**TERM : FALL 2022**

**A picture containing logo

Description automatically generated**

| **Course & Section Code:** | **COMP214** |
| --- | --- |
| **Course Name:** | **Advanced Database Concept** |
|  |  |
| **Q&A / Virtual Office Hour:** | **Wed/Thur** |
| **Instructor Name & Email:** | Ersan Cam  [ecam@my.centennialcollege.ca](mailto:ecam@my.centennialcollege.ca) |

Assigment#1

**Due Date : October 13th Midnight**

Instructions for delivery of this Labs file back to instructor.

**Step 1:** Download this word copy of Lab document.

**Step2:** Work on your question in SQL Developer.

**Step3:** Once you solve the problem copy paste the code under each question and Highlight with RED color

**Step4:** Also go to your Sql Developer and capture screen entire screen with command you execute and result at the bottom. Use Snipping tool in windows to capture screen shot. Below picture shows how to open free windows based snipping screen capture tool

Step5: Drop your finalized & saved word document to respective Lab /Assignment dropbox assignment folder.

**Please note that Instructor keeps the right to call out any students randomly to ask demonstration of their solution, and walk thru their work and justify their answers in one on one breakout room.**

**The tool to capture screen shot. Snipping Tool or equivalent.**

Graphical user interface, text, application

Description automatically generated

Demo Screen shot: Always share Editable version of your code and Screen shot of your SQL developer with your instructor.. Do not share just code or not only Screen shot…

Copy paste screen shot of same command

Graphical user interface, text, application, email

Description automatically generated

**For first 4 questions please use Just Lee Book database**

**Question#1 :**

**Create a list the title of all books in the same category as books previously purchased by customer 1005. Don’t include books this customer has already purchased.**

SELECT B.ISBN, B.TITLE FROM BOOKS B WHERE B.CATEGORY IN

(SELECT B.CATEGORY FROM ORDERS O JOIN ORDERITEMS OI ON O.ORDER#=OI.ORDER# JOIN BOOKS B ON OI.ISBN = B.ISBN WHERE O.CUSTOMER#=1005)

AND B.ISBN NOT IN

(SELECT OI.ISBN FROM ORDERS O JOIN ORDERITEMS OI ON O.ORDER#=OI.ORDER# JOIN BOOKS B ON OI.ISBN = B.ISBN WHERE O.CUSTOMER#=1005);

Graphical user interface, text, application, email

Description automatically generated

**Question #2** .

**List the shipping city and state for the order that had the longest shipping delay.**

SELECT SHIPCITY, SHIPSTATE, SHIPDATE-ORDERDATE

AS SHIPPING\_DELAY

FROM ORDERS

WHERE SHIPDATE-ORDERDATE = (SELECT MAX(SHIPDATE-ORDERDATE) FROM ORDERS);

**Graphical user interface, text, application

Description automatically generated**

**Question #3**

Determine which customers placed orders for the least expensive book (in terms of regular retail price) carried by JustLee Books.

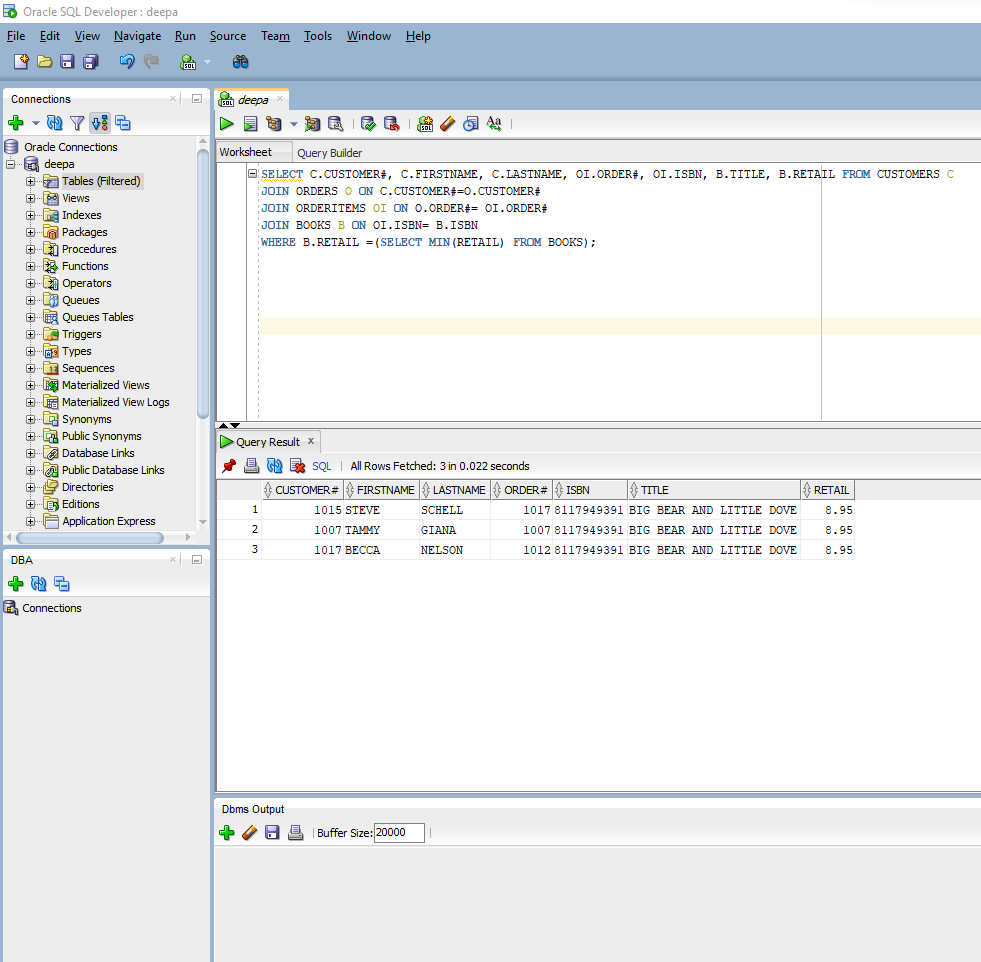
SELECT C.CUSTOMER#, C.FIRSTNAME, C.LASTNAME, OI.ORDER#, OI.ISBN, B.TITLE, B.RETAIL FROM CUSTOMERS C

JOIN ORDERS O ON C.CUSTOMER#=O.CUSTOMER#

JOIN ORDERITEMS OI ON O.ORDER#= OI.ORDER#

JOIN BOOKS B ON OI.ISBN= B.ISBN

WHERE B.RETAIL =(SELECT MIN(RETAIL) FROM BOOKS);



**Question#4 :**

Find out which other books were published by the same publisher of **BODYBUILD IN 10 MINUTES A DAY** book.

SELECT b.title, b.pubid FROM books b INNER JOIN publisher p ON b.pubid=p.pubid

WHERE B.pubid = ( SELECT PUBID FROM BOOKS WHERE TITLE IN 'BODYBUILD IN 10 MINUTES A DAY')

AND b.title !='BODYBUILD IN 10 MINUTES A DAY';

Graphical user interface, text, application, email

Description automatically generated

USE HR database for below questions ( for Question 5, 6 ,7)

**Question #5: Use HR Database for this question**

Find out employees’ ID who has never been changed his/her job since the beginning. (Find out whose record never gone to JOB\_HISTORY table and stayed only in EMPLOYEE Table)

Hint: Use MINUS set operator

SELECT EMPLOYEE\_ID

FROM HR\_EMPLOYEES

MINUS

SELECT EMPLOYEE\_ID

FROM HR\_JOB\_HISTORY;

Graphical user interface, text, application

Description automatically generated

**Question #6**

Create a report/query for the HR application to produce the address and city and country name details of all departments in the company.

Output should be similar to below.

Table

Description automatically generated

SELECT DISTINCT HL.LOCATION\_ID, HL.STREET\_ADDRESS, HL.CITY, HL.STATE\_PROVINCE, HC.COUNTRY\_NAME

FROM HR\_DEPARTMENTS HD

JOIN HR\_LOCATIONS HL

ON HD.LOCATION\_ID=HL.LOCATION\_ID

JOIN HR\_COUNTRIES HC ON HL.COUNTRY\_ID = HC.COUNTRY\_ID

WHERE HD.LOCATION\_ID=HL.LOCATION\_ID

ORDER BY LOCATION\_ID ASC;

Graphical user interface, text, application, email

Description automatically generated

**Question #7**

The HR department needs a report of employees in US country.

Display the last name of employee, salary, job, department\_number ,department name , city and country\_id details for those who are in country\_id as US and for those salary is in the range of 4000 and 10000

Sample output should be similar to below

Graphical user interface, text, application, table

Description automatically generated

SELECT E.EMPLOYEE\_ID, E.LAST\_NAME, E.SALARY, D.DEPARTMENT\_NAME, L.CITY, L.COUNTRY\_ID

FROM HR\_EMPLOYEES E

JOIN HR\_DEPARTMENTS D

ON E.DEPARTMENT\_ID = D.DEPARTMENT\_ID

JOIN HR\_LOCATIONS L

ON D.LOCATION\_ID=L.LOCATION\_ID

WHERE L.COUNTRY\_ID = 'US'

AND E.SALARY BETWEEN 4000 AND 10000

ORDER BY E.EMPLOYEE\_ID DESC;

Graphical user interface, text

Description automatically generated with medium confidence

Question #8

Explain Variable Scope and visibility and provide some examples of codes with Nested Block version

Variable scope and visibility refer to the access of a variable inside and outside a nested block, which is a block inside another block. The variable declared outside the block – global variable can be accessed outside and inside another block. However, when a variable is declared inside a block which is called a local variable, it can be accessed only inside the block.

This means the scope of a global variable is throughout the blocks of code while the scope of local variable is limited to the block of code in which it has been declared. Also, if the same variable (same name) is used inside and outside a block, the local variable takes precedence.

DECLARE

var\_global varchar2 (20) := 'I am global'; -- declared as a global variable--

var\_name varchar2(20) := 'James'; -- declared as a global variable--

BEGIN

DECLARE

var\_local varchar2 (20) := 'I am local' ;

var\_name varchar2 (20) := 'James Jr';

BEGIN

DBMS\_OUTPUT.PUT\_LINE(' This is the global variable: ' || var\_global); --Global variable is accessible inside the block--

DBMS\_OUTPUT.PUT\_LINE(' This is the local variable: ' || var\_local); --Local variable is accessible inside the block only--

DBMS\_OUTPUT.PUT\_LINE(' This is the name variable: ' || var\_name); --Local variable takes precedence and prints James Jr and not James--

END;

DBMS\_OUTPUT.PUT\_LINE(' This is the global variable: ' || var\_global); --global variable is accessible outside--

DBMS\_OUTPUT.PUT\_LINE(' This is the name variable: ' || var\_name); --Local variable is not accessible outside and only global variable James is printed--

END ;

Text

Description automatically generated with medium confidence

Graphical user interface, text, application, email

Description automatically generated

On trying to print local variable outside the block, it gives error.

Question #9 :

Explain the Scalar variables?

Scalar variables are those variables which can hold only a single value.

a)What type of data we can store them?

The data types that can be stored in Scalar variables are CHAR, VARCHAR2, NUMBER, BINARY\_INTEGER, PLS\_INTEGER, BOOLEAN, BINARY\_FLOAT,BINARY\_DOUBLE. Of these, char, varchar2, number and boolean are used the most.

b)Which other data types we can store in them (like CHAR is one of them how about others? ..)

Other data types which can be stored in scalar variables are DATE,TIMESTAMP,TIMESTAMP WITH TIME ZONE, TIMESTAMP WITH LOCAL TIME ZONE, INTERVAL YEAR TO MONTH, INTERVAL DAY TO SECOND

c)How and Why we use %TYPE… Provide some examples.

We use %TYPE to specify the data type during variable declaration in the DECLARE part of PL/SQL. The datatype can be declared directly by specifying the data type or can be done using %TYPE in which the dat type in specified indirectly. It can be used in two ways.

1. By database table and column name, using name of the declared variable

v\_lastname employees.last\_name%TYPE

Here a variable named v\_lastname is declared and then datatype is declared by using %TYPE- use the same datatype as last\_name in employee table. if it is varchar2 then the variable v\_lastname would be varchar2.

1. By database table and column name using another declared variable

v\_firstname varchar2(10)

v\_lastname v\_firstname %TYPE

Here, a variable v\_firstname is declared along with its data type and then variable v\_lastname is declared , its data type is specified using %TYPE which means use the same data type as v\_firstname.

d) How do you assign initial values to a variable ? give two options… If you do not have any initial value then what is going to be first value for those variable which you haven’t assign anything?

Initial values are assigned using 2 options . Those are 1. keyword DEFAULT and 2. by the symbol :=

for example-

v\_firstname varchar2(12) := ‘Deepa’ ;

v\_sal number(4,2) DEFAULT 2;

if an initial value has not been assigned, then it would have a null value.

Question 10 ) **Name that Shape**

Use IF THEN ELSE

Write a program that determines the name of a shape from its number of sides. Read

the number of sides from the user and then report the appropriate name as part of

a meaningful message. Your program should support shapes with anywhere from 3

up to (and including) 10 sides. If a number of sides outside of this range is entered

then your program should display an appropriate error message.

|  |  |
| --- | --- |
| **Triangle - 3 Sides** | **Square - 4 Sides** |
| Pentagon - 5 **Sides** | Hexagon - 6 **sides** |
| Heptagon - 7 **Sides** | Octagon - 8 **Sides** |

DECLARE

v\_sides NUMBER(8) := '&v\_sides' ;

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Please enter a number:' );

IF v\_sides = 3 THEN

DBMS\_OUTPUT.PUT\_LINE('You have entered three sides.The shape is a Triangle' );

ELSIF v\_sides = 4 THEN

DBMS\_OUTPUT.PUT\_LINE('You have entered four sides.The shape is a Square' );

ELSIF v\_sides = 5 THEN

DBMS\_OUTPUT.PUT\_LINE('You have entered five sides.The shape is a Pentagon' );

ELSIF v\_sides = 6 THEN

DBMS\_OUTPUT.PUT\_LINE('You have entered six sides.The shape is a Hexagon' );

ELSIF v\_sides = 7 THEN

DBMS\_OUTPUT.PUT\_LINE('You have entered seven sides.The shape is a Heptagon' );

ELSIF v\_sides = 8 THEN

DBMS\_OUTPUT.PUT\_LINE('You have entered eight sides.The shape is a Octagon' );

ELSIF v\_sides = 9 THEN

DBMS\_OUTPUT.PUT\_LINE('You have entered nine sides.The shape is a Nonagon' );

ELSIF v\_sides = 10 THEN

DBMS\_OUTPUT.PUT\_LINE('You have entered nine sides.The shape is a Decagon' );

ELSE

DBMS\_OUTPUT.PUT\_LINE(' The sides entered is out of range. Please enter a side between 3 to 10' );

END IF;

END;

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated