

Day-4 Assignment

1)

```
declare
n number;
i number;
begin
n:=&n;
for i in 1..10
loop
dbms_output.put_line(n||' x '||i||' = '||n*i);
end loop;
end;
```

2)

```
Begin
for loopcount in 1..50 loop
insert into temp values(loopcount);
end loop;
end;
/
```

3)

DECLARE

CURSOR z_emp_info IS

SELECT emp_id,

empname,

email,

salary

FROM employees where salary>=15000;

r_emp_info z_emp_info%ROWTYPE;

BEGIN

OPEN z_emp_info;

LOOP

FETCH z_emp_info INTO r_emp_info;

EXIT WHEN z_emp_info%NOTFOUND;

dbms_output.Put_line('Employees Information:: '

|| ' ID: '

||r_emp_info.emp_id

|| ' Name: '

||r_emp_info.empname

|| ' email: '

||r_emp_info.email

|| ' salary: '

||r_emp_info.salary);

END LOOP;

dbms_output.Put_line('Total number of rows : '

||z_emp_info%rowcount);

CLOSE z_emp_info; END; /

4)

```
CREATE OR REPLACE TRIGGER department_insert_update
    BEFORE INSERT OR UPDATE ON employees
    FOR EACH ROW
    DECLARE
        dup_flag INTEGER;
    BEGIN
        :NEW.empname := UPPER(:NEW.empname);
    END;
/
```

5)

```
DECLARE
    example employees%rowtype;
    BEGIN
        SELECT * INTO example FROM employees where emp_id=111;
    EXCEPTION
        WHEN no_data_found THEN
            dbms_output.put_line('employee data is unavailable');
    END;
/
```

6)

DECLARE

s VARCHAR2(10) := 'madam';

l VARCHAR2(20);

t VARCHAR2(10);

BEGIN

FOR i IN REVERSE 1..Length(s) LOOP

l := Substr(s, i, 1);

t := t

||"

||l;

END LOOP;

IF t = s THEN

dbms_output.Put_line(t

||"

||' is palindrome');

ELSE

dbms_output.Put_line(t

||"

||' is not palindrome');

END IF;

END;

/

7)

DECLARE

myex EXCEPTION;

i NUMBER;

BEGIN

FOR i IN (SELECT * FROM employees) LOOP

IF i.emp_id= 104 THEN

RAISE myex;

END IF;

END LOOP;

EXCEPTION

WHEN myex THEN

dbms_output.Put_line('Employee number already exist in employees table.');

END;

/

8)

CREATE OR REPLACE PROCEDURE empdel

AS

BEGIN

DELETE FROM employees WHERE emp_id=104;

dbms_output.put_line('deleted successfully!');

END;

/

EXECUTE empdel;