# Lab 5 - Stored Procedures/Conditional Statements

## Submission

***Your submission will be a single text-based SQL file with appropriate header and commenting. Please ensure your file runs when the entire file is executed in SQL Developer.***

Create a new Worksheet in SQL Developer. Save the file as L05\_ID#\_LASTNAME.sql

## Your submission needs to be commented and include the question, the solutions.

In this Lab, you create PL/SQL stored procedures to perform the following tasks. As you know, a stored procedure does not return any value. To send values back to the caller, you can use OUT parameters.

A parameter can be

* IN parameter
* OUT parameter
* IN OUT parameter

See the following template:

|  |
| --- |
| **CREATE** **OR REPLACE** *procedure\_name*(**arg1** **IN**/**OUT/IN OUT** data\_type, ...) AS  **BEGIN**  ....  **EXCEPTION**  **WHEN OTHERS**  **THEN**  DBMS\_OUTPUT.PUT\_LINE (Error!');  **END** procedure\_name; |

For all the stored procedures make sure you handle all exceptions such as

* TOO\_MANY\_ROWS
* NO\_DATA\_FOUND
* OTHERS
* . . .

Besides checking all required exceptions, have the OTHER exception checked just in case any error occurs that has not been anticipated at the time you write the code.

## Tasks

1. Write a store procedure that gets an integer number and prints

*The number is even.*

If a number is divisible by 2.

Otherwise, it prints

*The number is odd.*

1. Create a stored procedure named *find\_employee*. This procedure gets an employee number and prints the following employee information:

First name

Last name

Email

Phone

Hire date

Job title

The procedure gets a value as the employee ID of type NUMBER.

See the following example for employee ID 107:

First name: Summer

Last name: Payn

Email: summer.payne@example.com

Phone: 515.123.8181

Hire date: 07-JUN-16

Job title: Public Accountant

The procedure display a proper error message if any error accours.

1. In a unionized company, the collective bargaining agreement often contains a requirement for staff to receive a given percentage salary increase an on annual basis. These salary increases are different for different departments. For example, the company must give all employees who work in the marketing department an annual 2.5% raise.

Write a procedure named *update\_salary\_by\_dept* to update the salaries of all employees in a given department and the given percentage increase to be added to the current salary if the salary is greater than 0. The procedure shows the number of updated rows if the update is successful.

The procedure gets two input parameters and sends one back out again with the number of rows updated.   
- When defining the first parameter, set the data type to the same data type as the associated field using the %type attribute.  
- Make sure your solution handles any possible errors with appropriate responses.

Additionally, write the code to execute the procedure.

1. In an attempt to equalize salaries over time, every year, the company increase the monthly salary of all employees who make less than the average salary by 1% (salary \*1.01).   
     
   Write a stored procedure named spUpdateSalary\_UnderAvg.   
   This procedure will not have any parameters. You need to find the average salary of all employees and store it into a variable of the same type.   
   - If the average salary is less than or equal to $9000, update the employees’ salary by 2% if the salary of the employee is less than the calculated average.   
   - If the average salary is greater than $9000, update employees’ salary by 1% if the salary of the employee is less than the calculated average.   
     
   The query displays an error message if any error occurs. Otherwise, it displays the number of updated rows.
2. The company needs a report that shows three categories of employees based their salaries. The company needs to know if the salary is low, fair, or high. Let’s assume that

* If the salary is less than
  + (avg salary – min salary) / 2

The salary is low.

* If the salary is greater than
  + (max salary – avg salary) / 2

The salary is high.

* If the salary is between
  + (avg salary - min salary) / 2
  + and
  + (max salary - avg salary) / 2
  + the end values included

The salary is fair.

Write a procedure named *spSalaryReport* to show the number of employees in each price category:

The following is a sample output of the procedure if no error occurs:

Low: 4

Fair: 12

High: 6

The values in the above examples are just random values and may not match the real numbers in your result.

Make sure you choose a proper data type for each variable. You may need to define more variables based on your solution.