**Institute of Engineering & Management**

**Department of Computer Science & Engineering**

**Data-Base Management System Lab for 3rd year 6th semester 2019**

**Code: CS 691**

**Date:** 04/04/19

**WEEK-6**

**Assignment-1**

**Problem Statement:** Write a PL/SQL block to find the second highest salary from the customer table.

**SQL :**

SQL> CREATE OR REPLACE FUNCTION secondhighest

2 RETURN number IS

3 sech customers.salary%type;

4 BEGIN

5 SELECT max(salary) into sech

6 FROM customers

7 WHERE salary<( SELECT max(salary) FROM customers );

8 return sech;

9 END;

10 /

Function created.

SQL> DECLARE

2 BEGIN

3 dbms\_output.put\_line(secondhighest());

4 END;

5 /

2500

PL/SQL procedure successfully completed.

**Assignment-2**

**Problem Statement:** Write a PL/SQL block of code that first withdraws an amount of Rs. 1000. Then deposits an amount of Rs. 1,40,000. Update the current balance. Then check to see that the current balance of ALL the accounts in the bank does not exceed Rs. 2,00,000. If the balance exceeds, then undo the deposit just made.(Hint: create EMP\_MSTR table before writing this block)

**SQL :**

SQL> DECLARE

2 cid customers.id%type := &cid;

3 csal customers.salary%type := 200001;

4 BEGIN

5 UPDATE customers

6 SET salary=salary-100

7 WHERE id=cid;

8 COMMIT;

9 UPDATE customers

10 SET salary=salary+140000

11 WHERE id=cid;

12 SELECT salary INTO csal

13 FROM customers WHERE salary>200000;

14 dbms\_output.put\_line('Transaction Failed!');

15 ROLLBACK;

16 EXCEPTION

17 WHEN no\_data\_found THEN

18 dbms\_output.put\_line('Transaction complete!');

19 END;

20 /

Enter value for cid: 2

old 2: cid customers.id%type := &cid;

new 2: cid customers.id%type := 2;

Transaction complete!

PL/SQL procedure successfully completed.

SQL> select \* from customers;

ID NAME AGE ADDRESS SALARY

---------- --------- ---------- ---------- ----------

1 Ramesh 35 Ahmedabad 500

2 Chetan 20 Delhi 141400

3 Kapil 28 Delhi 875

4 Chaitali 25 Kolkata 4500

5 Bikash 28 Kolkata 1250

6 Sadhana 30 Pune 2500

7 Keshav 39 Mumbai 2250

7 rows selected.

**Assignment-3**

**Problem Statement:** The bank manager has decided to transfer employees accross branches. Write a PL/SQL block to accept an employee number and the branch number followed by updating the branch number of that employee to which he belongs appropriately. Display an appropriate message using SQL%FOUND based on the existence of the record in the EMP\_MSTR table. Otherwise, display message using SQL%NOTFOUND based on the non-existence of the record.

**SQL :**

SQL> DECLARE

2 eno emp\_mstr.empno%type := &eno;

3 edpt emp\_mstr.deptno%type := &edpt;

4 BEGIN

5 UPDATE emp\_mstr

6 SET deptno=edpt

7 WHERE empno=eno;

8 IF SQL%NOTFOUND THEN

9 dbms\_output.put\_line('No Employee Found!!');

10 ELSIF SQL%FOUND THEN

11 dbms\_output.put\_line('Employee found and changed dept.!!');

12 END IF;

13 END;

14 /

Enter value for eno: 7800

old 2: eno emp\_mstr.empno%type := &eno;

new 2: eno emp\_mstr.empno%type := 7800;

Enter value for edpt: 40

old 3: edpt emp\_mstr.deptno%type := &edpt;

new 3: edpt emp\_mstr.deptno%type := 40;

No Employee Found!!

PL/SQL procedure successfully completed.

**Assignment-4**

**Problem Statement:** The bank manager of Kolkata branch decides to activate all those accounts, which were previously marked as inactive for performing no transactions in last 365 days. Write a PL/SQL block to update the status of accounts. Display an appropriate message based on the number of rows affected by the update fired.(use SQL%ROWCOUNT)

**SQL :**

SQL> select \* from accounts;

ACC NAME STATUS

---------- ---------- ---------

1234 Ranajit active

2234 Arnab active

1233 Ankur active

1230 Attri inactive

1000 Swapnil inactive

2000 Random inactive

6 rows selected.

SQL> DECLARE

2 updated\_rows number(2);

3 BEGIN

4 UPDATE accounts

5 SET status='active'

6 WHERE status='inactive';

7 IF sql%notfound THEN

8 dbms\_output.put\_line('No customers were inactive');

9 ELSIF sql%found THEN

10 updated\_rows := sql%rowcount;

11 dbms\_output.put\_line(updated\_rows || ' customers are updated');

12 END IF;

13 END;

14 /

3 customers are updated

PL/SQL procedure successfully completed.

SQL> select \* from accounts;

ACC NAME STATUS

---------- ---------- ---------

1234 Ranajit active

2234 Arnab active

1233 Ankur active

1230 Attri active

1000 Swapnil active

2000 Random active

6 rows selected.

**Assignment-5**

**Problem Statement:** The bank manager has decided to mark all those accounts as inactive(I) on which there are no transactions performed in last 365 days. Whenever any such update takes place, a record for the same is maintained in the INACTV table comprising of the account number, the opening date and the type of account.Write a PL/SQL block to do the same.

**SQL :**

SQL> select \* from inactv;

no rows selected

SQL> select \* from transactions;

SNO ACC TYPE TDATE

---------- ---------- -------- ---------

1 1234 savings 01-JAN-19

2 2234 current 01-JAN-18

3 1233 current 01-JUN-18

4 1230 savings 01-FEB-18

5 1000 savings 01-NOV-17

6 2000 current 01-DEC-18

7 1234 savings 01-DEC-17

7 rows selected.

SQL> DECLARE

2 trdate transactions.tdate%type;

3 years number(4);

4 dacc transactions.acc%type;

5 CURSOR trans is SELECT max(tdate) FROM transactions WHERE acc=dacc;

6 CURSOR dtrans is SELECT distinct acc FROM transactions;

7 BEGIN

8 OPEN dtrans;

9 LOOP

10 FETCH dtrans into dacc;

11 EXIT WHEN dtrans%notfound;

12 OPEN trans;

13 LOOP

14 FETCH trans into trdate;

15 EXIT WHEN trans%notfound;

16 years := ceil( (sysdate - trdate)/365 );

17 IF years>1 THEN

18 INSERT INTO inactv values(dacc);

19 dbms\_output.put\_line('Acc no '|| dacc ||' is made inactive');

20 ELSE

21 dbms\_output.put\_line('Acc no '|| dacc ||' is active');

22 END IF;

23 END LOOP;

24 CLOSE trans;

25 END LOOP;

26 CLOSE dtrans;

27 END;

28 /

Acc no 1000 is made inactive

Acc no 1234 is active

Acc no 1233 is active

Acc no 2234 is made inactive

Acc no 1230 is made inactive

Acc no 2000 is active

PL/SQL procedure successfully completed.

SQL> select \* from inactv;

ACC

----------

1000

2234

1230

**Assignment-6**

**Problem Statement:** Write a PL/SQL block of code that will display the customer name, the fixed deposit number and the fixed deposit amount of the first 5 customers holding the highest amount in fixed deposits.

**SQL :**

SQL> select \*from customers;

CNAME FDNO AMNT

-------- ---------- ----------

Ranajit 1234 999999

Ankur 1235 199000

Arnab 1236 600000

Random1 1237 80000

Random2 1238 400000

Random3 1239 260000

Random4 1240 2000

7 rows selected.

SQL> DECLARE

2 name customers.cname%type;

3 f\_no customers.fdno%type;

4 f\_amnt customers.amnt%type;

5 CURSOR fd is select cname, fdno, amnt from customers order by amnt desc;

6 BEGIN

7 OPEN fd;

8 FOR i in 1..5

9 LOOP

10 FETCH fd into name,f\_no,f\_amnt;

11 EXIT WHEN fd%Notfound;

12 dbms\_output.put\_line(name||' with '||f\_no||' has '||f\_amnt);

13 END LOOP;

14 CLOSE fd;

15 END;

16 /

Ranajit with 1234 has 999999

Arnab with 1236 has 600000

Random2 with 1238 has 400000

Random3 with 1239 has 260000

Ankur with 1235 has 199000

PL/SQL procedure successfully completed.