Homework #02

Create a PL/SQL Exercise: Functions and Procedures

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 **https://forms.gle/4Kiy8D13FoMRLPrU9 on Google Form. Peer assessment assignments will be managed** 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 **Functions and Procedures** 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 functions and procedures.
- 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 the Circuit tool, you will evaluate the exercises created by your classmates and yourself.

Submission

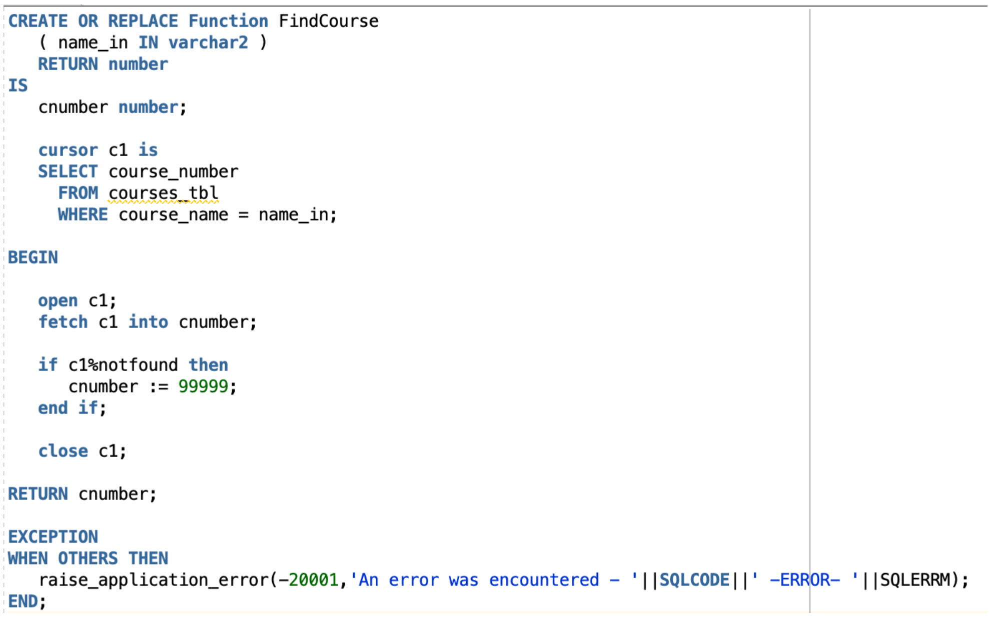
- Submit your answers to Part A and Part C in **a single text file** before the Content Submission deadline on Circuit. You can use the template provided to formulate your submission.
- Use **https://forms.gle/4Kiy8D13FoMRLPrU9** to conduct self and peer assessment on Google Form. The Circuit will automatically assign submissions 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.

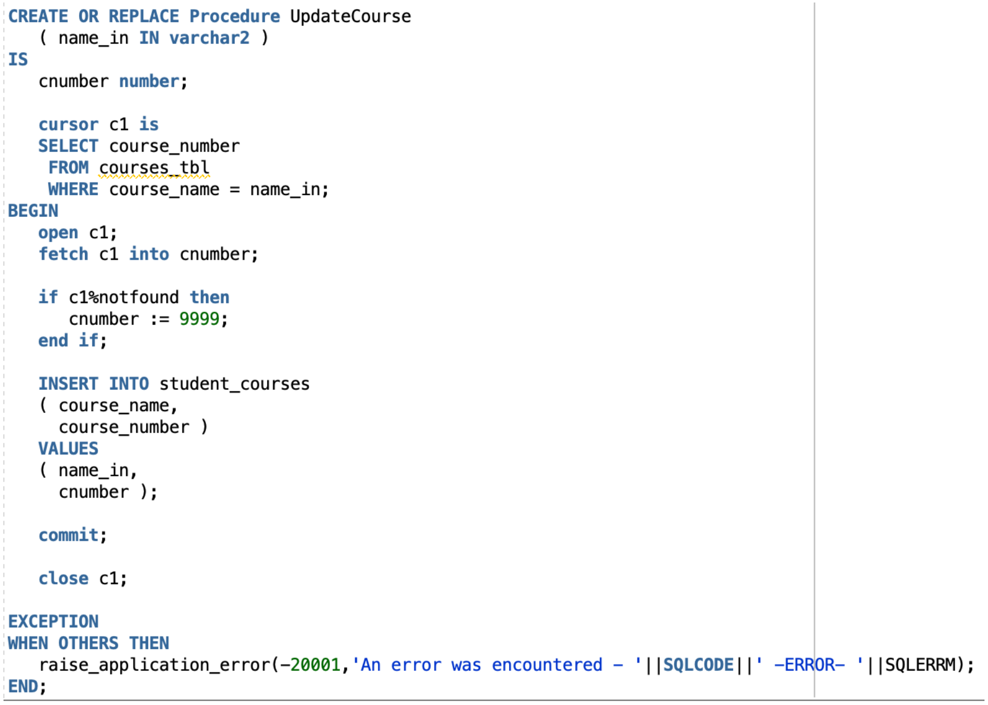
Part A (4pts): PL/SQL Functions and Procedures.

QA.1. What is the parameter of this function? What would be the output if a course number does not exist? (2pts)



```
CREATE OR REPLACE Function FindCourse
   ( name_in IN varchar2 )
   RETURN number
IS
   cnumber number;

   cursor c1 is
   SELECT course_number
     FROM courses_tbl
     WHERE course_name = name_in;

BEGIN

   open c1;
   fetch c1 into cnumber;

   if c1%notfound then
      cnumber := 99999;
   end if;

   close c1;

RETURN cnumber;

EXCEPTION
WHEN OTHERS THEN
   raise_application_error(-20001,'An error was encountered - '||SQLCODE||' -ERROR- '||SQLERRM);
END;
```

QA.2. What is the parameter of this procedure? Please briefly describe what would happen if one executes: UpdateCourse(372). (2pts)



```
CREATE OR REPLACE Procedure UpdateCourse
   ( name_in IN varchar2 )
IS
   cnumber number;

   cursor c1 is
   SELECT course_number
    FROM courses_tbl
    WHERE course_name = name_in;
BEGIN
   open c1;
   fetch c1 into cnumber;

   if c1%notfound then
      cnumber := 9999;
   end if;

   INSERT INTO student_courses
   ( course_name,
     course_number )
   VALUES
   ( name_in,
     cnumber );

   commit;

   close c1;

EXCEPTION
WHEN OTHERS THEN
   raise_application_error(-20001,'An error was encountered - '||SQLCODE||' -ERROR- '||SQLERRM);
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 functions and procedures.

Special thanks to Estevan Becerra, from CNIT372 in Spring 2022. This exercise is adapted from a worked-out example created by Estevan Becerra.

Problem Background:

Alex owns a video game store, and one of his daily tasks is to keep track of the information on different video games.

Overall Problem:

Write a procedure such that Alex can use the UPC codes to find out the publisher's ID of a game.

A table in Use for this Problem:

Name the table:

- GAME_SHOP

The table contains the following attributes (columns):

- Game_id (number)
- UPC (number)
- Publisher_id (number)
- Release_year (number)

Here are some example rows of the table:

Game_id	UPC	Publisher_id	Release_year
1	8564	4	2007
2	9852	4	2007
3	11063	7	2006
4	9065	15	2011

Create a table with dummy data:

Download the **LoadGameShop.txt** in the attachment on the **Circuit** to create the table and import the data. (**Hint**: Due to the limitation of file types on the circuit, remember to change the **.txt** to **.sql** before you use it to import the data into Oracle SQL.)

Step-by-step Solutions:

/* Step 1: Discuss if a function or procedure is more appropriate to solve this problem. */

Suggested answer: A function can be used to return the publisher's ID. One can also use a procedure to write to a variable that represents the publisher's ID.

/ Step 2: Create a function declaration that allows users to pass in a string value that represents the UPC code. */*

Suggested answer:

```
CREATE OR REPLACE FUNCTION GETPUBLISHER (P_UPC NUMBER)
RETURN NUMBER AS
     /* PRIVATE VARIABLES */
BEGIN
     /* EXECUTABLE COMMANDS */
END;
```

/ Step 3: Building on your answer to Step 2, declare a private variable to store the publisher’s ID. */*

Suggested answer:

```
CREATE OR REPLACE FUNCTION GETPUBLISHER (P_UPC NUMBER)
RETURN NUMBER AS
          v_output NUMBER;
BEGIN
     /* EXECUTABLE COMMANDS */
END;
```

/ Step 4: Write a select statement to find the publisher’s name whose UPC code is '2705'. */*

Suggested answer:

```
SELECT PUBLISHER_ID FROM GAME_SHOP WHERE UPC='2705';
```

/ Step 5: Complete the executable component of the function you created in Steps 1-3.*

*Rewrite the select statement in Step 4, to use the parameter you defined in Step 2 to dynamically retrieve the publisher’s ID and store the publisher’s ID in the private variable you created in Step 3. Then return the publisher’s ID at the end of the function. */*

Suggested answer:

```
CREATE OR REPLACE FUNCTION GETPUBLISHER (P_UPC NUMBER)
RETURN NUMBER AS
          v_output NUMBER;
BEGIN
     SELECT publisher_id
INTO v_output
FROM GAME_SHOP
WHERE UPC = p_upc;

Return v_output;
END;
```

/ Step 6: Test and execute the function.*/*

Suggested answer:

```
    BEGIN
    dbms_output.put_line('The publisher ID of this game is: '
    || GETPUBLISHER('2705'));
    END;
```

Result:

```
    The publisher ID of this game is: 13
```

/ Step 7: How would you write a procedure to solve this problem?*/*

Suggested answer:

```
    CREATE OR REPLACE procedure FillInPublisher (P_UPC NUMBER,
    P_publisherid out NUMBER) AS
    BEGIN
         SELECT publisher_id
    INTO P_publisherid
    FROM GAME_SHOP
    WHERE UPC = p_upc;
    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, healthcare, 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.
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 a self and peer review of this homework.

All reviews need to be submitted through Google Forms. 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).

Please note: you need to submit your evaluation to Google Forms to receive the self and peer assessment credits. Links to the Google Form survey are provided in the rubrics on Circuit.