



VIT[®]
UNIVERSITY
(Estd. u/s 3 of UGC Act 1956)

Winter – SEMESTER 2022 - 23
Course Code: MCSE506P
Course-Title: – Database Systems Lab
DIGITAL ASSIGNMENT - 3
(LAB)

Name: Nidhi Singh
Reg. No:22MAI0015

Slot- L29+L30

Faculty: Dr. KARTHIK G M - SCOPE

COMPOSITE DATA TYPE

6A. //DISPLAY THE TOTAL SALARY WHICH INCLUDES COMMISSION OF EMPNO.7369. IT SHOULD ALSO DISPLAY EMPLOYEE NAME, HIS DEPARTMENT DETAILS AND HIS OLD AND NEW SALARY.

SQL>

DECLARE

x NUMBER(4);

TYPE dr IS RECORD(dno dept.deptno % TYPE, vname dept.dname %
TYPE, vloc dept.loc % TYPE, name emp.ename %
TYPE,vsal emp.sal % TYPE,vcom emp.comm % TYPE,newsal
emp.sal % TYPE);

d dr;

BEGIN

SELECT ename,sal,comm,dept.deptno,dname,loc INTO d.name,d.vsal,
d.vcom,d.dno,d.vname,d.vloc FROM emp,dept WHERE emp.deptno
=dept.deptno AND empno=&x;

d.newsal:=d.vsal+NVL(d.vcom,0);

dbms_output.put_line(d.dno||' '||d.vname||' '||' '||d.vloc||' '||' '||d.vsal||' '||d.vcom||' '
||d.newsal);

END;

/

SQL> set serveroutput on;

SQL> ed

Wrote file afiedt.buf

```
1 DECLARE
2   x NUMBER(4);
3   TYPE dr IS RECORD(dno dept.deptno % TYPE, vname dept.dname %
4                     TYPE, vloc dept.loc % TYPE, name emp.ename %
5                     TYPE,vsal emp.sal % TYPE,vcom emp.comm %
6                     TYPE,newsal
7                     emp.sal % TYPE);
8   d dr;
9 BEGIN
10  SELECT ename,sal,comm,dept.deptno,dname,loc INTO d.name,d.vsal,
11         d.vcom,d.dno,d.vname,d.vloc FROM emp,dept WHERE emp.deptno
12         =dept.deptno AND empno=&x;
13  d.newsal:=d.vsal+NVL(d.vcom,0);
14  dbms_output.put_line(d.dno||' '||d.vname||' '||' '||d.vloc||' '||' '||d.
15  vsal||' '||
16  d.vcom||' '||d.newsal);
17* END;
18 /
```

Enter value for x: 7369

old 11: =dept.deptno AND empno=&x;

new 11: =dept.deptno AND empno=7369;

20 research dallas 800 300 1100

PL/SQL procedure successfully completed.

SQL>

SQL>

6B. //TO LOAD THE EMPLOYEE NAME AND SALARIES INTO PL/SQL TABLES AND THEN DISPLAY THE CONTENT OF THE TABLE.

SQL> ed

DECLARE

TYPE empnameTYPE IS TABLE OF emp.ename % TYPE INDEX BY
BINARY_INTEGER;

TYPE empsalTYPE IS TABLE OF emp.ename % TYPE INDEX BY
BINARY_INTEGER;

enamelist empnameTYPE;
salarylist empsalTYPE;

SUBSCRIPT BINARY_INTEGER:=1;
ctr NUMBER(2):=1;

BEGIN

FOR empree IN(SELECT ename,sal FROM emp)
LOOP
 enamelist(SUBSCRIPT):=empree.ename;
 salarylist(SUBSCRIPT):=empree.sal;
 SUBSCRIPT:=SUBSCRIPT+1;
END LOOP;

WHILE ctr<SUBSCRIPT
LOOP
 dbms_output.put_line(enamelist(ctr));
 dbms_output.put_line(salarylist(ctr));
 ctr:=ctr+1;
END LOOP;

END;

/

```
SQL> set serveroutput on;
```

```
SQL> ed
```

```
Wrote file afiedt.buf
```

```
1 DECLARE
2   TYPE empnameTYPE IS TABLE OF emp.ename % TYPE INDEX BY
3     BINARY_INTEGER;
4   TYPE empsalTYPE IS TABLE OF emp.ename % TYPE INDEX BY
5     BINARY_INTEGER;
6   enamelist empnameTYPE;
7   salarylist empsalTYPE;
8   SUBSCRIPT BINARY_INTEGER:=1;
9   ctr NUMBER(2):=1;
10 BEGIN
11   FOR empree IN(SELECT ename,sal FROM emp)
12   LOOP
13     enamelist(SUBSCRIPT):=empree.ename;
14     salarylist(SUBSCRIPT):=empree.sal;
15     SUBSCRIPT:=SUBSCRIPT+1;
16   END LOOP;
17   WHILE ctr<SUBSCRIPT
18   LOOP
19     dbms_output.put_line(enamelist(ctr));
20     dbms_output.put_line(salarylist(ctr));
21     ctr:=ctr+1;
22   END LOOP;
23* END;
```

```
SQL> /
```

```
smith
800
allen
1600
ward
1250
jones
2975
martin
1250
blake
2850
clark
2450
scott
3050
king
```

```
5000
turner
1500
adams
1100
james
950
ford
3000
miller
1300
```

PL/SQL procedure successfully completed.

SQL>

CURSOR MANAGEMENT IN PL/SQL

7A. USING CURSOR DISPLAY THE DETAILS OF ALL THE EMPLOYEE FROM EMP TABLES WHOSE SUM OF SAL AND COMM. IS MORE THAN RS.3000.

SQL> ed

DECLARE

```
    vempno emp.empno % TYPE;
    vename emp.ename % TYPE;
    vsal emp.sal % TYPE;
    vdeptno emp.deptno % TYPE;
```

```
    CURSOR c1 IS SELECT empno,ename,sal,deptno FROM emp WHERE
                    sal+NVL(comm,0)>3000;
```

BEGIN

```
    OPEN c1;
```

```
    LOOP
```

```
        FETCH c1 INTO vempno,vename,vsal,vdeptno;
```

```
        IF c1 % FOUND THEN
```

```
            dbms_output.put_line(vempno||' '||vename||' '||vsal||' '
                                   ||vdeptno);
```

```
        ELSE
```

```
            EXIT;
```

```
        END IF;
```

```
    END LOOP;
```

```
    CLOSE c1;
```

```
END;
```

```
/
```

SQL> set serveroutput on;

SQL> ed

Wrote file afiedt.buf

```
1 DECLARE
2 vempno emp.empno % TYPE;
3 vename emp.ename % TYPE;
4 vsal emp.sal % TYPE;
5 vdeptno emp.deptno % TYPE;
6 CURSOR c1 IS SELECT empno,ename,sal,deptno FROM emp WHERE
7 sal+NVL(comm,0)>3000;
8 BEGIN
9 OPEN c1;
10 LOOP
11 FETCH c1 INTO vempno,vename,vsal,vdeptno;
12 IF c1 % FOUND THEN
13 dbms_output.put_line(vempno||' '||vename||' '||vsal||' '
14 ||vdeptno);
15 ELSE
16 EXIT;
17 END IF;
18 END LOOP;
19 CLOSE c1;
20* END;
```

SQL> /

7788 scott 3050 20

7839 king 5000 10

PL/SQL procedure successfully completed.

SQL>

7B. DISPLAY EMPLOYEE NUMBER, SALARY AND DEPT FOR WHOM SAL+COMM.>3000.

SQL>> ed

```
DECLARE
    CURSOR c1 IS SELECT empno,ename,sal,deptno FROM emp WHERE sal + NVL
        (comm,0) >3000;
    crr c1%ROWTYPE;
BEGIN
    OPEN c1;
    LOOP
        Fetch c1 INTO crr;
        IF c1%FOUND THEN
            dbms_output.put_line(crr.empno|| ' '|| crr.sal|| ' '||crr.deptno);
        ELSE
            EXIT;
        END IF;
    END LOOP;
    CLOSE c1;
END;
/
```

SQL> ed

Wrote file afiedt.buf

```
1 DECLARE
2 CURSOR c1 IS SELECT empno,ename,sal,deptno FROM emp WHERE sal + NVL
3 (comm,0) >3000;
4 crr c1%ROWTYPE;
5 BEGIN
6 OPEN c1;
7 LOOP
8 Fetch c1 INTO crr;
9 IF c1%FOUND THEN
10 dbms_output.put_line(crr.empno|| ' '|| crr.sal|| ' '||crr.deptno);
11 ELSE
12 EXIT;
13 END IF;
14 END LOOP;
15 CLOSE c1;
16* END;
SQL> /
7788 3050 20
7839 5000 10
```

PL/SQL procedure successfully completed.

SQL>

7C. DISPLAY THE DETAILS OF EMPLOYEES BELONGING TO DEPARTMENT NO 10 GETTING SALARY >5000 AND STORES THE DETAILS OF EMPLOYEES BELONGING TO DEPARTMENT NO 30 AND GETTING SALARY > 5000 IN THE TEMP TABLE.

SQL>

```
DECLARE
    CURSOR c1(p1 NUMBER) IS SELECT empno,ename,sal,deptno FROM emp
                                WHERE deptno=p1;

    crr c1 % ROWTYPE;
    vdeptno NUMBER(2):=30;
BEGIN
    OPEN c1(10);
    LOOP
        FETCH c1 INTO crr;
        IF c1 % FOUND THEN
            IF crr.sal>2500 THEN
                dbms_output.put_line(crr.empno||' '||crr.ename||
                                     ' '||crr.sal||' '||crr.deptno);
            END IF;
        ELSE
            EXIT;
        END IF;
    END LOOP;
    CLOSE c1;

    OPEN c1(vdeptno);
    DECLARE
    LOOP
        FETCH c1 INTO crr;
        IF c1 % FOUND THEN
            IF crr.sal>3000 THEN
                INSERT INTO temp2 VALUES (crr.empno,
                                           crr.ename, crr.sal,crr.deptno);
            END IF;
        ELSE
            EXIT;
        END IF;
    END LOOP;
    CLOSE c1;
END;
/
```



```
SQL> create table temp2 (empno numeric(12,2),ename varchar(20),sal numeric(12,2),  
deptno numeric(12,2));
```

Table created.

```
SQL> ed
```

Wrote file afiedt.buf

```
1 DECLARE  
2 CURSOR c1(p1 NUMBER)IS SELECT empno,ename,sal,deptno FROM emp  
3 WHERE deptno=p1;  
4 crr c1 % ROWTYPE;  
5 vdeptno NUMBER(2):=30;  
6 BEGIN  
7 OPEN c1(10);  
8 LOOP  
9 FETCH c1 INTO crr;  
10 IF c1 % FOUND THEN  
11 IF crr.sal>2500 THEN  
12 dbms_output.put_line(crr.empno||' '||crr.ename||  
13 ' '||crr.sal||' '||crr.deptno);  
14 END IF;  
15 ELSE  
16 EXIT;  
17 END IF;  
18 END LOOP;  
19 CLOSE c1;  
20 OPEN c1(vdeptno);  
21 LOOP  
22 FETCH c1 INTO crr;  
23 IF c1 % FOUND THEN  
24 IF crr.sal>3000 THEN  
25 INSERT INTO temp2 VALUES (crr.empno,  
26 crr.ename, crr.sal,crr.deptno);  
27 END IF;  
28 ELSE  
29 EXIT;  
30 END IF;  
31 END LOOP;  
32 CLOSE c1;  
33* END;  
SQL> /
```

```
7839 king 5000 10
```

PL/SQL procedure successfully completed.

```
SQL>
```