Assignment:

PLSQL (Part-2)

Overview:

This assignment has ten exercises, each for around 15-30 minutes. You are expected to complete the course/ reference reading before attempting the exercise. On completion of the exercises, you will be able to achieve the following objectives:

S.No	Exercise	Description	Learning Objective
1	Using Procedures	This exercise is about creating a procedure to call a function and update database.	Create procedure suitable to the need
2	Using Packages	This exercise is about creating a package and creating functions inside it.	Create packages and functions
3	Using Triggers	This exercise is about creating trigger for preventing certain operation based on system time.	Create trigger based on timings
4	Using Triggers	This exercise is about creating trigger for preventing certain operation based on certain data.	Create trigger based on certain criteria
5	Using Procedures and Triggers	This exercise is about using a procedure inside a trigger.	Use procedure and triggers
6	Using Triggers	This exercise is about updating data on tables based on operations of a table.	Use triggers effectively
7	Using Procedures and Triggers	This exercise is about creating a procedure for updating a data and calling it inside trigger.	Call procedure in trigger
8	Using Triggers	This exercise is about creating trigger for certain data.	Create trigger effectively

9	Using Anonymous Block	This exercise is about writing anonymous block of code and execute it	Use anonymous block
10	Using Triggers	This exercise is about cascade deleting using triggers.	Use trigger effectively

PLSQL (Part 2)

Exercise 1: Using procedure

Create a procedure, NEW_EMP to insert a new employee into the EMPLOYEES table. The procedure should contain a call to the VALID_DEPTID function to check whether the department ID specified for the new employee exists in the DEPARTMENTS table.

Recommended duration: *20 minutes*

Exercise 2: Using Packages and functions

- Create a package called OVER_LOAD_Pack. Create two functions in this package, name each function PRINT_IT. The function accepts a date or a character string and prints a date or a number, depending on how the function is invoked.
- To print the date value, use DD-MON-YY as the input format, and FmMonth, dd, yyyy as the output format. Make sure to handle invalid input.
- To print out the number, use 999,999.00 as the input format.

Recommended duration: 25 *minutes*

Exercise 3: Using trigger

Ensure no changes can be made to EMPLOYEES table before 6am and after 10pm in a day.

Recommended duration: *10 minutes*

Exercise 4: Using trigger for certain events

Create a Trigger to ensure the salary of the employee is not decreased.

Recommended duration: *10 minutes*

Exercise 5: Using Procedures and Triggers

- Modifications to data are allowed on tables only during normal office hours of 9:00 a.m. until 6:00 p.m., i.e. Monday through Friday.
- Create a stored procedure called RESTRICT_DML that prevents the DML statements from executing outside of normal office hours, returning an alert message, “You may make changes only during normal business hours.” Then create a statement trigger on the NEW_EMP table that calls the above procedure.

Recommended duration: *20 minutes*

Exercise 6: Using triggers

Employee should receive an automatic increase in salary if the minimum salary for a job is increased. Implement this requirement through a trigger on the JOBS table.

Recommended duration: *15minutes*

Exercise 7: Using procedure and trigger

Create a stored procedure named UPD_EMP_SAL to update the salary amount. This procedure accepts two parameters: the job ID for which salary has to be updated and the new minimum salary for this job ID. This procedure is executed from the trigger on the JOBS table.

Recommended duration: *25 minutes*

Exercise 8: Using trigger

Create a row trigger named UPDATE_EMP_SALARY on the JOBS table that invokes the procedure UPD_EMP_SAL when the minimum salary in the JOBS table is updated for a specified job ID .

Recommended duration: *15 minutes*

Exercise 9: Using Anonymous Block

Query the EMPS table to see the current salary for employees who are programmers. Increase the minimum salary for the programmer job from 20,000 to 25,000.

Recommended duration: *20 minutes*

Exercise 10: Using trigger

Create a trigger for the DEPT table so that whatever a row is deleted from it, the corresponding rows of EMPS table are also deleted.

Recommended duration: *15 minutes*