**TERM– WINTER 2023 **

| **Course & Section Code:** | **DBS501S1A** |
| --- | --- |
| **Course Name:** | **Procedure Using Oracles PL/SQL** |
| **Course Start & End Dates:** |  |
| **Q&A / Virtual Office Hour:** | **6:30 PM – 9:30 PM** |
| **Instructor Name & Email:** | **Ersan Cam: ersan.cam@senecacollege.ca** |

Lab#8 (Stored Procedures)

**1 -2 ) Stored Procedure which does two operations (Insert and Update)**

*CREATE TABLE ADDRESS\_POSTAL*

*(*

*Postal\_code varchar2(6) PRIMARY KEY,*

*City varchar2(40),*

*Province varchar2(3)*

*)*

|  |
| --- |
|  |

**Sample insert:**

M5G 2G4 London ON

M5A 2J2 Toronto ON

Please note that if you copy paste from Word document ‘ (Single quote) may cause problem. Try to copy in Notepad first to make sure single quote is good.

INSERT INTO *ADDRESS\_POSTAL* VALUES (‘M5G 2G4’ , ‘London’ ,’ON’)

INSERT INTO *ADDRESS\_POSTAL* VALUES (‘M5A 2J2’ , ‘Toronto’, ‘ON’)

Commit;

|  |
| --- |
|  |

==============

Write a procedure to have TWO main section in same procedure..

It will take input 4 parameters as **postal code, city, province and operation code** and inserts the values into the ZIPCODE table or update record.

Here are the operations that you need to perform

If operation is **‘I’** then it means INSERT , If operation is **‘U’** then it means UPDATE

**If it is ‘I’ then**

There should be a check to see if the postal code s already in the table.

If it is, an exception will be raised and an error message will be displayed stating “ this postal code is already exists in the system”

Otherwise you just insert the record.

Write an anonymous block that uses the procedure and inserts your zip code

**If the operation code is ‘U’ then** it does Update the city name for that particular Postal Code.

For instance if the parameters are ‘M5A 2J2’ ‘Milton’ ‘ON’ , ‘U’

Change the city info Toronto to Milton for Postal Code : ‘M5A 2J2’

Test case for Insert operations:

**Test Case #1:** Call Procedure with ‘M5G 2G4’ , ‘Niagrara’ , ‘ON’ and get message on screen shot

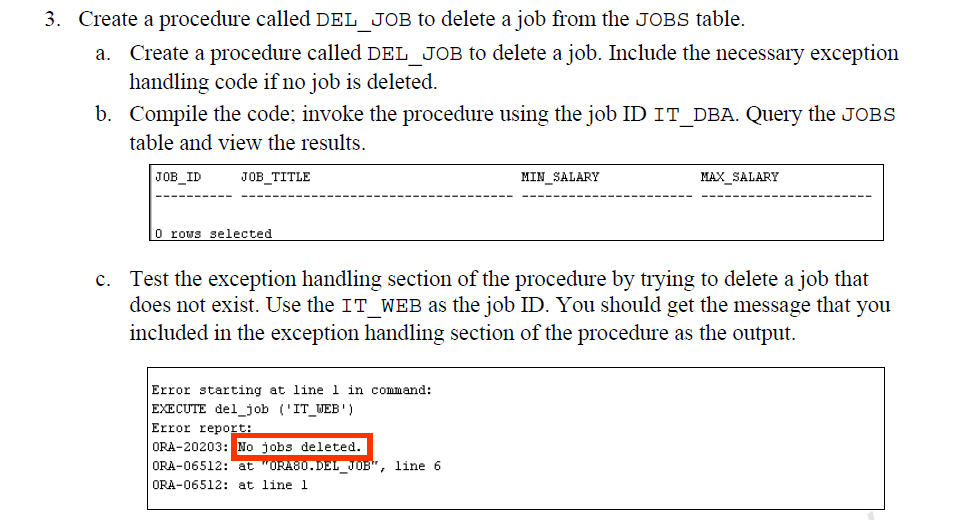
**Test Case #2 :** Call Procedure with M5J 3A1 a postal code for Scarborough ON

Test Case for Update operations

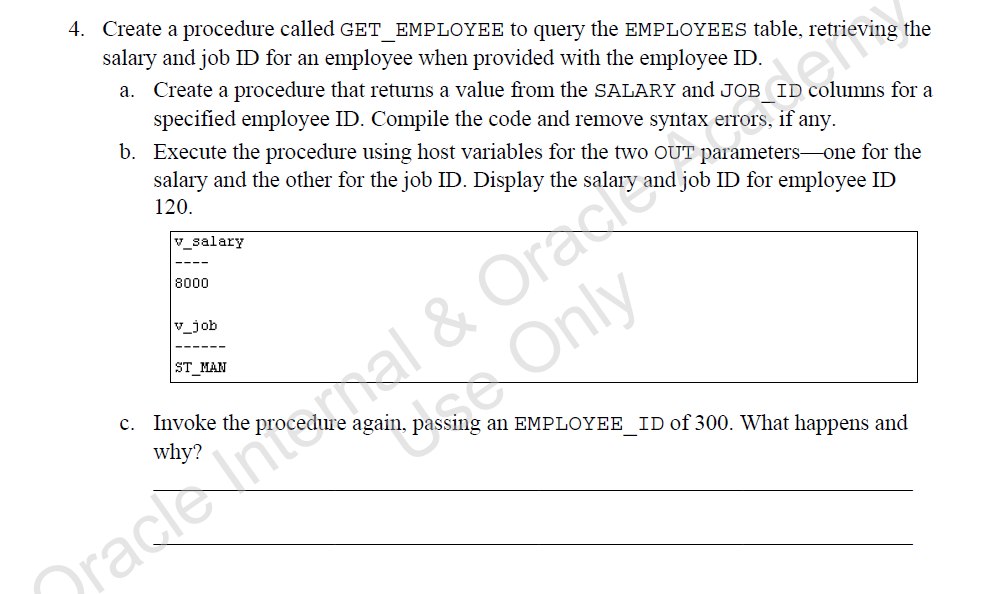
Test Case #3: parameters are ‘M5A 2J2’ ‘Milton’ ‘ON’ , ‘U’

update postal code of ‘M5A 2J2’ city to Milton

|  |
| --- |
| create or replace procedure insert\_update\_address  (v\_postal\_code address\_postal.postal\_code%type,  v\_city address\_postal.city%type,  v\_province address\_postal.province%type,  v\_operation varchar)  as  v\_count number;  begin  if (v\_operation = 'I' or v\_operation = 'i') then  select count(\*) into v\_count from address\_postal where postal\_code = v\_postal\_code;  if (v\_counnt = 1) then  raise\_application\_error(-20001, 'This postal code already exists in the system');  end if;  insert into address\_postal(postal\_code, city, province) values (v\_postal\_code, v\_city, v\_province);  dbms\_output.put\_line('New address has been added');  elsif (v\_operation = 'U' or v\_operation = 'u') then  update address\_postal  set city = v\_city  where postal\_code = v\_postal\_code;  if (sql%rowcount = 0) then  raise\_application\_error(-20002, 'No address found');  else  dbms\_output.put\_line('Address has been updated');  end if;  else  dbms\_output.put\_line('Invalid operation');  end if;  commit;  end; |



|  |
| --- |
| create or replace procedure del\_job  (v\_job\_id hr\_jobs.job\_id%type)  is  v\_count number;  begin  select count(\*) into v\_count from hr\_jobs where job\_id = v\_job\_id;  if (v\_count = 1) then  delete from hr\_jobs where job\_id = v\_job\_id;  dbms\_output.put\_line(v\_job\_id || ' has been deleted.');  else  raise\_application\_error(-20203, 'No jobs deleted.');  end if;  commit;  end; |



|  |
| --- |
| create or replace procedure get\_employee  (v\_emp\_id in hr\_employees.employee\_id%type,  v\_salary out hr\_employees.salary%type,  v\_job\_id out hr\_employees.job\_id%type) is  begin  select salary, job\_id into v\_salary, v\_job\_id from hr\_employees where employee\_id = v\_emp\_id;  end;    declare  v\_job\_id hr\_employees.job\_id%type;  v\_salary hr\_employees.salary%type;  begin  get\_employee(107, v\_salary, v\_job\_id);  dbms\_output.put\_line('v\_salary');  dbms\_output.put\_line('----');  dbms\_output.put\_line(v\_salary);  dbms\_output.put\_line('v\_job');  dbms\_output.put\_line('----');  dbms\_output.put\_line(v\_job\_id);  end;    When I call the procedure for employee id 300, I get an error -01403 for no data found. If I added exception handlers for with no\_data\_found, I could display an error message. |