Database using cursors

DECLARE

CURSOR c1 IS SELECT last\_name, salary FROM employees;

my\_ename employees.last\_name%TYPE; my\_salary employees.salary%TYPE;

BEGIN

OPEN c1;

LOOP

FETCH c1 INTO my\_ename, my\_salary;

IF c1%FOUND THEN -- fetch succeeded DBMS\_OUTPUT.PUT\_LINE('Name = ' ||

my\_ename || ', salary = '|| my\_salary);

ELSE -- fetch failed, so exit loop

EXIT; END IF; END LOOP; CLOSE c1;

END;

PL/SQL Function to get the factorial of given number

set serveroutput on

declare

n number:=4;

result number;

function fact(n out number) return number AS f number

begin

f:=1;

n:=6;

for i in 1..n

loop

f:=f\*i;

end loop;

return f;

end fact;

begin

result:=fact(n);

dbms\_output.put\_line('n is...'||n);

dbms\_output.put\_line('Factorial of '||n||' is '||result);

end;

1. Creates a procedure ‘employer\_details’ which gives the details of the employee.

CREATE OR REPLACE PROCEDURE employer\_details

IS

CURSOR emp\_cur IS

SELECT first\_name, last\_name, salary FROM emp\_tbl;

emp\_rec emp\_cur%rowtype;

BEGIN

FOR emp\_rec in sales\_cur

LOOP

dbms\_output.put\_line(emp\_cur.first\_name || ' ' ||emp\_cur.last\_name

|| ' ' ||emp\_cur.salary);

END LOOP;

END;

2. let’s create a frunction called ''employer\_details\_func' similar to the one created in stored proc

CREATE OR REPLACE FUNCTION employer\_details\_func

RETURN VARCHAR(20);

IS

emp\_name VARCHAR(20);

BEGIN

SELECT first\_name INTO emp\_name

FROM emp\_tbl WHERE empID = '100';

RETURN emp\_name;

END;

3. create a procedure which gets the name of the employee when the employee id is passed.

CREATE OR REPLACE PROCEDURE emp\_name (id IN NUMBER, emp\_name OUT NUMBER)

IS

BEGIN

SELECT first\_name INTO emp\_name

FROM emp\_tbl WHERE empID = id;

END;

4. We can call the procedure ‘emp\_name’ in this way from a PL/SQL Block.

DECLARE

empName varchar(20);

CURSOR id\_cur SELECT id FROM emp\_ids;

BEGIN

FOR emp\_rec in id\_cur

LOOP

emp\_name(emp\_rec.id, empName);

dbms\_output.putline('The employee ' || empName || ' has id ' || emp-rec.id);

END LOOP;

END;

5. Using IN OUT parameter in procedures:

CREATE OR REPLACE PROCEDURE emp\_salary\_increase

(emp\_id IN emptbl.empID%type, salary\_inc IN OUT emptbl.salary%type)

IS

tmp\_sal number;

BEGIN

SELECT salary

INTO tmp\_sal

FROM emp\_tbl

WHERE empID = emp\_id;

IF tmp\_sal between 10000 and 20000 THEN

salary\_inout := tmp\_sal \* 1.2;

ELSIF tmp\_sal between 20000 and 30000 THEN

salary\_inout := tmp\_sal \* 1.3;

ELSIF tmp\_sal > 30000 THEN

salary\_inout := tmp\_sal \* 1.4;

END IF;

END;

5.To Display Empno,Name & Basic pay of 3 highest paid employees USING CURSOR

Declare

Eno salary.EMPNo%type;

Ename EMP.EmpName%type;

EBasic Salary.Basic%type;

CURSOR CUR\_SAL IS

select distinct salary.empno,emp\_name,Basic from Salary,emp

where Salary.Empno=Emp.Empno

order by Basic Desc;

begin

OPEN cur\_sal;

LOOP

FETCH CUR\_SAL into Eno,Ename,Ebasic;

ExIT WHEN CUR\_SAL%NOTFOUND;

EXIT WHEN (CUR\_SAL%ROWCOUNT > 3);

DBMS\_OUTPUT\_put\_line(Eno||' '||Ename || ' '||Ebasic);

END LOOP;

CLOSE CUR\_SAL;

END;

6.Using Cursor

declare

CURSOR C1 is select constant1,constant2 from constant\_values;

C2 c1%rowtype;

Length numeric;

Breadth numeric;

Radius numeric;

Height numeric;

Side numeric;

Ra numeric;

Cv numeric;

Ca numeric;

Sa numeric;

Sv numeric;

Sqa numeric;

Cyv numeric;

begin

OPEN C1

LOOP

EXIT WHEN C1%NOTFOUND;

FETCH C1 INTO C2;

if (C2.constant1 >0 and C2.constant2>0) then

Length := C2.constant1;

Radius :=C2.Constant1;

Breadth :=C2.constant2;

Height :=C2.constnat2;

Side :=C2.constant2;

Ra:=2\*Length \*Breadth;

Cyv :=3.14 \* Radius \*Radius \* Height;

Ca :=3.14 \* Radius \* Radius;

Sa :=4\*3.14\*Radius \* Radius;

Sv :=3.14 \* Radius \* Radius \*Radius \* 4/3;

Sqa := side \*side \*0.5;

Cv := Side \* Side \*Side;

end If;

insert into areacv(circle\_area,sphere\_area,sphere\_volume,square\_area,Cube\_volume,

rectangle\_area,Cylinder\_volume,input\_value1,input\_value2) values

(Ca,Sa,Sv,Sqa,Cv,Ra,Cyv,Radius,Side);

END LOOP;

CLOSE C1;

DBMS\_OUTPUT.PUT\_LINE('Inserted');

end;

7. Deletion or Updation Using CURSOR

In order to Deletion or Update rows, the cursor must be defined with FOR UPDATE Clause

Eg. To delete Records Where BASIC > 2500

Declare

REC\_SAL Salary%Rowtype;

CURSOR CUR\_SAL IS

SELECT \* FROM SALARY WHERE BASIC>2500 FOR UPDATE;

BEGIN

OPEN CUR\_SAL;

LOOP

FETCH CUR\_SAL INTO REC\_SAL;

EXIT WHEN CUR\_SAL%NOTFOUND;

DELETE FROM SALARY WHERE CURRENT OF CUR\_SAL;

END LOOP;

CLOSE CUR\_SAL;

END;

• FOR Update ,instead of delete command

UPDATE SALARY SET BASIC =5000 WHERE CURRENT OF

CUR\_VAL;