Homework #01

Create a PL/SQL Exercise: Anonymous Blocks, Data Types, and Variables

40 pts	A single text file containing the answers to **Part A and Part C** must be submitted via **Circuit**. Self and peer assessment must be submitted through the **links** provided, and you will need to **submit the confirmation code** of each survey in the self and peer assessment feedback on **Circuit**.
Notes:	Use grammatically correct sentences to convey your thoughts when answering questions. One-word answers, “naked” bulleted lists, etc. will receive a score of zero.

Overall Instruction

In this homework, your goal is to brainstorm a real-world problem-solving scenario where PL/SQL **basics** *(Anonymous Blocks, Data Types, and Variables)* are needed to provide a solution.

This homework consists of four parts:

- In Part A, you will first answer a few questions to demonstrate your understanding of PL/SQL Basics.
- In Part B, you will be guided through an example process to create a PL/SQL exercise.
- In Part C, you will need to come up with your own PL/SQL exercise.
- In Part D, which is done through the self and peer assessment stage on Circuit tool, you will evaluate the exercises created by your classmates and yourself.

Submission

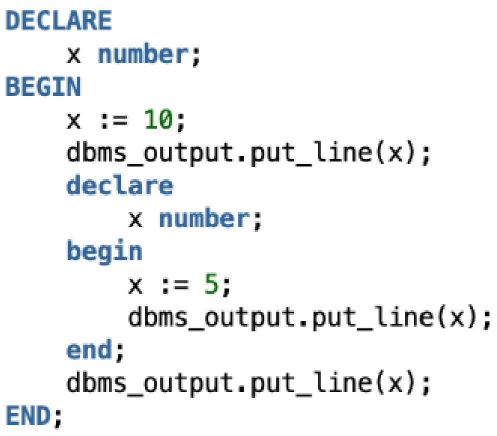
- Submit your answers to Part A and Part C in **a single text file** before the Content Submission deadline on Circuit. You can use the template provided to formulate your submission.
- Use the survey links provided in Part D to conduct self and peer assessment. Circuit will automatically assign submission for you to review in the Reviews Stage.

Grading

- Your grade for this homework is going to be effort-based. Your instructor and TA will read your submission, self and peer assessment, to determine whether you have created meaningful PL/SQL exercises for each other, and whether you have carefully read and evaluated the PL/SQL exercises created by others and yourself.

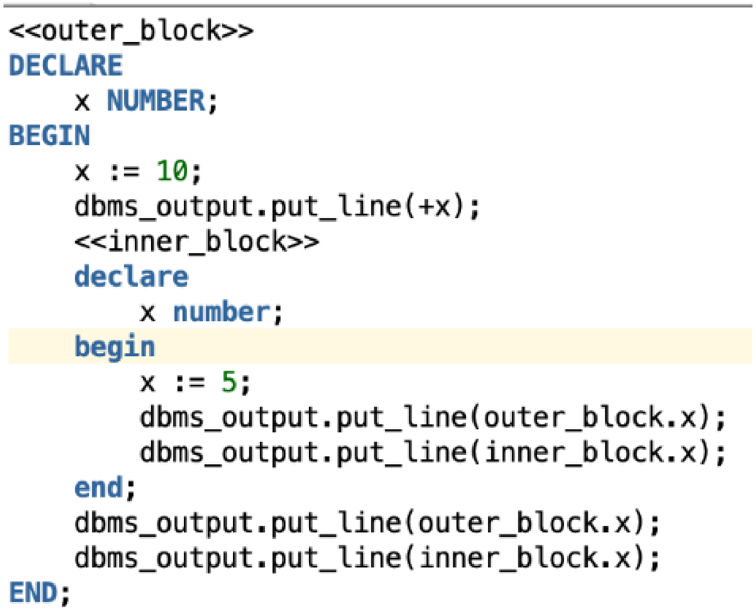
Part A (4pts): PL/SQL basics.

QA.1. What would be the output for the following code? (2pts)



```
DECLARE
    x number;
BEGIN
    x := 10;
    dbms_output.put_line(x);
    declare
        x number;
    begin
        x := 5;
        dbms_output.put_line(x);
    end;
    dbms_output.put_line(x);
END;
```

QA.2. What would be the output for the following code? (2pts)



```
<<outer_block>>
DECLARE
    x NUMBER;
BEGIN
    x := 10;
    dbms_output.put_line(+x);
    <<inner_block>>
    declare
        x number;
    begin
        x := 5;
        dbms_output.put_line(outer_block.x);
        dbms_output.put_line(inner_block.x);
    end;
    dbms_output.put_line(outer_block.x);
    dbms_output.put_line(inner_block.x);
END;
```

Part B: First Experience of Creating a PL/SQL Exercise.

This part demonstrates an example process of creating a PL/SQL exercise using the knowledge of control structures.

Problem Background:

Nina runs a food delivery business. One of her daily tasks is to keep track of the food packages ordered by different customers.

Overall Problem:

How many days did Nina receive her peak number of orders?

Table in Use for this Problem:

Name the table:

- FOOD_PACKAGE

The table contains the following attributes (columns):

- PackageID (integer)
- CustomerID (string)
- OrderDate (date)

Here are some example rows of the table:

PackageID	CustomerID	OrderDate
43	C-300004	27-JAN-11
43	C-300005	27-JAN-11
44	C-300001	27-JAN-11
44	C-300002	27-JAN-11
44	C-300003	27-JAN-11

Query for generate FOOD_PACKAGE table populated with dummy data (125 rows):

```
CREATE TABLE FOOD_PACKAGE (
    PACKAGEID INT,
    CUSTOMERID VARCHAR2(10),
    ORDERDATE DATE
);

INSERT INTO FOOD_PACKAGE
SELECT * FROM (
    (SELECT PACKAGENUMBER FROM PACKINGSLIP FETCH FIRST 5 ROWS
ONLY) CROSS JOIN
    (SELECT CUSTOMERID FROM CUSTOMER FETCH FIRST 5 ROWS ONLY)
CROSS JOIN
    (SELECT ORDERDATE FROM CUSTORDER FETCH FIRST 5 ROWS ONLY)
);
```

Step-by-step Solutions:

Step 1: List the order date and the number of orders placed in each day.

Answer Query:

```
SELECT OrderDate, COUNT(*) AS Number_of_Orders
FROM FOOD_PACKAGE
GROUP BY OrderDate;
```

Result:

```
ORDERDATE   NUMBER_OF_ORDERS
---------   ----------------
27-JAN-11   50
28-JAN-11   75
```

Step 2: Use the table returned in Step1, find the maximum number of orders placed in a day.

Answer Query:

```
SELECT MAX(Number_of_Orders)
FROM (SELECT OrderDate, COUNT(*) AS Number_of_Orders
FROM FOOD_PACKAGE
GROUP BY OrderDate);
```

Result:

```
MAX(NUMBER_OF_ORDERS)
---------------------
75
```

Step 3: List the order date(s) with the maximum number of orders placed. You can hard code the maximum number of orders using the value obtained from Step 2.

Answer Query:

```
SELECT OrderDate
FROM FOOD_PACKAGE
GROUP BY OrderDate
HAVING COUNT(*) = 75;
```

Result:

```
ORDERDATE
---------
28-JAN-11
```

Step 4: List the number of order dates returned in Step 3

Answer Query:

```
SELECT COUNT(*)
FROM (
     SELECT ORDERDATE
     FROM FOOD_PACKAGE
     GROUP BY ORDERDATE
     HAVING COUNT(*) = 75
);
```

Result:

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

Step 5: Translate the code in Step 1 to Step 4 into one anonymous PL/SQL block.

You should store the intermediate results into a variable. Print the answer to the overall question using dbms output.

Answer Query:

```
DECLARE

    MAX_ORDER INTEGER;

    NUMBER_DAYS INTEGER;

BEGIN

    SELECT MAX(COUNT(*)) INTO MAX_ORDER

    FROM FOOD_PACKAGE GROUP BY ORDERDATE;


    SELECT COUNT(*) INTO NUMBER_DAYS

    FROM (

    SELECT ORDERDATE

    FROM FOOD_PACKAGE GROUP BY ORDERDATE

    HAVING COUNT(*)= MAX_ORDER) T1;

     DBMS_OUTPUT.PUT_LINE('THERE IS/ARE ' ||
     TO_CHAR(NUMBER_DAYS) || ' DAY(S) WHEN NINA RECEIVED
     HER PEAK NUMBER OF ORDERS.');

END;
```

Part C (28 pts): Now It's Your Turn to Create a PL/SQL Exercise Related to Your Interest!

QC.1. Problem Background (4 pts): Please provide some context for your PL/SQL exercise with less than 100 words.
Hints: You can think of any areas you are interested in, or any real-world problems that can be solved by this process. Examples include details about video game developers and the day that particular movie theaters sell the most tickets. It can even be used in other fields like finance, education, etc.

QC.2. Overall Problem (4 pts): Please formulate the overall PL/SQL problem you try to solve. The overall problem should be complicated enough to require PL/SQL blocks. There should be **at least 3 steps** needed to solve this problem.

QC.3. Create a table with dummy data for solving the problem **(10 pts)**. The table should have at least 5 attributes, and at least 100 rows of data. Please use what you learned from Lab 2 to generate dummy data.
In your submission, include:

- **Table specification**: table name, attribute name and data types
- **An SQL query to create and populate your table with dummy data**: You can use the Eagle database as a source for generating dummy data. If you use other online resources, please include the reference and instructions for us to run your query.

QC.4. List the key steps to solve the overall problem (5 pts). Each step should have a standalone answer in SQL or PL/SQL. The problem should require **at least 3 steps** to solve.

QC.5. Please provide the correct answer for each of the steps you listed above. (5 pts) Please provide the SQL or PL/SQL code for each answer.

Part D (8 pts): Self and Peer assessment.

After the content submission deadline, you will conduct self and peer review of this homework.

All reviews need to be submitted through Google Form. The review for each submission (your self assessment, as well as each of the three peer assessments) should be submitted separately (4 submissions in total, each worth 2 points).

Links to the Google Form survey are provided in the rubrics on Circuit.