

EXP 15

FRAME AND EXECUTE THE APPROPRIATE PL/SQL CURSORS AND EXCEPTIONAL HANDLING FOR THE PROJECT

AIM: To execute CURSORS & EXCEPTIONAL
HANDLING in pl/sql

1) CODE IMPLEMENTATION WITH THE IMPLICIT CURSOR:

DECLARE

total_count number(30);

BEGIN

--updating a row

UPDATE TUTOR

SET TEACHER = 'Zen' where CODE = 1;

-- result in boolean, true returned if no rows affected

IF sql%notfound THEN

dbms_output.put_line('no subjects fetched');

-- result in boolean, true returned if any rows affected

ELSIF sql%found THEN

-- count the number of rows affected rows affected

total_count := sql%rowcount;

dbms_output.put_line(total_count || ' teacher name updated ');

END IF;

END;

/

```
SQL> DECLARE
  2     total_count number(30);
  3 BEGIN
  4
  5     --updating a row
  6     UPDATE TUTOR1
  7     SET TEACHER = 'Zen' where CODE = 1;
  8
  9     -- result in boolean, true returned if no rows affected
 10    IF sql%notfound THEN
 11        dbms_output.put_line('no subjects fetched');
 12
 13
 14        -- result in boolean, true returned if any rows affected
 15        ELSIF sql%found THEN
 16
 17            -- count the number of rows affected rows affected
 18            total_count := sql%rowcount;
 19            dbms_output.put_line( total_count || ' teacher name updated ');
 20        END IF;
 21    END;
 22 /
1 teacher name updated

PL/SQL procedure successfully completed.
```

2)CODE IMPLEMENTATION WITH EXPLICIT CURSOR:

DECLARE

-- cursor declaration

CURSOR t_tutorials is

SELECT code, subject, teacher FROM Tutor;

t_code Tutor.code%type;

t_subject Tutor.subject%type;

t_teacher Tutor.teacher%type;

BEGIN

-- opening a cursor

OPEN t_tutorials;

LOOP

-- fetching values from cursor

FETCH t_tutorials into t_code, t_subject, t_teacher;

EXIT WHEN t_tutorials%notfound;

-- printing in console

dbms_output.put_line('Code is: ' || t_code || ' ' || 'Subject is: ' || t_subject || ' ' || 'Teacher is: ' || t_teacher);

```

|| t_subject || ' Teacher is: ' || t_teacher);

END LOOP;

CLOSE t_tutorials;

END;

/

```

```

SQL> DECLARE
2  -- cursor declaration
3  CURSOR t_tutorials is
4  SELECT code, subject, teacher FROM Tutor1;
5  t_code Tutor1.code%type;
6  t_subject Tutor1.subject%type;
7  t_teacher Tutor1.teacher%type;
8  BEGIN
9
10  -- opening a cursor
11  OPEN t_tutorials;
12  LOOP
13
14  -- fetching values from cursor
15  FETCH t_tutorials into t_code, t_subject, t_teacher;
16  EXIT WHEN t_tutorials%notfound;
17
18  -- printing in console
19  dbms_output.put_line('Code is: ' || t_code || ' ' || 'Subject is: ' || t_subject || ' Teacher is: ' || t_teacher);
20  END LOOP;
21  CLOSE t_tutorials;
22  END;
23  /
Code is: 1 Subject is: Automation Teacher is: Zen
Code is: 4 Subject is: PLSQL Teacher is: Anand
Code is: 2 Subject is: Performance Teacher is: Arvind

PL/SQL procedure successfully completed.

```

OUTPUT

```

PL/SQL procedure successfully completed.

```

```

SQL> SELECT * FROM TUTOR;

```

CODE	SUBJECT	TEACHER	REVIEWS
1	Automation	Zen	five stars
4	PLSQL	Anand	four stars
2	Performance	Arvind	four stars

3)EXCEPTIONAL HANDLING

set serveroutput on;

create or replace procedure findGrade(n IN number, grade
OUT varchar) as

begin

declare

no_score exception;

begin

if n > 90 then

grade:='A';

elsif n > 80 then

grade:='B';

elsif n > 70 then

grade:='C';

elsif n > 60 then

grade:='D';

elsif n > 50 then

grade:='E';

elsif n <= 50 and n >=0 then

grade:='F';

else

raise no_score;

```
end if;
exception
when no_score then
dbms_output.put_line('Invalid score!');
end;
end;
/
declare
n number;
grade varchar(1);
begin
n:=-1;
findGrade(n, grade);
if grade is not null then
dbms_output.put_line('Grade: '||grade);
end if;
end;
```

```
SQL> declare
  2  n number;
  3  grade varchar(1);
  4  begin
  5  n:=-1;
  6  findGrade(n, grade);
  7  if grade is not null then
  8  dbms_output.put_line('Grade: '||grade);
  9  end if;
 10  end;
 11  /
Invalid score!

PL/SQL procedure successfully completed.
```

RESULT: Cursors and Exceptional Handling in PL/SQL were successfully implemented and executed

CHITRALEKHA.CH

RA191100301087