Advanced Database - TP 2 - Part 2
PL/SQL

Exercice 1. Functions

1. Create a function that gets an employee's name from it's empno.

2. Test the query with

```
SELECT myFunction(7654) FROM DUAL;
```

Note: The DUAL table is a special one-row, one-column table present by default in Oracle and other database installations. In Oracle, the table has a single VARCHAR2(1) column called DUMMY that has a value of 'X'. It is suitable for use in selecting a pseudo column such as SYSDATE or USER. More info: https://www.google.com/search?q=dual+table+oracle

Exercice 2. Procedure & Display & Cursors

Create a procedure that displays the net salary of this employee and the average salary of employees doing the same job

Exercice 3. Procedure & Update

Create a procedure that updates the employee's salary:

      a. if his/her salary is greater or equal to the average one, wage increase of 10%

      b. else his/her new salary becomes the average

  Note: Displays initial and resulting salary.

Exercice 4. Procedure

Calculate a bonus for the employees of table EMP, as follows:

      • SALESMAN: bonus = commission * 2

- CLERK: bonus = wage increase of 15%

- MANAGER: bonus = wage increase of 18%

Insert the calculated bonuses in the table BONUS.

Exercice 5. SELECT for UPDATE

Change the commission for employees by putting them all to 0

Write the procedure UpdateCommission for modifying the commission of employees based on their salary

- Sal<=1000 COMM=800

- Sal<=2000 COMM=1200

- Sal>2000 COMM=1500

Indications:

use the instruction "SELECT ... FROM table FOR UPDATE"

This will allow you to put a lock on the rows you want to update

The following statement specifies the current tuple to modify with an UPDATE or DELETE WHERE CURRENT OF cursor

Exercice 6. Condition on EXIT loop

Write a procedure that displays the five employees with the highest salaries and the five employees with the lowest salaries. This procedure should use a cursor, a loop and a condition to exit the loop.

Exercice 7. Triggers

The database administrator (DBA) wants to store the user who modifies (insert, delete, or update) the EMP table and also the time of modification.

1. **Create The triggers:** Set one or many triggers that can take into account these requirements. Use the functions user() and sysdate() that can retrieve the name of the Oracle user and current date and time respectively.

2. **Function to analyze results:** Write a function AnalyzeActivity that accepts a Oracle user name and/or a date and calculate the number of operations performed by the user, whether for all the users during the specific day, or for the specific user on the specific day, or for a specific user since the table creation.