

```

DECLARE
A INT := &X;
BEGIN
FOR I IN REVERSE 1..10
LOOP
DBMS_OUTPUT.PUT_LINE(A||CHR(9)||'*'||CHR(9)||'||CHR(9)||'='||CHR(9)||A*I);
END LOOP;
END;

```

\*\*\*\*\*

-----  
\*\*\*\*\*

```

DECLARE
A INT := &X;
I INT:=1;
BEGIN
WHILE I<10
LOOP
DBMS_OUTPUT.PUT_LINE(A||CHR(9)||'*'||CHR(9)||'||CHR(9)||'='||CHR(9)||A*I);
I:=I+1;
END LOOP;
END;

```

-----  
\*\*\*\*\*

```

DECLARE
A INT := &X;
I INT:=1;
BEGIN
LOOP
DBMS_OUTPUT.PUT_LINE(A||CHR(9)||'*'||CHR(9)||'||CHR(9)||'='||CHR(9)||A*I);
I:=I+1;
EXIT WHEN I>10;
END LOOP;
END;

```

\*\*\*\*\*

-----  
\*\*\*\*\*

```

CREATE TABLE EMP9 AS SELECT * FROM EMP;
CREATE OR REPLACE PROCEDURE ABC(ID IN EMP9.EMPNO%TYPE)
AS
EDATA EMP9%ROWTYPE;
BEGIN
SELECT * INTO EDATA FROM EMP9 WHERE EMPNO=ID;
IF (EDATA.JOB='CLERK' OR EDATA.JOB='SALESMAN')
THEN
UPDATE EMP9 SET SAL=EDATA.SAL*1.05 WHERE EMPNO=ID;
ELSE
UPDATE EMP9 SET SAL=EDATA.SAL*1.075 WHERE EMPNO=ID;
END IF;
END;

```

\*\*\*\*\*

-----

\*\*\*\*\*

```
DECLARE
TYPE DATA_TYPE IS RECORD(
ROW INT,
ID INT,
ENAM VARCHAR2(10),
JOB VARCHAR2(9),
MGR NUMBER(4),
HIREDATE DATE,
SAL NUMBER(7,2),
COMM NUMBER(7,2),
DEPTNO NUMBER(2),
DNAME VARCHAR2(14),
LOC VARCHAR2(13)
);
EDATA DATA_TYPE;
BEGIN
SELECT      *          INTO          EDATA          FROM          (SELECT
ROWNUM,EMPNO,ENAME,JOB,MGR,HIREDATE,SAL,COMM,DEPTNO,DNAME,LOC FROM
(SELECT EMPNO,ENAME,JOB,MGR,HIREDATE,SAL,COMM,E.DEPTNO,DNAME,LOC FROM
EMP E JOIN DEPT D ON E.DEPTNO=D.DEPTNO ORDER BY SAL DESC)) WHERE
ROWNUM=1;
  DBMS_OUTPUT.PUT_LINE(EDATA.ID||'   '||EDATA.ENAM||'   '||EDATA.JOB||'   '||EDATA.
SAL);
END;
```

\*\*\*\*\*

\*\*\*\*\*

```
CREATE OR REPLACE PROCEDURE ABC(D NUMERIC)
AS
CURSOR      CUR      IS      SELECT      ROWNUM      AS      SNO,D.
DEPTNO,DNAME,LOC,ENAME,SAL,JOB,COMM FROM EMP E JOIN DEPT D ON E.
DEPTNO=D.DEPTNO WHERE D.DEPTNO=D;
INFO CUR%ROWTYPE;
BEGIN
OPEN CUR;
FETCH CUR INTO INFO;
DBMS_OUTPUT.PUT_LINE('DEPTNO '||INFO.DEPTNO);
DBMS_OUTPUT.PUT_LINE('DNAME '||INFO.DNAME);
DBMS_OUTPUT.PUT_LINE('LOCATION '||INFO.LOC);
DBMS_OUTPUT.PUT_LINE('=====');
DBMS_OUTPUT.PUT_LINE('S.NO   ENAME   SAL JOB COMM');
DBMS_OUTPUT.PUT_LINE('=====');
LOOP
DBMS_OUTPUT.PUT_LINE(INFO.SNO||'   '||INFO.ENAME||'   '||INFO.SAL||'   '||INFO.JOB||'
'||INFO.COMM);
FETCH CUR INTO INFO;
EXIT WHEN CUR%NOTFOUND;
END LOOP;
CLOSE CUR;
END;
```

\*\*\*\*\*

```
CREATE OR REPLACE PROCEDURE ABC1
AS
CURSOR CUR1 IS SELECT * FROM EMP WHERE SAL > 1000 AND JOB IN
('CLERK','MANAGER');
INFO CUR1%ROWTYPE;
BEGIN
OPEN CUR1;
LOOP
FETCH CUR1 INTO INFO;
EXIT WHEN CUR1%NOTFOUND;
IF (INFO.JOB='CLERK')
THEN
INSERT INTO TABL1 VALUES (INFO.EMPNO,INFO.ENAME,INFO.JOB,INFO.MGR,INFO.
HIREDATE,INFO.SAL*1.15,INFO.COMM,INFO.DEPTNO);
ELSE
INSERT INTO TABL1 VALUES (INFO.EMPNO,INFO.ENAME,INFO.JOB,INFO.MGR,INFO.
HIREDATE,INFO.SAL*1.2,INFO.COMM,INFO.DEPTNO);
END IF;
END LOOP;
CLOSE CUR1;
END;
```

\*\*\*\*\*

#### ASSINGMENT 1----

```
CREATE OR REPLACE PROCEDURE ABC(ID EMP.EMPNO%TYPE)
AS
CURSOR CUR IS SELECT * FROM EMP WHERE EMPNO=ID;
CURSOR CUR1(SAL1 INT) IS SELECT ROWNUM AS SNO,A.EMPNO,A.ENAME,B.ENAME AS
"MNAME",A.DEPTNO,A.SAL FROM EMP A LEFT OUTER JOIN EMP B ON B.EMPNO=A.
MGR WHERE A.SAL > SAL1;
EDATA CUR%ROWTYPE;
D CUR1%ROWTYPE;
BEGIN
OPEN CUR;
FETCH CUR INTO EDATA;
OPEN CUR1(EDATA.SAL);
DBMS_OUTPUT.PUT_LINE('SRNO    EMPNO    ENAME    MANAGER NAME
    DEPTNO');
LOOP
FETCH CUR1 INTO D;
EXIT WHEN CUR1%NOTFOUND;
DBMS_OUTPUT.PUT_LINE(D.SNO||CHR(9)||D.EMPNO||CHR(9)||D.ENAME||CHR(9)||D.
MNAME||CHR(9)||D.DEPTNO||CHR(9)||D.SAL);
END LOOP;
CLOSE CUR;
CLOSE CUR1;
END;
```

\*\*\*\*\*

\*\*\*\*\*

## ASSINGMENT 2-----

\*\*\*\*\*

\*\*\*\*\*

```
CREATE OR REPLACE FUNCTION VALIDATEDNO(D EMP.EMPNO%TYPE) RETURN
VARCHAR2
```

```
AS
```

```
CURSOR CUR IS SELECT DEPTNO FROM EMP WHERE DEPTNO=D;
```

```
Z INT;
```

```
BEGIN
```

```
OPEN CUR;
```

```
FETCH CUR INTO Z;
```

```
IF (CUR%NOTFOUND)
```

```
THEN
```

```
RETURN 'FALSE';
```

```
ELSE
```

```
RETURN 'TRUE';
```

```
END IF;
```

```
CLOSE CUR;
```

```
END;
```

```
CREATE OR REPLACE PROCEDURE ABC(X INT)
```

```
AS
```

```
Y VARCHAR2(5);
```

```
TYPE ED IS RECORD(DEPTNO INT,DNAME VARCHAR2(20),MNSAL INT,MXSAL INT,ASAL
INT,SSAL INT,COUNT INT);
```

```
EDATA ED;
```

```
BEGIN
```

```
Y:=VALIDATEDNO(X);
```

```
IF(Y='TRUE')
```

```
THEN
```

```
SELECT * INTO EDATA FROM (SELECT E.DEPTNO,DNAME,MIN(SAL) AS
MNSAL,MAX(SAL) AS MXSAL,AVG(SAL) AS ASAL,SUM(SAL) AS SSAL,COUNT(SAL) AS
COUNT FROM EMP E JOIN DEPT D ON E.DEPTNO=D.DEPTNO GROUP BY E.
DEPTNO,DNAME) WHERE DEPTNO=X;
```

```
DBMS_OUTPUT.PUT_LINE('DEPARTMENT NO. = '||X);
```

```
DBMS_OUTPUT.PUT_LINE('DEPARTMENT NAME = '||EDATA.DNAME);
```

```
DBMS_OUTPUT.PUT_LINE('MINIMUM SALARY = '||EDATA.MNSAL);
```

```
DBMS_OUTPUT.PUT_LINE('MAXIMUM SALARY = '||EDATA.MXSAL);
```

```
DBMS_OUTPUT.PUT_LINE('AVERAGE SALARY = '||EDATA.ASAL);
```

```
DBMS_OUTPUT.PUT_LINE('TOTAL SALARY = '||EDATA.SSAL);
```

```
ELSE
```

```
DBMS_OUTPUT.PUT_LINE('DEPTNO NOT PRESENT');
```

```
END IF;
```

```
END;
```

\*\*\*\*\*

DECLARE

```

A INT;
BEGIN
A:=ANNUALSAL(50);
DBMS_OUTPUT.PUT_LINE(A);
END;
*****

-----

*****

PRACTICE PRACTICE 1---ASSI.1
CREATE OR REPLACE FUNCTION SAL_CALC(ID INT) RETURN INT
AS
CURSOR CUR IS SELECT EMPNO FROM EMP WHERE EMPNO=ID;
Z CUR%ROWTYPE;
BEGIN
OPEN CUR;
FETCH CUR INTO Z;
IF(CUR%NOTFOUND)
THEN
RETURN 0;
ELSE
RETURN 1;
END IF;
END;
CREATE OR REPLACE PROCEDURE ABC(X INT)
AS
Y INT:=SAL_CALC(X);
CURSOR CUR IS SELECT * FROM EMP WHERE EMPNO=X;
EDATA CUR%ROWTYPE;
TYPE ED IS RECORD(DEPTNO INT,SSAL INT,ASAL INT);
EDATA1 ED;
BEGIN
IF (Y=1)
THEN
OPEN CUR;
FETCH CUR INTO EDATA;
SELECT * INTO EDATA1 FROM (SELECT DEPTNO,SUM(SAL) AS SSAL, AVG(SAL) AS
ASAL FROM EMP GROUP BY DEPTNO) WHERE DEPTNO=EDATA.DEPTNO;
DBMS_OUTPUT.PUT_LINE('EMPLOYEE NUMBER: '||EDATA.EMPNO);
DBMS_OUTPUT.PUT_LINE('NAME OF THE EMPLOYEE: '||EDATA.ENAME);
DBMS_OUTPUT.PUT_LINE('Total Salary paid in department: '||EDATA1.SSAL);
DBMS_OUTPUT.PUT_LINE('Average Salary: '||EDATA1.ASAL);
ELSE
DBMS_OUTPUT.PUT_LINE('EMPLOYEE NOT PRESENT');
END IF;
CLOSE CUR;
END;
*****

-----

*****

-----ASSINGMENT 2---PRACTICE1-----
CREATE OR REPLACE PROCEDURE ABC(M INT)

```



```

AS
CURSOR CUR IS SELECT ROWNUM AS SNO,A.MGR,A.EMPNO,A.ENAME,A.
HIREDATE,NVL(A.COMM,0) AS COMM,B.ENAME AS MNAME,C.DNAME FROM EMP A
JOIN EMP B ON A.MGR=B.EMPNO JOIN DEPT C ON A.DEPTNO=C.DEPTNO WHERE A.
MGR=M;
EDATA CUR%ROWTYPE;
BEGIN
OPEN CUR;
FETCH CUR INTO EDATA;
IF(CUR%NOTFOUND)
THEN
DBMS_OUTPUT.PUT_LINE('MGR NOT PRESENT');
ELSE
DBMS_OUTPUT.PUT_LINE('MANAGER ID: '||M);
DBMS_OUTPUT.PUT_LINE('MANAGER NAME: '||EDATA.MNAME);
DBMS_OUTPUT.PUT_LINE('DEPARTMENT NAME: '||EDATA.DNAME);
DBMS_OUTPUT.PUT_LINE('S.NO      EMP ID      EMP NAME      HIREDATE
      COMM');
LOOP
DBMS_OUTPUT.PUT_LINE(EDATA.SNO||CHR(9)||EDATA.
EMPNO||CHR(9)||EDATA.ENAME||CHR(9)||EDATA.HIREDATE||CHR(9)||EDATA.COMM);
FETCH CUR INTO EDATA;
EXIT WHEN CUR%NOTFOUND;
END LOOP;
END IF;
CLOSE CUR;
END;

```

\*\*\*\*\*

-----

\*\*\*\*\*

-----ASSINGMENT3--PRACTICE1-----

```

CREATE OR REPLACE FUNCTION ANNUALSAL(DE INT) RETURN INT
AS
CURSOR CUR IS SELECT ASAL FROM (SELECT DEPTNO, SUM(SAL)*12 AS ASAL FROM
EMP GROUP BY DEPTNO) WHERE DEPTNO=DE;
EDATA CUR%ROWTYPE;
DN VARCHAR2(20);
A INT;
BEGIN
OPEN CUR;
FETCH CUR INTO EDATA;
IF(CUR%NOTFOUND)
THEN
RETURN 0;
ELSE
A:=EDATA.ASAL;
RETURN A;
END IF;
CLOSE CUR;
END;
DECLARE

```

```

        A INT;
        DE INT:=&DEPTNO;
        DN VARCHAR2(20);
BEGIN
IF(ANNUALSAL(DE)=0)
THEN
    DBMS_OUTPUT.PUT_LINE('DEPARTMENT NOT PRESENT');
ELSE
    SELECT DNAME INTO DN FROM DEPT WHERE DEPTNO=DE;
    DBMS_OUTPUT.PUT_LINE('DEPT ID: '||DE);
    DBMS_OUTPUT.PUT_LINE(CHR(9));
    DBMS_OUTPUT.PUT_LINE('DEPT NAME: '||DN);
    DBMS_OUTPUT.PUT_LINE(CHR(9));
    DBMS_OUTPUT.PUT_LINE('ANNUAL SALARY: '||ANNUALSAL(DE));
    DBMS_OUTPUT.PUT_LINE(A);
END IF;
END;

```

\*\*\*\*\*

\*\*\*\*\*

-----ASSINGMENT4---PRACTICE1=====

```

CREATE OR REPLACE PROCEDURE ABC(S INT)
AS
CURSOR CUR IS SELECT EMPNO,ENAME,SAL,DEPTNO FROM EMP WHERE SAL >= S;
EDATA CUR%ROWTYPE;
BEGIN
OPEN CUR;
FETCH CUR INTO EDATA;
LOOP
DBMS_OUTPUT.PUT_LINE(EDATA.EMPNO||' '||EDATA.ENAME||' '||EDATA.SAL||' '||EDATA.
DEPTNO);
FETCH CUR INTO EDATA;
EXIT WHEN CUR%NOTFOUND;
END LOOP;
END;

```

```

CREATE OR REPLACE PROCEDURE ABC1(H DATE)
AS
CURSOR CUR IS SELECT EMPNO,ENAME,SAL,DEPTNO,HIREDATE FROM EMP WHERE
HIREDATE >= H;
EDATA CUR%ROWTYPE;
BEGIN
OPEN CUR;
FETCH CUR INTO EDATA;
LOOP
DBMS_OUTPUT.PUT_LINE(EDATA.EMPNO||' '||EDATA.ENAME||' '||EDATA.SAL||' '||EDATA.
DEPTNO||' '||EDATA.HIREDATE);
FETCH CUR INTO EDATA;
EXIT WHEN CUR%NOTFOUND;
END LOOP;
END;

```

```

*****DECLARE
BEGIN
EXEC ABC(950);
--EXEC ABC1(TO_DATE('1-1-1981','DD-MM-YYYY'));
END;*****

-----
*****
-----

=====ASSINGMENT1----PRACTICE2=====
CREATE OR REPLACE FUNCTION SAL_CALC(E INT) RETURN INT
AS
CURSOR CUR IS SELECT EMPNO FROM EMP WHERE EMPNO=E;
EDATA CUR%ROWTYPE;
BEGIN
OPEN CUR;
FETCH CUR INTO EDATA;
IF (CUR%NOTFOUND)
THEN
RETURN 0;
ELSE
RETURN 1;
END IF;
CLOSE CUR;
END;
CREATE OR REPLACE PROCEDURE ABC(E1 INT)
AS
V INT:=SAL_CALC(E1);
CURSOR CUR IS SELECT * FROM EMP WHERE EMPNO=E1;
EDATA CUR%ROWTYPE;
CURSOR CUR1(DE INT) IS SELECT TSAL,ASAL,MNSAL,MXSAL FROM (SELECT
DEPTNO,SUM(SAL) AS TSAL,AVG(SAL) AS ASAL,MIN(SAL) AS MNSAL,MAX(SAL) AS
MXSAL FROM EMP GROUP BY DEPTNO) WHERE DEPTNO=DE;
EDATA1 CUR1%ROWTYPE;
BEGIN
OPEN CUR;
FETCH CUR INTO EDATA;
OPEN CUR1(EDATA.DEPTNO);
FETCH CUR1 INTO EDATA1;
IF(V=1)
THEN
DBMS_OUTPUT.PUT_LINE('=====');
DBMS_OUTPUT.PUT_LINE('EMPLOYEE NUMBER: '||E1);
DBMS_OUTPUT.PUT_LINE('NAME: '||EDATA.ENAME);
DBMS_OUTPUT.PUT_LINE('DEPTNO: '||EDATA.DEPTNO);
DBMS_OUTPUT.PUT_LINE('=====');
DBMS_OUTPUT.PUT_LINE('TOTAL SALARY: '||EDATA1.TSAL);
DBMS_OUTPUT.PUT_LINE('AVERAGE SALARY: '||EDATA1.ASAL);
DBMS_OUTPUT.PUT_LINE('MINIMUM: '||EDATA1.MNSAL);
DBMS_OUTPUT.PUT_LINE('MAXIMUM: '||EDATA1.MXSAL);
ELSE
DBMS_OUTPUT.PUT_LINE('EMPLOYEE NOT PRESENT');

```



```

END IF;
CLOSE CUR;
CLOSE CUR1;
END;
*****
-----
*****
=====ASSINGMENT 5---PRACTICE2=====
CREATE TABLE BOOKDETAILS( BOOK_CODE VARCHAR2(6) PRIMARY KEY,ORDER_ID
NUMERIC(4) UNIQUE,AMT INT);
CREATE OR REPLACE PROCEDURE BOOK(B VARCHAR2,O NUMERIC,A INT)
AS
BEGIN
INSERT INTO BOOKDETAILS VALUES(B,O,A);
END;
*****
-----
*****
=====DAY5---ASSINGMENT 1=====
--select text from user_source where name='MAIN';--TO KNOW CODE OF THAT PACKAGE
OR PROCEDURE OR FUNCTION.
CREATE OR REPLACE PACKAGE MAIN
AS
E_INSERT INT:=0;
E_DELETE INT:=0;
PROCEDURE HIRE_EMP1(E NUMBER,EN VARCHAR2,J VARCHAR2,M NUMBER,H DATE,S
NUMBER,C NUMBER,D NUMBER);
PROCEDURE FIRE_EMP1(E1 NUMBER);
END MAIN;
CREATE OR REPLACE PACKAGE BODY MAIN
IS
FUNCTION VALIDATE(E3 INT) RETURN VARCHAR2
IS
CURSOR CUR IS SELECT EMPNO FROM EMP9 WHERE EMPNO=E3;
EDATA CUR%ROWTYPE;
BEGIN
OPEN CUR;
FETCH CUR INTO EDATA;
IF(CUR%NOTFOUND)
THEN
RETURN 'FALSE';
ELSE
RETURN 'TRUE';
END IF;
CLOSE CUR;
END VALIDATE;
PROCEDURE HIRE_EMP1(E NUMBER,EN VARCHAR2,J VARCHAR2,M NUMBER,H DATE,S
NUMBER,C NUMBER,D NUMBER)
IS
V VARCHAR2(5):=VALIDATE(E);
BEGIN

```

```

IF(V='FALSE')
THEN
  INSERT INTO EMP9 VALUES(E,EN,J,M,H,S,C,D);
  DBMS_OUTPUT.PUT_LINE('1 ROW INSERTED');
  DBMS_OUTPUT.PUT_LINE('TOTAL INSERTED= '||E_INSERT);
  E_INSERT:=E_INSERT+1;
ELSE
  DBMS_OUTPUT.PUT_LINE('EMPLOYEE NO. ALREADY IN USE');
END IF;
END HIRE_EMP1;
PROCEDURE FIRE_EMP1(E1 NUMBER)
IS
  V1 VARCHAR2(5):=VALIDATE(E1);
BEGIN
  IF(V1='TRUE')
  THEN
    DELETE FROM EMP9 WHERE EMPNO=E1;
    DBMS_OUTPUT.PUT_LINE('1 ROW DELETED');
    E_DELETE:=E_DELETE+1;
    DBMS_OUTPUT.PUT_LINE('TOTAL DELETED= '||E_DELETE);
  ELSE
    DBMS_OUTPUT.PUT_LINE('EMPLOYEE NOT EXIT');
  END IF;
END FIRE_EMP1;
END MAIN;
--EXEC MAIN.HIRE_EMP1(7778,'BIKRAM','MANAGER',7788,'6-APR-2017',50000,0,10);
--select text from user_source where name='MAIN' AND TYPE='PACKAGE' AND LINE>40;
*****
-----
*****

CREATE OR REPLACE PACKAGE MAIN
AS
  PROCEDURE HIGH(E1 INT);
  PROCEDURE RAISE1(E2 INT);
  FUNCTION CHEC(E3 INT) RETURN INT;
END MAIN;

CREATE OR REPLACE PACKAGE BODY  MAIN
IS
  PROCEDURE HIGH(E1 INT)
  IS

  CURSOR CHR IS SELECT DEPTNO FROM(SELECT ROWNUM AS SNO,DEPTNO,NU FROM
  (SELECT DEPTNO,COUNT(EMPNO) AS NU FROM EMP GROUP BY DEPTNO)ORDER BY NU
  DESC)WHERE SNO=1;
  EDATA CHR%ROWTYPE;
  EN INT:=MAIN.CHEC(E1);
  BEGIN
  IF(EN=1)
  THEN
  OPEN CHR;

```

```
FETCH CHR INTO EDATA;  
DBMS_OUTPUT.PUT_LINE(EDATA.DEPTNO);  
ELSE  
DBMS_OUTPUT.PUT_LINE('EMPLOYEE NOT PRESENT');  
END IF;  
CLOSE CHR;  
END HIGH;
```

```
PROCEDURE RAISE1(E2 INT)  
IS  
EM INT:=MAIN.CHEC(E2);  
BEGIN  
IF(EM=1)  
THEN  
UPDATE EMP9 SET SAL=SAL*1.03 WHERE JOB='MANAGER';  
UPDATE EMP9 SET SAL=SAL*1.07 WHERE JOB='CLERK';  
UPDATE EMP9 SET SAL=SAL*1.075 WHERE JOB NOT IN('MANAGER','CLERK');  
ELSE  
DBMS_OUTPUT.PUT_LINE('EMPLOYEE NOT PRESENT');  
END IF;  
END RAISE1;
```

```
FUNCTION CHEC(E3 INT) RETURN INT  
IS  
CURSOR CHR1 IS SELECT EMPNO FROM EMP WHERE EMPNO=E3;  
EDATA1 CHR1%ROWTYPE;  
BEGIN  
OPEN CHR1;  
FETCH CHR1 INTO EDATA1;  
IF(CHR1%NOTFOUND)  
THEN RETURN 0;  
ELSE  
RETURN 1;  
END IF;  
CLOSE CHR1;  
END CHEC;  
END MAIN;
```