

1.

a) creation and insertion of table DEPT_1102290 :

```
CREATE TABLE DEPT_1102290(DEPTNO NUMBER(3) NOT NULL, DNAME VARCHAR(20) NOT NULL, LOC  
VARCHAR(20) NOT NULL);
```

```
BEGIN
```

```
INSERT INTO DEPT_1102290(DEPTNO,DNAME,LOC)
```

```
VALUES(10,'RESEARCH','HDC');
```

```
INSERT INTO DEPT_1102290(DEPTNO,DNAME,LOC)
```

```
VALUES(20,'SALES','BDC');
```

```
INSERT INTO DEPT_1102290(DEPTNO,DNAME,LOC)
```

```
VALUES(30,'COMPUTERS','CDC');
```

```
END;
```

b) creation and insertion of table EMP_1102290 :

```
CREATE TABLE EMP_1102290(EMPNO NUMBER(3) NOT NULL, ENAME VARCHAR(20), SAL NUMBER(5),  
COMM NUMBER(5), DEPTNO NUMBER(3));
```

```
BEGIN
```

```
INSERT INTO EMP_1102290(EMPNO,ENAME,SAL,COMOM,DEPTNO)
```

```
VALUES(101,'ANTHONY',6000,NULL,10);
```

```
INSERT INTO EMP_1102290(EMPNO,ENAME,SAL,COMOM,DEPTNO)
```

```
VALUES(102,'JOHN',7600,1000,10);
```

```
INSERT INTO EMP_1102290(EMPNO,ENAME,SAL,COMOM,DEPTNO)
```

```
VALUES(103,'ANDERSON',4500,NULL,20);
```

```
INSERT INTO EMP_1102290(EMPNO,ENAME,SAL,COMOM,DEPTNO)
```

```
VALUES(104,'KEVIN',2300,500,30);
```

```
END;
```

2.

a) sequence for DEPTNO :

```
CREATE SEQUENCE DEPTNO_SEQ
```

MIN VALUE 10

MAX VALUE 30

START WITH 10

INCREAMENT BY 10

CACHE 0;

b) sequence for EMPNO :

CREATE SEQUENCE EMPNO_SEQ

MIN VALUE 101

MAX VALUE 104

START WITH 101

INCREAMENT BY 1

CACHE 0;

-.....-

3.

SELECT ENAME FROM EMP_1102290

WHERE LOWER(ENAME) LIKE ('ANTHONY','JOHN')

(OR)

SELECT DEPTNO = 10

FROM EMP_1102290

WHERE LOWER(ENAME)

-.....-

4.

creation of view :

CREATE VIEW MY_VIEW

AS

SELECT *

FROM EMP_1102290;

-.....-

5.

creation of synonym :

```
CREATE PUBLIC SYNONYM MY_VIEW  
FOR HR_INFO;
```

-.....-

6.

```
CREATE PACKAGE EMP_PK  
FUNCTION CHECK_MAXSAL(NSAL EMP.SAL%TYPE) RETURN BOOLEAN;  
PROCEDURE INSERT_TR(  
VEMPNO EMP.EMPNO%TYPE, VENAME EMP.ENAME%TYPE, VSAL EMP.SAL%TYPE) IS  
BEGIN  
SELECT CNT *  
INTO VCNT FROM EMP  
WHERE EMPNO=VEMPNO;  
IF VCNT=0 THEN  
IF CHECK_MAXSAL(VSAL) THEN  
INSERT INTO EMP(EMPNO,ENAME,SAL)  
VALUES(VEMPNO,VENAME,VSAL);  
DBMS_OUTPUT.PUT_LINE(INSERT_TR);  
ELSE  
RAISE EXCEPTION('NSAL IS NOT MORE THAN MAXSAL');  
END IF;  
ELSE  
RAISE EXCEPTION('INSERTION DOESN'T HAPPEN BECAUSE EMPNO ALREADY EXIST');  
END IF;  
COMMIT;  
END;  
FUNCTION CHECK_MAXSAL(NSAL EMP.SAL%TYPE) RETURN BOOLEAN;  
MAXSAL EMP.SAL%TYPE;  
BEGIN
```

```

SELECT MAX(SAL) INTO MAXSAL FROM EMP;

IF MAXSAL >= NSAL THEN

RETURN TRUE;

ELSE

RETURN FALSE;

END IF;

END;

PROCEDURE UPDATE_TR(
VEMPNO EMP.EMPNO%TYPE, VENAME EMP.ENAME%TYPE, VSAL EMP.SAL%TYPE) IS

BEGIN

SELECT CNT *

INTO VCNT FROM EMP

WHERE EMPNO=VEMPNO;

IF VCNT>0 THEN

IF CHECK MAXSAL(VSAL) THEN

UPDATE EMP SET ENAME=VENAME, SAL=VSAL

WHERE EMPNO=VEMPNO

DBMS_OUTPUT.PUT_LINE('UPDATE_TR');

ELSE

RAISE EXCEPTION('NSAL IS NOT MORE THAN MAXSAL');

END IF;

ELSE

RAISE EXCEPTION('EMPNO DOESN'T EXIST');

END IF;

COMMIT;

END;

PROCEDURE DELETE_TR(VEMPNO EMP.EMPNO%TYPE) IS

BEGIN

SELECT CNT *

```

```

INTO VCNT FROM EMP

WHERE EMPNO=VEMPNO;

IF VCNT>0 THEN

DELETE FROM EMP WHERE EMPNO=VEMPNO;

DBMS_OUTPUT.PUT_LINE('DELETE_TR');

ELSE

RAISE EXCEPTION('DELETION NOT APPLICABLE BECAUSE EMPNO DOESN'T EXIST');

END IF;

COMMIT;

END;

DECLARE

VDNAME EMP.DNAME%TYPE;

VENAME EMP.ENAME%TYPE;

BEGIN

SELECT DNAME,ENAME INTO VDNAME,VENAME

FROM EMP;

WHERE DNAME=VDNAME AND ENAME=VENAME;

DBMS_OUTPUT.PUT_LINE('RESOURCES' || DNAME);

DBMS_OUTPUT.PUT_LINE("ANTHONY JOHN" || ENAME);

DBMS_OUTPUT.PUT_LINE('SALES' || DNAME);

DBMS_OUTPUT.PUT_LINE('ANDERSON' || ENAME);

DBMS_OUTPUT.PUT_LINE('COMPUTERS' || DNAME);

DBMS_OUTPUT.PUT_LINE('KEVIN' || ENAME);

END;

```