Assignment 5

IMPLICIT CURSOR:

Q1. The bank manager has decided to activate all those accounts which were previously marked as inactive for performing no transaction in last 365 days.

Write a PL/SQ block (using implicit cursor) to update the status of account, display an approximate message based on the no. of rows affected by the update.

(Use of %FOUND, %NOTFOUND, %ROWCOUNT)

SQL> select \* from bank;

ACCNO STATUS

101 inactive

102 inactive

103 active

104 active

105 inactive

SQL> set serveroutput on;

declare rc number(2);

begin update bank set status='active' where status='inactive';

if SQL%found then rc:=SQL%rowcount;

dbms\_output.put\_line('Status Changed');

dbms\_output.put\_line('No. of rows affected: '||rc);

end if;

if SQL%notfound then

dbms\_output.put\_line('Not Found');

end if;

end;

/

Status Changed

No. of rows affected: 3

PL/SQL procedure successfully completed.

SQL> select \* from bank;

ACCNO STATUS

101 active

102 active

103 active

104 active

105 active

EXPLICIT CURSOR:

Q2. Organization has decided to increase the salary of employees by 10% of existing salary, who are having salary less than average salary of organization. Whenever such salary updates take place, a record for the same is maintained in the increment\_salary table.

EMP (E\_no , Salary) increment\_salary(E\_no ,Salary)

SQL> select \* from emp;

E\_NO SALARY

1111 5000

2222 7000

3333 10000

SQL> select \* from increment\_salary;

no rows selected

SQL> set serveroutput on;

Declare s number(6);

up\_sal number(6);

Cursor crsr\_sal is select E\_no, Salary from emp;

mE\_no emp.E\_no%type;

mSalary emp.Salary%type;

Begin open crsr\_sal;

select avg(Salary) into s from emp;

if crsr\_sal%isopen then loop fetch crsr\_sal into mE\_no,mSalary;

exit when crsr\_sal%notfound ;

if crsr\_sal%found and mSalary<s then

update emp set Salary=(Salary+(0.1\*Salary)) where E\_no=mE\_no;

select Salary into up\_sal from emp where E\_no=mE\_no;

insert into increment\_salary values(mE\_no,up\_sal);

end if;

end loop;

end if;

end;

/

PL/SQL procedure successfully completed.

SQL> select \* from emp;

E\_NO SALARY

1111 5500

2222 7700

3333 10000

SQL> select \* from increment\_salary;

E\_NO SALARY

1111 5500

Q3. Write PL/SQL block using explicit cursor for following requirements:College has decided to mark all those students detained (D) who are havingattendance less than 75%. Whenever such update takes place, a record for the same is maintained in the D\_Stud table.

create table stud21(roll number(4), att number(4), status varchar(1));

create table d\_stud(roll number(4), att number(4));

SQL> select \* from stud21;

ROLL ATT S

1 75

2 70

3 80

SQL> set serveroutput on;

Declare

Cursor crsr\_att is select roll, att,status from stud21 where att<75;

mroll stud21.roll%type;

matt stud21.att%type;

mstatus stud21.status%type;

Begin

open crsr\_att;

if crsr\_att%isopen then

loop

fetch crsr\_att into mroll,matt,mstatus;

exit when crsr\_att%notfound;

If crsr\_att%found then

update stud21 set status='D' where roll=mroll;

insert into d\_stud values(mroll,matt);

end if;

end loop;

end if;

end;

/

PL/SQL procedure successfully completed.

SQL> select \* from stud21;

ROLL ATT S

1 75

2 70 D

3 80

SQL> select \* from d\_stud;

ROLL ATT

2 70

PARAMETERIZED CURSOR:

4. Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table N\_RollCall with the data available in the table O\_RollCall. If the data in the first table already exist in the second table then that data should be skipped.

SQL> select \* from O\_rollcall;

ROLL NAME

1 aaa

2 bbb

3 ccc

SQL> select \* from N\_rollcall;

ROLL NAME

3 ccc

set serveroutput on;

declare

cursor crsr\_class is select \* from O\_rollcall;

cursor crsr\_chk(str\_name varchar) is select roll from N\_rollcall where name =

str\_name;

str\_roll N\_rollcall.roll%type;

str\_name N\_rollcall.name%type;

v varchar(10);

begin

open crsr\_class;

loop

fetch crsr\_class into str\_roll,str\_name;

exit When crsr\_class%NOTFOUND;

open crsr\_chk(str\_name);

fetch crsr\_chk into V;

if crsr\_chk%FOUND then

dbms\_output.put\_line('stud'|| str\_name || 'exist');

else

dbms\_output.put\_line('stud'|| str\_name || ' not exist. Inserting in N\_rollcall table');

insert into N\_rollcall values(str\_roll,str\_name);

end if;

close crsr\_chk;

end loop;

close crsr\_class;

end;

/

SQL> select \* from N\_rollcall;

ROLL NAME

1 aaa

2 bbb

3 ccc

Q5. Write the PL/SQL block for following requirements using parameterized Cursor: Consider table EMP(e\_no, d\_no, Salary), department wise average salary should be inserted into new table dept\_salary(d\_no, Avg\_salary)

EXPLICIT CURSOR:

Cursor for loop:

Q6. Write PL/SQL block using explicit cursor: Cursor FOR Loop for following requirements: College has decided to mark all those students detained (D) who are having attendance less than 75%. Whenever such update takes place, a record for the same is maintained in the D\_Stud table.

create table stud21(roll number(4), att number(4), status varchar(1));

create table d\_stud(roll number(4), att number(4));

SQL> select \* from stud21;

ROLL ATT S

1011 70

1012 85

1013 75

1014 97

1015 80

SQL> select \* from d\_stud;

no rows selected

set serveroutput on;

declare

cursor crsr\_att is select roll, att, status from stud21 where att<75;

begin

for demo IN crsr\_att loop update stud21 set status='D' where roll=demo.roll;

insert into d\_stud values(demo.roll, demo.att);

end loop;

end;

/

PL/SQL procedure successfully completed.

SQL> select \* from stud21;

ROLL ATT S

1011 70 D

1012 85

1013 75

1014 97

1015 80

SQL> select \* from d\_stud;

ROLL ATT

1011 70