMLTABLE function is used to produce a virtual SQL table containing data derived from XML values on which the function operates.

Example: List all staffs who are females. The result should include staff No, first name and last name and show data in virtual table.

```
SELECT T.*
FROM staffs , XMLTABLE('/staffs/staff[sex="F"]'
                          PASSING staffs.data
                          COLUMNS  staffno   VARCHAR2(6)  PATH '@staffno',
                                   firstname VARCHAR2(30) PATH '//firstname',
                                   lastname  VARCHAR2(30) PATH '//lastname')   T ;
```

Write an SQL/XML statement by using FLWOR expression for each following query.

1. Show staffs whose salary is greater than 15,000.

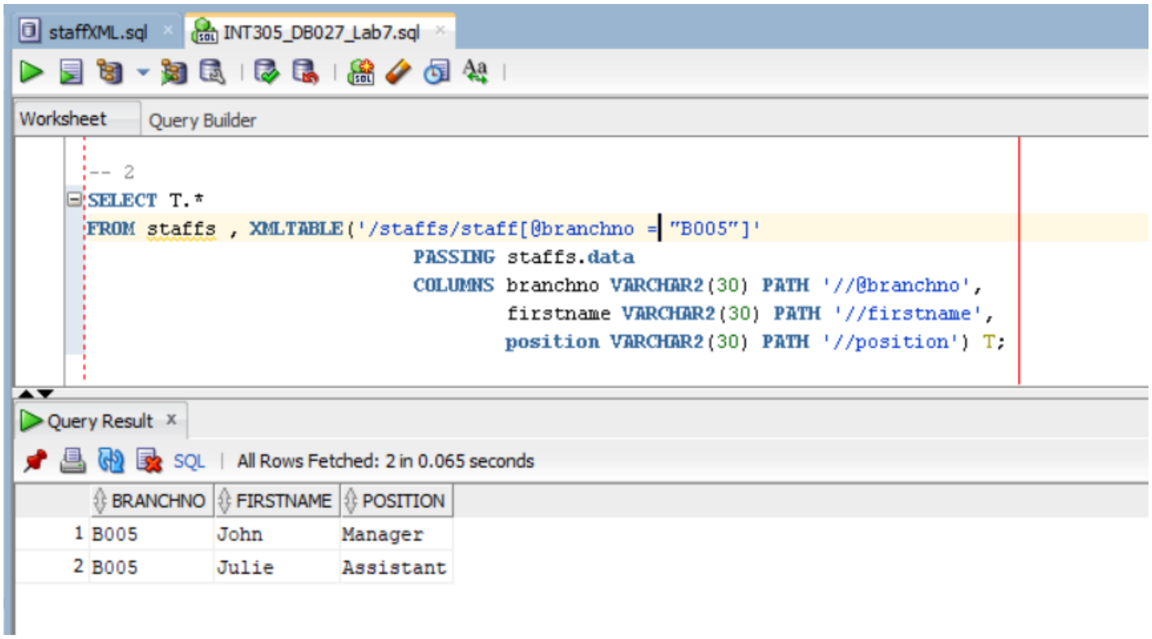


```
SELECT T.*
FROM staffs , XMLTABLE('/staffs/staff[salary > 15000]'
                        PASSING staffs.data
                        COLUMNS firstname VARCHAR2(30) PATH '//firstname',
                                lastname VARCHAR2(30) PATH '//lastname',
                                salary INTEGER PATH '//salary') T;
```

All Rows Fetched: 3 in 0.038 seconds

	FIRSTNAME	LASTNAME	SALARY
1	David	Ford	18000
2	Susan	Brand	24000
3	John	White	30000

2. Show the first name and position of staffs who works in the branch number B005.

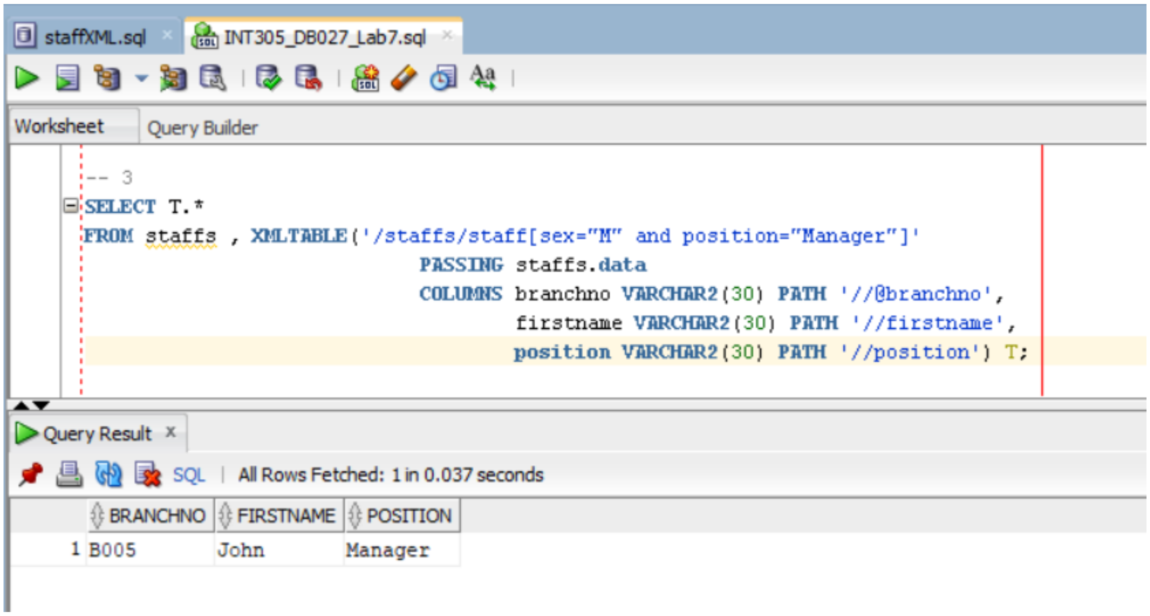


```
-- 2
SELECT T.*
FROM staffs , XMLTABLE('/staffs/staff[@branchno ="B005"]'
                            PASSING staffs.data
                            COLUMNS branchno VARCHAR2(30) PATH '//@branchno',
                                    firstname VARCHAR2(30) PATH '//firstname',
                                    position VARCHAR2(30) PATH '//position') T;
```

All Rows Fetched: 2 in 0.065 seconds

```
    BRANCHNO  FIRSTNAME  POSITION
  1 B005      John       Manager
  2 B005      Julie      Assistant
```

3. Show staffs who are males and whose position is manager.

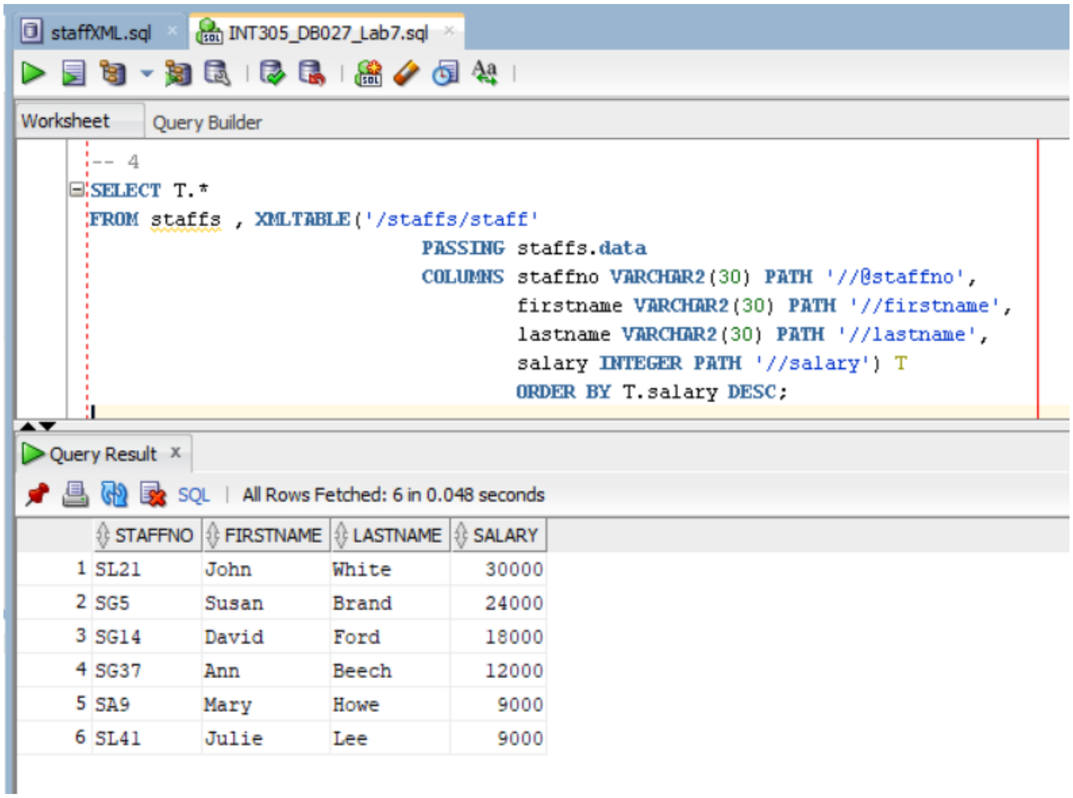


```
-- 3
SELECT T.*
FROM staffs , XMLTABLE('/staffs/staff[sex="M" and position="Manager"]'
                            PASSING staffs.data
                            COLUMNS branchno VARCHAR2(30) PATH '//@branchno',
                                    firstname VARCHAR2(30) PATH '//firstname',
                                    position VARCHAR2(30) PATH '//position') T;
```

All Rows Fetched: 1 in 0.037 seconds

```
    BRANCHNO  FIRSTNAME  POSITION
  1 B005      John       Manager
```

4. Show the staff number, first name, last name and salary of all staffs. Arrange the result in descending order by salary.



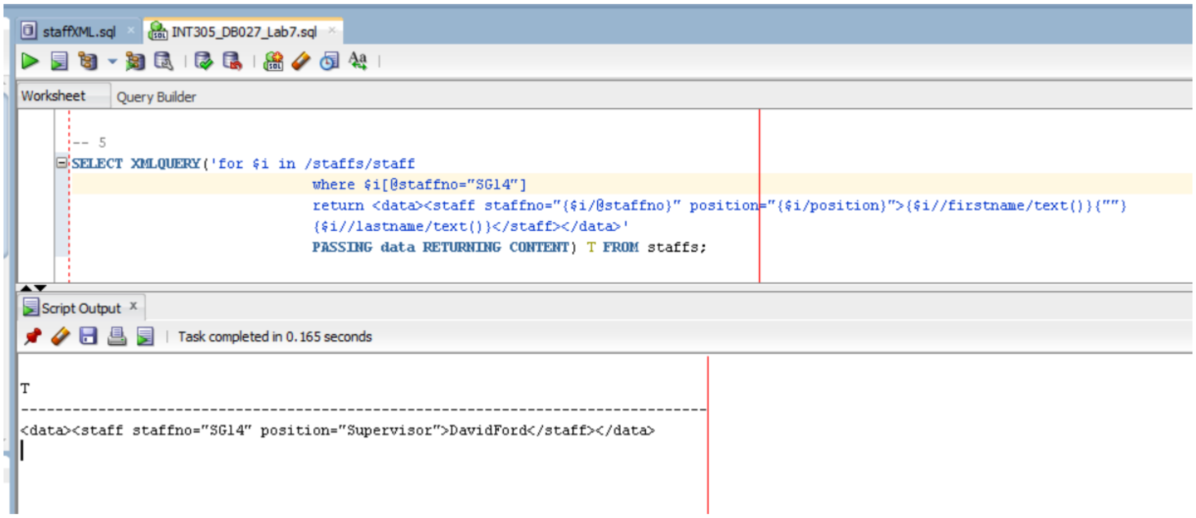
```
-- 4
SELECT T.*
FROM staffs , XMLTABLE('/staffs/staff'
                          PASSING staffs.data
                          COLUMNS staffno VARCHAR2(30) PATH '//@staffno',
                                  firstname VARCHAR2(30) PATH '//firstname',
                                  lastname VARCHAR2(30) PATH '//lastname',
                                  salary INTEGER PATH '//salary') T
                                  ORDER BY T.salary DESC;
```

All Rows Fetched: 6 in 0.048 seconds

	STAFFNO	FIRSTNAME	LASTNAME	SALARY
1	SL21	John	White	30000
2	SG5	Susan	Brand	24000
3	SG14	David	Ford	18000
4	SG37	Ann	Beech	12000
5	SA9	Mary	Howe	9000
6	SL41	Julie	Lee	9000

5. Show the staff whose staff number is SG14. The result should show as follows:

<data><staff staffno="SG14" position="Supervisor">David Ford</staff></data>



```
-- 5
SELECT XMLQUERY('for $i in /staffs/staff
                    where $i[@staffno="SG14"]
                    return <data><staff staffno="{$i/@staffno}" position="{$i/position}">{$i//firstname/text()}{""}
                    {$i//lastname/text()}</staff></data>'
                    PASSING data RETURNING CONTENT) T FROM staffs;
```

Task completed in 0.165 seconds

```
T
--------------------------------------------------------------------------------
<data><staff staffno="SG14" position="Supervisor">DavidFord</staff></data>
```