**COMP214**

**Assignment 4**

**INSTRUCTOR NAME: ERSAN CAM**

**MARKS ALLOTTED: 100 (%70 back end + %20 Front End GUI + %10 Presentation)**

**WEIGHTING: 15%**

**General Instructions :**

You can use any Web based front end programming tool which can connect to our database in Lab or your own database

* Clean Coding, Interaction with Database (Connectivity, Login page etc ) , GUI features (Drop down menu , list box, Radio boxes etc if needed) and actual Presentation skills will be evaluated during presentation time.
* Expectation is that everyone in the team shares responsibility. If one person writes some stored procedure and the other one writes function , one or two can work on GUI same as in real life environment.
* **Presentation will be on last day.**
* Make power point presentation and do share with instructor in advance. Instructor will open presentation for you to present in front of class or you can plug your own laptop.
* If you want to use your own laptop , it is your responsibility to test connectivity of Projector otherwise do send all files and presentation to instructor and do DEMO from Instructor’s computer.

**Marking guidelines**

**Back-end programming (Up to 70 points)**

All the procedure ,function, triggers must be cleanly compiled with no errors. Tested on SQL Developer

You will be collecting points by doing successful Demo and test in front of Instructor and audience

**2-Front end GUI effectiveness (0 up to30 pnts)**

How user friendly is your GUI? Is it just basic & quickly written-basic but functional GUI or efficient and colorful/fancy GUI ?

Presentation skills (Eye contact with audience - Voice control - body gesture etc (Do some research on basic presentation skills )

**Please Note that all the table names here (EMLOYEES , JOBS, DEPARTMENTS) must refer to HR\_EMPLOYEES, HR\_JOBS, HR\_DEPARTMENTS naming convention as your Ryerson database has this HR\_ prefix**

**Employee Hiring / Terminating / Update Employee Info Screen**

**~~Task1-1~~**

Create a Procedure called Employee\_hire\_sp which accepts 9 parameters (p\_first\_name, p\_last\_name, p\_email, p\_salary, p\_hire\_date, p\_phone, p\_job\_id , p\_manager\_id, p\_department\_id )

|  |
| --- |
| Task 1-1 Code |
| create or replace PROCEDURE Employee\_hire\_sp (  p\_first\_name IN VARCHAR2,  p\_last\_name IN VARCHAR2,  p\_email     IN VARCHAR2,  p\_salary    IN NUMBER,  p\_hire\_date IN DATE,  p\_phone     IN VARCHAR2,  p\_job\_id    IN VARCHAR2,  p\_manager\_id IN NUMBER,  p\_department\_id IN NUMBER  ) AS  BEGIN  INSERT INTO hr\_employees (       EMPLOYEE\_ID,       FIRST\_NAME,       LAST\_NAME,       EMAIL,       PHONE\_NUMBER,       HIRE\_DATE,       JOB\_ID,       SALARY,       MANAGER\_ID,       DEPARTMENT\_ID  ) VALUES (          EMPLOYEES\_SEQ.NEXTVAL,       p\_first\_name,       p\_last\_name,       p\_email,       p\_phone,       p\_hire\_date,       p\_job\_id,       p\_salary,       p\_manager\_id,          p\_department\_id  );    COMMIT;  END Employee\_hire\_sp; |

**Task1-2 - GUI work**

Create a Page (Form) for Employee Hiring activities UNDER Employee Main Menu

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Task 1-3**

Under **Employee Main Menu** , create a link to list and Update all or any one the employee record.

Make a form to list all employees like excel grid format on the page and then let end user edit SALARY or PHONE or EMAIL information not any other info just any of these 3.

If any one of this info or all of these 3 info got change by end user then accept the new value and send to database as UPDATE COMMAND and replace old value with new one in employee table.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Task 2-1** **Identify JOB description**

Under the JOBS MAIN menu create a link called **Identify JOB description**

In this link or (page or form) idea is End user will enter JOB \_ID let’s say SA\_REP and form will bring accurate **JOB\_Description** from **Jobs** table

There will be one Edit box to accept JOB\_ID from end user

To do that, you can either use FUNCTION or Front end SQL Select

|  |
| --- |
| Task 2-1 Code |
| create or replace FUNCTION get\_job (  p\_job\_id IN VARCHAR2  ) RETURN VARCHAR2 IS  v\_job\_title VARCHAR2(50);  BEGIN  SELECT JOB\_TITLE INTO v\_job\_title  FROM HR\_JOBS  WHERE JOB\_ID = p\_job\_id;  RETURN v\_job\_title;    END get\_job; |

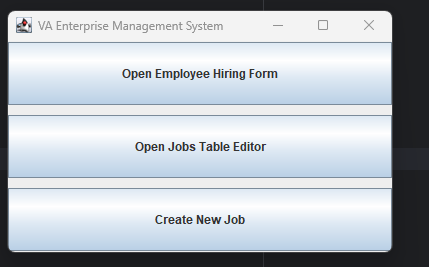
**Task 2-2 - Allow end user to change JOB\_DESCRIPTIPON**

|  |
| --- |
| Task 2-2 Code |
| CREATE OR REPLACE PROCEDURE UpdateJobDetails (  p\_job\_id IN VARCHAR2,  p\_job\_title IN VARCHAR2,  p\_min\_salary IN NUMBER,  p\_max\_salary IN NUMBER  )  AS  BEGIN  UPDATE hr\_jobs  SET job\_title = p\_job\_title,  min\_salary = p\_min\_salary,  max\_salary = p\_max\_salary  WHERE job\_id = p\_job\_id;  COMMIT;  END; |

**Task 2-3 Create a New JOB**

|  |
| --- |
| Task 2-3 Code |
| create or replace PROCEDURE new\_job(  p\_jobid IN hr\_jobs.job\_id%TYPE,  p\_title IN hr\_jobs.job\_title%TYPE,  v\_minsal IN hr\_jobs.min\_salary%TYPE)  IS  v\_maxsal hr\_jobs.max\_salary%TYPE := 2 \* v\_minsal;  BEGIN  INSERT INTO hr\_jobs(job\_id,job\_title, min\_salary, max\_salary)  VALUES (p\_jobid, p\_title, v\_minsal, v\_maxsal);  DBMS\_OUTPUT.PUT\_LINE('New row added to Jobs table');  DBMS\_OUTPUT.PUT\_LINE(p\_jobid || '  ' || p\_title || ' '|| v\_minsal || ' ' || v\_maxsal);  END new\_job;  -----------------------------------  CREATE OR REPLACE PROCEDURE new\_job(  p\_job\_id IN VARCHAR2,  p\_job\_title IN VARCHAR2,  p\_min\_salary IN NUMBER,  p\_max\_salary IN NUMBER  )  AS  BEGIN  INSERT INTO hr\_jobs (job\_id, job\_title, min\_salary, max\_salary)  VALUES (p\_job\_id, p\_job\_title, p\_min\_salary, p\_max\_salary);    COMMIT;  EXCEPTION  WHEN OTHERS THEN  ROLLBACK;  RAISE;  END; |

**Task 2-3 GUI work**



A screenshot of a computer

Description automatically generated

A screenshot of a computer screen

Description automatically generated

**Task 3 – Creating a Trigger and also a Store Procedure to verify and check any Job’s minimum and Maximum Salary range if they are in acceptable limit**

|  |
| --- |
| Task 3 Code – Part 1 |
| CREATE OR REPLACE PROCEDURE check\_salary (p\_the\_job IN VARCHAR2, p\_the\_salary NUMBER)  IS  v\_minsal  hr\_jobs.min\_salary%TYPE;  v\_maxsal hr\_jobs.max\_salary%TYPE;  BEGIN  SELECT min\_salary, max\_salary INTO v\_minsal, v\_maxsal  FROM hr\_jobs  WHERE job\_id = UPPER(p\_the\_job);  IF p\_the\_salary NOT BETWEEN v\_minsal AND v\_maxsal  THEN  RAISE\_APPLICATION\_ERROR(-20100, 'Invalid salary $' || p\_the\_salary ||'. '||'Salaries for the job'||p\_the\_job|| 'must be between $'||   v\_minsal ||' and $' || v\_maxsal);   END IF;   END; |

|  |
| --- |
| Task 3 Code Part-2 |
| CREATE OR REPLACE TRIGGER check\_salary\_trg  BEFORE INSERT OR UPDATE OF job\_id, salary ON hr\_employees  FOR EACH ROW  BEGIN  check\_salary(:new.job\_id, :new.salary);  END; |

**Test Cases:**

**Test Case 1:** Test whether this trigger and Stored procedure control mechanism works or not

|  |
| --- |
| Test Case 1 Code |
| create or replace TRIGGER check\_salary\_trg  BEFORE INSERT OR UPDATE OF salary ON hr\_employees  FOR EACH ROW  BEGIN  -- Check if the salary is valid by calling the check\_salary trigger  IF INSERTING OR UPDATING THEN  check\_salary(:NEW.job\_id, :NEW.salary);  ELSIF DELETING THEN  -- Optionally, handle deletion logic here if required  NULL; -- No specific action for DELETE, can be omitted  END IF;  END; |

Explain what happened and show this in your GUI and /or SQL developer  
  
A pop-up occurred stating that it was invalid, and salary should fall within the respective range in GUI. In SQL an error message appeared. Please see the image below for reference.

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Test case 2:**

UPDATE the salary of Employee ID 115 to $2000

You can use main Employee Menu Employee INFO update feature to do these test cases. If you can not then use SQAL developer to run simple UPDATE command.

Explain what happened and why? show us warning message on GUI and /or also in SQL developer

In SQL I was unable to update Employee 115 Karen C. Salary. I got the following Error/Warning Message:

A screenshot of a computer

Description automatically generated

However. In the GUI pop up platform, I was able to successfully change the employee salary manually. Below you will find the screenshot reference.

A yellow line on a white background

Description automatically generated

A screenshot of a computer

Description automatically generated

**Test case 3:**

CHANGE (UPDATE ) employee\_ID 115’s JOB\_ID to HR\_

Explain what happened and why? show us warning message on GUI and /or also in SQL developer

I was unable to update the employee 115 Karen C job id in SQL or in the GUI. I got the following error message. Screenshot reference below.

**THE END**