1. **Display the name of that manager who has joined the organization in the year 1999. if there is no output or if there is more than one output then instead of giving error the program must give proper message**.

SQL> set serveroutput on;

SQL> select \* from emp where empno=7788;

EMPNO ENAME JOB MGR HIREDATE SAL

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

COMM DEPTNO

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

7788 scott analyst 7566 09-DEC-82 3050

20

SQL> ed;

Wrote file afiedt.buf

1 DECLARE

2 vename emp.ename % TYPE;

3 vjob emp.job % TYPE;

4 error\_msg VARCHAR(100);

5 error\_code NUMBER;

6 BEGIN

7 SELECT ename,job INTO vename,vjob FROM emp WHERE hiredate

8 BETWEEN '01-jan-09' AND '31-dec-09';

9 dbms\_output.put\_line(vename||' '||vjob);

10 EXCEPTION

11 WHEN no\_data\_found THEN

12 dbms\_output.put\_line('no employee hired in 09');

13 WHEN too\_many\_rows THEN

14 dbms\_output.put\_line('more than one manager has joined in 09');

15 WHEN others THEN

16 error\_msg:=SUBSTR(sqlerrm,1,100);

17 error\_code:=SQLCODE;

18 dbms\_output.put\_line('error during block execution');

19 dbms\_output.put\_line(error\_msg||' '||error\_code);

20\* END;

21 /

no employee hired in 09

PL/SQL procedure successfully completed.

**SUBPROGRAMS IN PL/SQL USING PROCEDURES & FUNCTIONS**

**10.A Create a procedure incr and increase the salary of employee.**

SQL> ed;

Wrote file afiedt.buf

1 CREATE or REPLACE PROCEDURE incr ( e\_id NUMBER, amt NUMBER) IS

2 vsalary NUMBER;

3 salary\_missing EXCEPTION;

4 BEGIN

5 SELECT sal INTO vsalary FROM emp WHERE empno = e\_id;

6 IF vsalary IS NULL THEN

7 RAISE salary\_missing;

8 ELSE

9 UPDATE emp SET sal = sal + amt WHERE empno = e\_id;

10 END IF;

11 EXCEPTION

12 WHEN salary\_missing THEN

13 dbms\_output.put\_line( e\_id || 'has salary as NULL');

14 WHEN NO\_DATA\_FOUND THEN

15 dbms\_output.put\_line( e\_id|| 'no such NUMBER');

16\* END incr;

17 /

Procedure created.

SQL> EXECUTE incr(7902,3000);PL/SQL procedure successfully completed.

SQL> SELECT \* FROM emp WHERE empno=7876;

EMPNO ENAME JOB MGR HIREDATE SAL

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

COMM DEPTNO

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

7876 adams clerk 7788 12-JAN-83 1100

20

SQL> EXECUTE incr(7876,1100);

PL/SQL procedure successfully completed.SQL> SELECT \* FROM emp WHERE empno=7876;

EMPNO ENAME JOB MGR HIREDATE SAL

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

COMM DEPTNO

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

7876 adams clerk 7788 12-JAN-83 2200

20

**FUNCTION**

**10.B Create a function to calculate the bonus of particular employee. The function returns the value of the bonus amount.**

SQL> set serveroutput on;

SQL> ed;

Wrote file afiedt.buf

1 CREATE OR REPLACE FUNCTION bonus(id emp.empno%type)

2 RETURN number is

3 bsal emp.sal%type;

4 cc emp.comm%type;

5 amt number;

6 BEGIN

7 select sal, comm into bsal, cc from emp where empno=id;

8 amt := bsal + nvl(cc,0)+ 5000;

9 RETURN(amt);

10\* END bonus;

SQL> /

Function created.

SQL> ed;

Wrote file afiedt.buf

1 BEGIN

2 dbms\_output.put\_line (bonus(7369));

3\* END;

SQL> /

6175

**PACKAGE PROGRAM**

**11. Create a package with subprogram to display the bonus amount for employee.**

**PACKAGE DECLARATION**

SQL> ed;

Wrote file afiedt.buf

1 CREATE OR REPLACE PACKAGE emp\_bonus AS

2 PROCEDURE calc\_bonus (date\_hired emp.hiredate%TYPE, bonus number);

3\* END emp\_bonus;

4 /

Package created.

**PACKAGE BODY:**

SQL> ed;

Wrote file afiedt.buf

1 CREATE OR REPLACE PACKAGE BODY emp\_bonus AS

2 PROCEDURE calc\_bonus (date\_hired emp.hiredate%TYPE, bonus number ) IS

3 BEGIN

4 DBMS\_OUTPUT.PUT\_LINE ('Employees hired on ' || date\_hired || ' get bonus:-'||bonus);

5 END;

6\* END emp\_bonus;

SQL> /

Package body created.

**PL/SQL block with package:**

SQL> ed;

Wrote file afiedt.buf

1 DECLARE

2 dd emp.hiredate%type;

3 id emp.empno%type;

4 bonus number;

5 BEGIN

6 select hiredate, sal+comm into dd, bonus from emp where empno=&id; emp\_bonus.calc\_bonus(dd, bonus+1000);

7\* END;

SQL> /

Enter value for id: 7369

old 6: select hiredate, sal+comm into dd, bonus from emp where empno=&id; emp\_bonus.calc\_bonus(dd, bonus+1000);

new 6: select hiredate, sal+comm into dd, bonus from emp where empno=7369; emp\_bonus.calc\_bonus(dd, bonus+1000);

Employees hired on 17-DEC-80 get bonus:-2175

PL/SQL procedure successfully completed.

**TRIGGER**

**12. Write a trigger total sal to maintain a derived column totsal that stores total salary of all members in a department.**

1.Alter a table dept with an additional column totasal

SQL> alter table dept add totsal varchar(20);

Table altered.

2.UPDATE dept by totsal

SQL> update dept set totsal=(select sum(sal) from emp where emp.deptno=dept.deptno);

4 rows updated.

SQL> select \* from dept;

DEPTNO DNAME LOC TOTSAL

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

10 accounting new york 8750

20 research dallas 15025

30 sales chicago 9400

40 operations boston

3.Create the TRIGGER

SQL> ed;

Wrote file afiedt.buf

1 CREATE OR REPLACE TRIGGER total\_salary

2 AFTER DELETE OR INSERT OR UPDATE OF deptno,sal ON emp

3 FOR EACH ROW

4 BEGIN

5 IF DELETING OR(UPDATING AND:old.deptno=:new.deptno)

6 THEN

7 UPDATE dept SET totsal=totsal-:old.sal

8 WHERE deptno=:old.deptno;

9 END IF;

10 IF INSERTING OR(UPDATING AND:old.deptno=:new.deptno)

11 THEN

12 UPDATE dept SET totsal=totsal+:new.sal

13 WHERE deptno=:new.deptno;

14 END IF;

15 IF UPDATING AND :old.deptno=:new.deptno AND:old.sal!=:new.sal

16 THEN

17 UPDATE dept SET totsal=totsal-:old.sal+:new.sal

18 WHERE deptno=:new.deptno;

19 END IF;

20 IF(:new.sal<500 OR :new.sal>5000)

21 THEN

22 RAISE\_APPLICATION\_ERROR(-20230,'salary out of range');

23 END IF;

24\* END;

25 /

Trigger created.

4. With delete option on emp

SQL> select \* from dept;

DEPTNO DNAME LOC TOTSAL

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

10 accounting new york 8750

20 research dallas 15025

30 sales chicago 9400

40 operations boston

SQL> delete from emp where empno=7900;

1 row deleted.

SQL> select \* from dept;

DEPTNO DNAME LOC TOTSAL

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

10 accounting new york 8750

20 research dallas 15025

30 sales chicago 8450

40 operations boston

5. With insert option on emp

SQL> select \* from dept;

DEPTNO DNAME LOC TOTSAL

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

10 accounting new york 8750

20 research dallas 15025

30 sales chicago 8450

40 operations boston

SQL> insert into emp values(8000,'Karthik','dbaprgm',7782,'12-aug-08',800,500,30);

1 row created.

SQL> select \* from dept;

DEPTNO DNAME LOC TOTSAL

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

10 accounting new york 8750

20 research dallas 15025

30 sales chicago 9250

40 operations boston

SQL> insert into emp values(8000,'Karthik','dbaprgm',7782,'12-aug-08',8000,500,30);

SQL> insert into emp values(8000,'Karthik','dbaprgm',7782,'12-aug-08',8000,500,30);

insert into emp values(8000,'Karthik','dbaprgm',7782,'12-aug-08',8000,500,30)

\*

ERROR at line 1:

ORA-20230: salary out of range

ORA-06512: at "NANDU.TOTAL\_SALARY", line 19

6.With update option on emp

SQL> select \* from dept;

DEPTNO DNAME LOC TOTSAL

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

10 accounting new york 8750

20 research dallas 15025

30 sales chicago 9250

40 operations boston

SQL> update emp set sal=800 where empno=7876;

1 row updated.

SQL> select \* from dept;

DEPTNO DNAME LOC TOTSAL

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

10 accounting new york 8750

20 research dallas 12225

30 sales chicago 9250

40 operations boston