**ASSIGNMENT 5**

Do the following query on the same table of Assignment No. 4 for 5.1

Answer the following queries:

CREATE TABLE CLIENT\_MASTER(

CLIENT\_NO VARCHAR2(6) PRIMARY KEY CHECK(CLIENT\_NO LIKE 'C%'),

NAME VARCHAR2(20) NOT NULL,

CITY VARCHAR2(15),

PINCODE NUMBER(8),

STATE VARCHAR2(15),

BAL\_DUE NUMBER(10,2));

INSERT ALL

INTO CLIENT\_MASTER VALUES ('C1','ARKA PRATIM','KOLKATA',700059,'WEST BENGAL',1000.00)

INTO CLIENT\_MASTER VALUES ('C2','SOUMYADEEP','MUMBAI',700002,'MAHARASHTRA',2000.00)

INTO CLIENT\_MASTER VALUES ('C3','DEBARGHYA','NOIDA',700078,'UTTAR PRADESH',3000.00)

INTO CLIENT\_MASTER VALUES ('C4','PRITHWISH','KOLKATA',800059,'WEST BENGAL',4000.00)

INTO CLIENT\_MASTER VALUES ('C5','SAGNIK','BENGALURU',900059,'KARNATAKA',5000.00)

SELECT \* FROM DUAL;

**1.List the names of all clients having ‘a’ as the third letter in their names.**

SELECT NAME FROM CLIENT\_MASTER WHERE NAME LIKE '--A%';

**2.List the clients who stay in a city whose first letter is ‘K’.**

SELECT \* FROM CLIENT\_MASTER WHERE CITY LIKE 'K%';

**3.List all the clients who stay in ‘Mumbai’ or ‘Kolkata’.**

SELECT \* FROM CLIENT\_MASTER WHERE CITY IN ('MUMBAI','KOLKATA');

**4.List all the clients whose BalDue is greater than value 1000.**

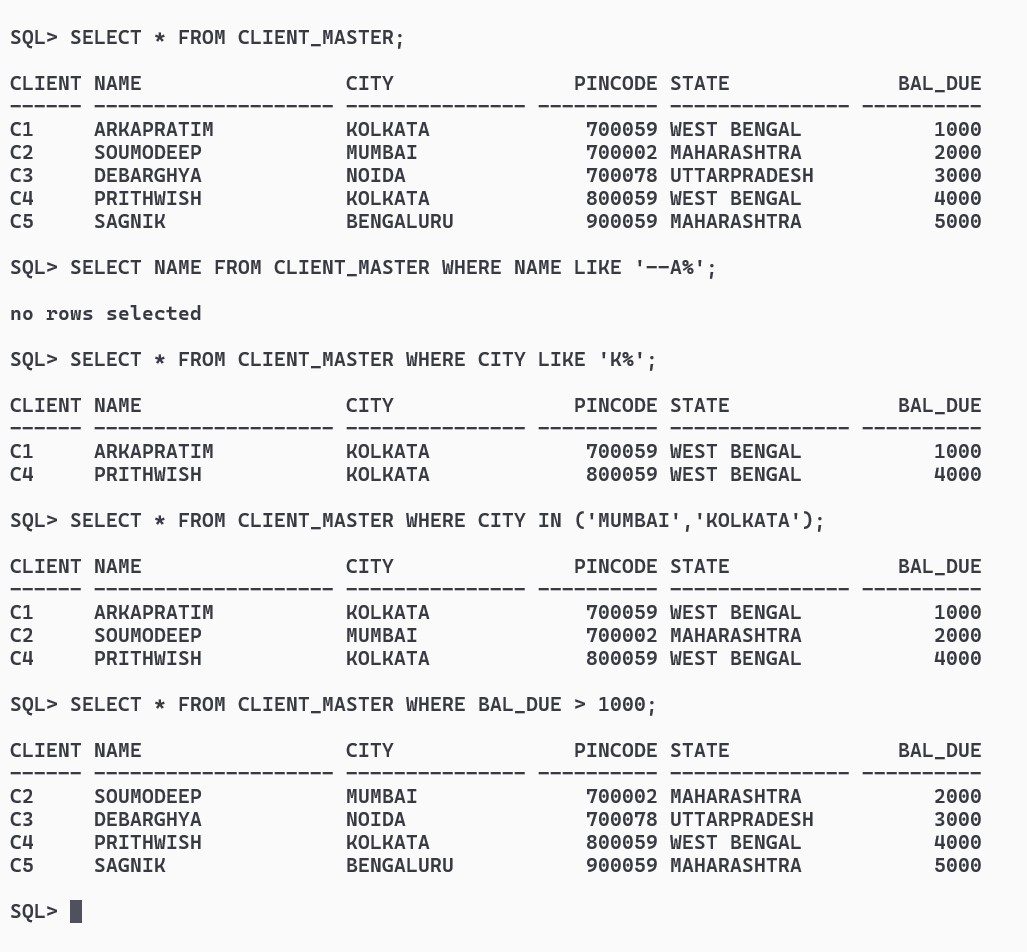
SELECT \* FROM CLIENT\_MASTER WHERE BAL\_DUE > 1000;

**5.List all information from the Sales\_Order table for orders placed in the month of June.**

UPDATE SALES\_ORDER SET ORDER\_DATE = (TO\_DATE('12-06-23','DD-MM-YYYY'))

WHERE ORDER\_NO = 'O001';

SELECT \* FROM SALES\_ORDER WHERE EXTRACT(MONTH FROM ORDER\_DATE) = '6';



**6.List the order information for Client\_no ‘C00001’ and ‘C00003’.**

SELECT \* FROM SALES\_ORDER\_DETAILS S WHERE S.ORDER\_NO IN

(SELECT ORDER\_NO FROM SALES\_ORDER WHERE CLIENT\_NO IN ('C1','C3'));

**7.List products whose selling price is greater than 500 and less than or equal to 750**

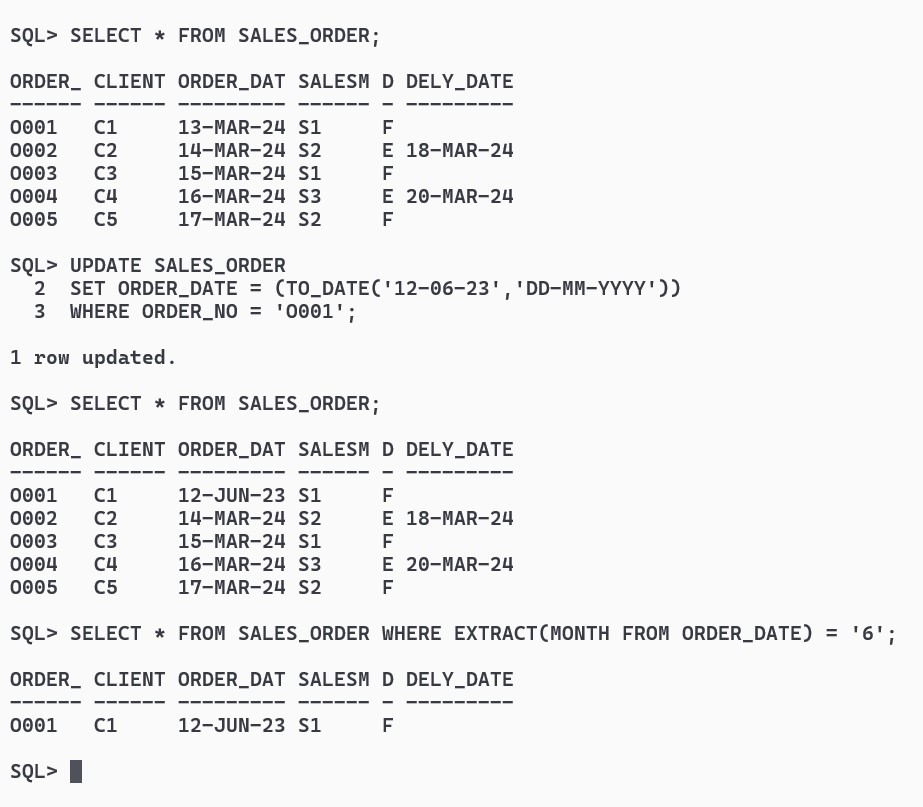
SELECT \* FROM PRODUCT\_MASTER WHERE SELL\_PRICE > 500 AND SELL\_PRICE <= 750;

**8.Count the total number of order.**

SELECT COUNT(ORDER\_NO) AS TOTAL\_ORDERS FROM SALES\_ORDER;

**9.Determine the maximum and minimum product prices. Rename the output as max\_price and min\_price respectively.**

SELECT MAX(SELL\_PRICE) max\_price, MIN(SELL\_PRICE) min\_price FROM PRODUCT\_MASTER;



**10.Count the number of client who live in Kolkata.**

SELECT COUNT(CITY) PEOPLE\_LIVING\_IN\_KOLKATA FROM CLIENT\_MASTER WHERE CITY = 'KOLKATA';

**11.Count the number of products having price less than or equal to 500.**

SELECT COUNT(SELL\_PRICE) PRICE\_LESSTHANEQUALTO\_500 FROM PRODUCT\_MASTER WHERE SELL\_PRICE <= 500;

**12.List the order number and day on which clients placed their order.**

SELECT ORDER\_NO , TO\_CHAR(ORDER\_DATE,'DAY') AS DAY FROM SALES\_ORDER;

**13.List the Order\_Date in the format ‘DD-Month-YY’.**

SELECT TO\_DATE(ORDER\_DATE,'DD-MONTH-YY') AS ORDER\_DATE FROM SALES\_ORDER;

**14.List the date, 20 days after today’s date.**

SELECT SYSDATE + 20 AFTER\_20\_DAYS FROM DUAL;

**15.List name of the client who has maximum BalDue.**

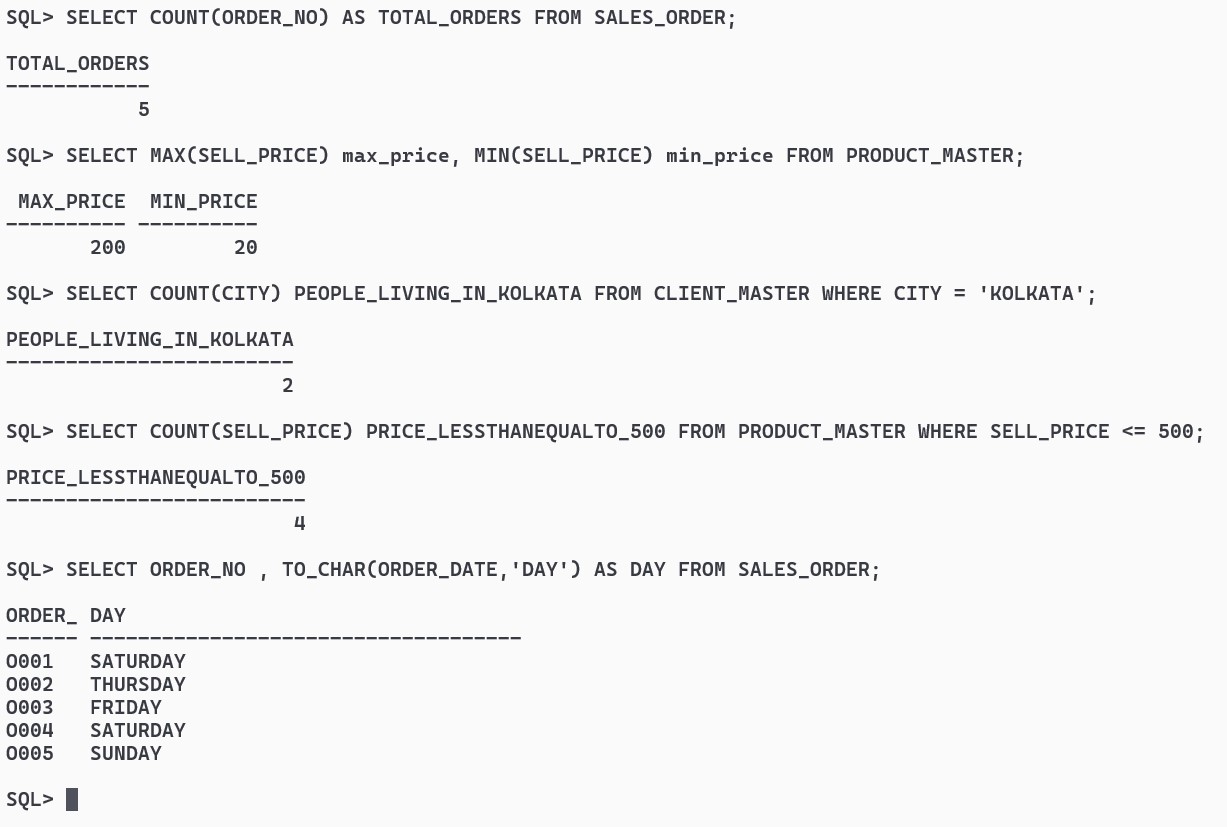
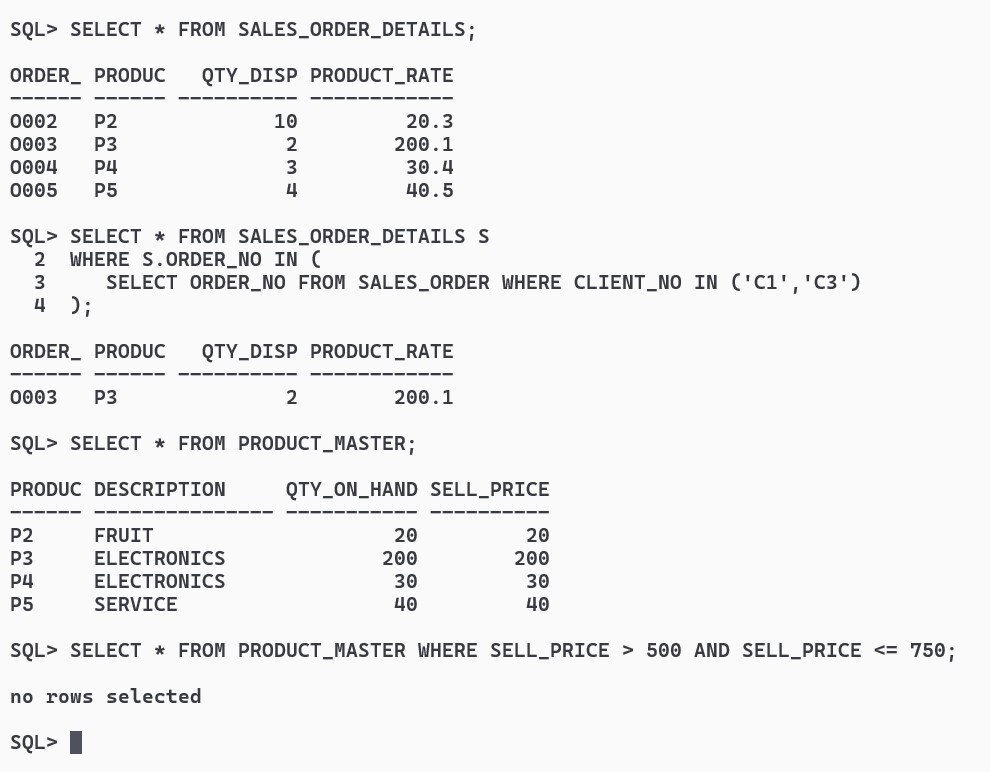
SELECT NAME FROM CLIENT\_MASTER WHERE BAL\_DUE = (SELECT MAX(BAL\_DUE) FROM CLIENT\_MASTER);

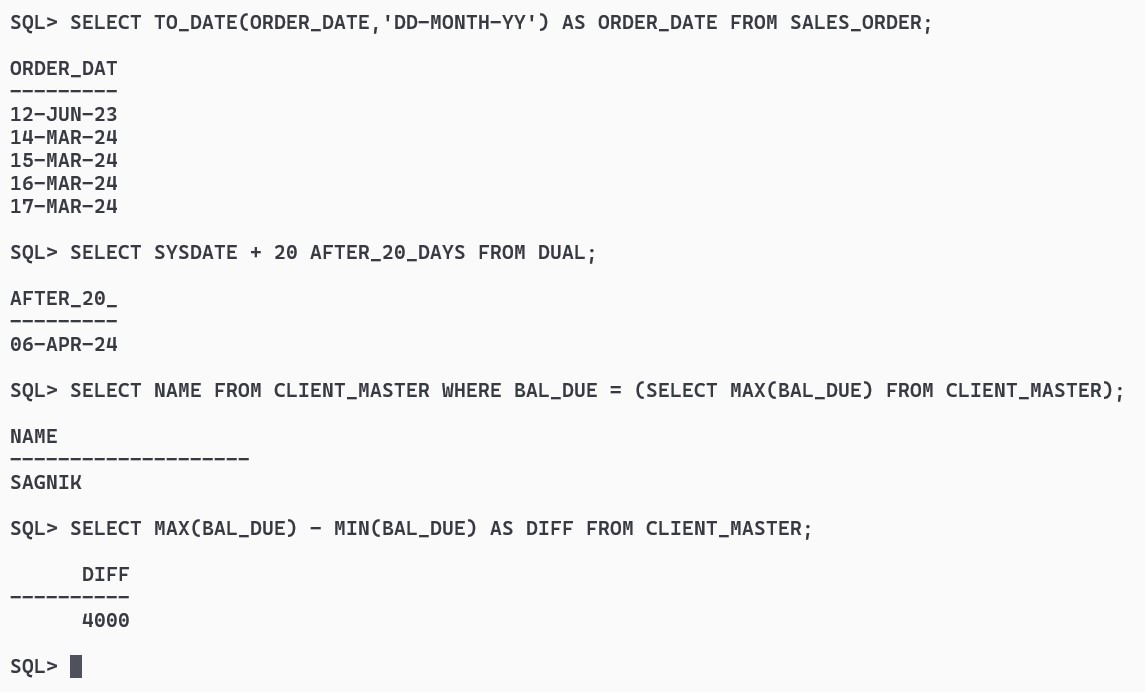
**16.Find the difference between maximum BalDue and minimum BalDue.**

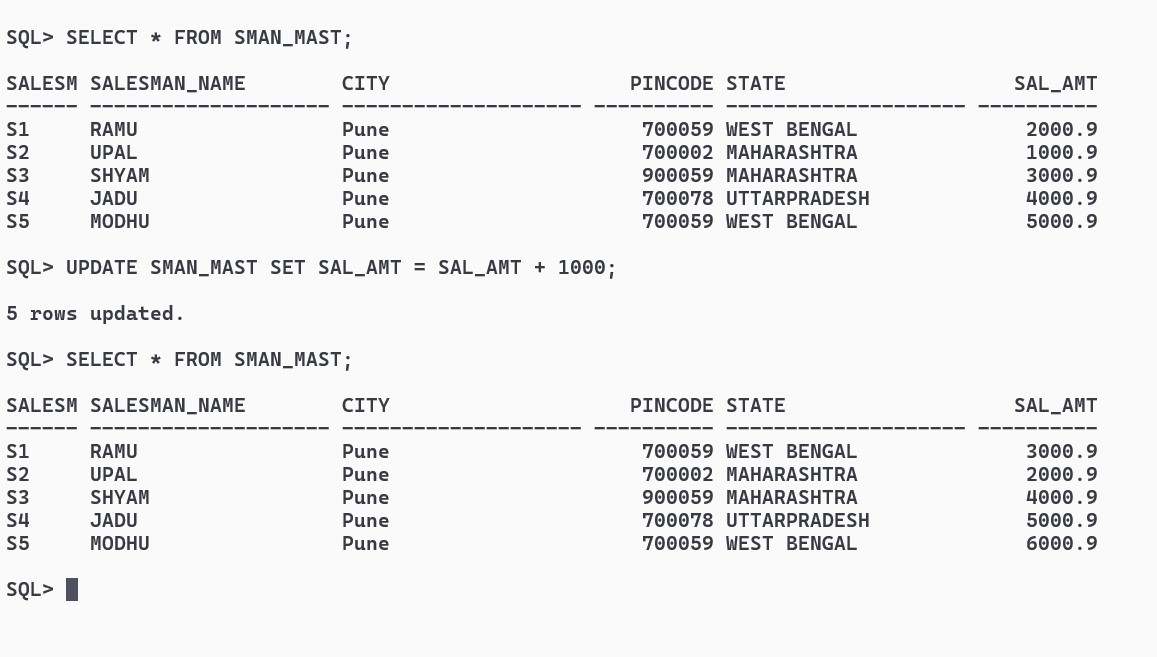
SELECT MAX(BAL\_DUE) - MIN(BAL\_DUE) AS DIFF FROM CLIENT\_MASTER;

**17.Add Rs.1000/- with the salary amount of every salesmen.**

UPDATE SMAN\_MAST SET SAL\_AMT = SAL\_AMT + 1000;







**Create the following tables and insert the values then do the queries for 5.2 employee:**

**emp\_no, name, dob, sex, address, salary**

**company: comp\_no, name, address**

**works: emp\_no, comp\_no**

CREATE TABLE EMPLOYEE (

EMP\_NO VARCHAR2(8) PRIMARY KEY CHECK(EMP\_NO LIKE 'E%'),

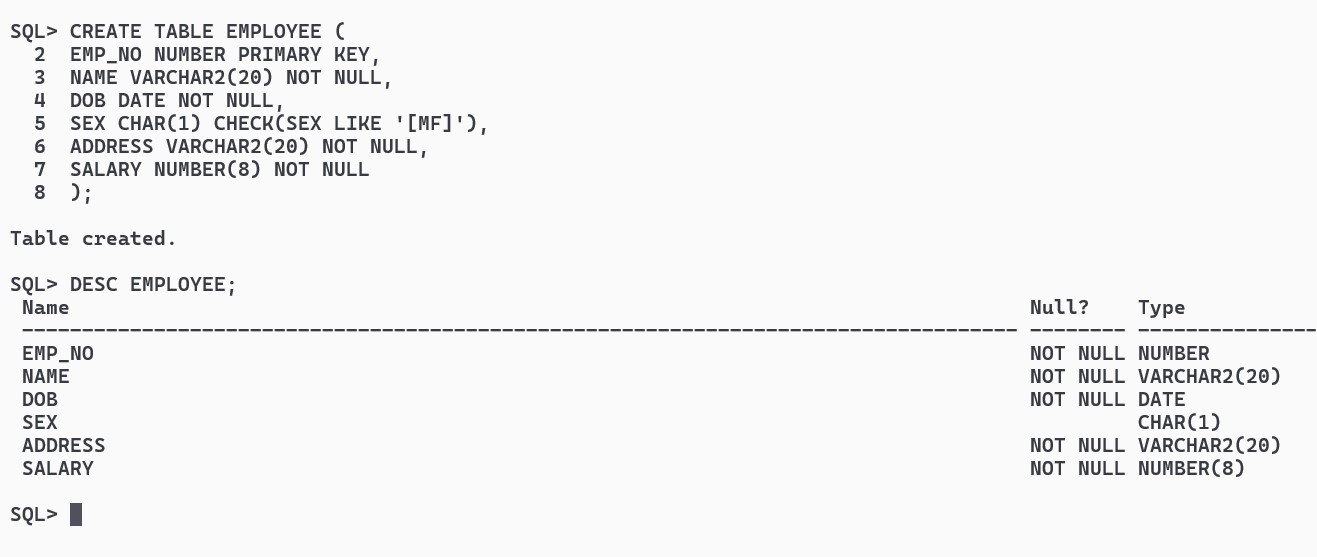
NAME VARCHAR2(20) NOT NULL,

DOB DATE NOT NULL,

SEX CHAR(1) CHECK(SEX IN ('M','F')),

ADDRESS VARCHAR2(20) NOT NULL,

SALARY NUMBER(8) NOT NULL);



CREATE TABLE COMPANY (

COMP\_NO VARCHAR2(8) PRIMARY KEY CHECK(COMP\_NO LIKE 'C%'),

NAME VARCHAR2(20) NOT NULL,

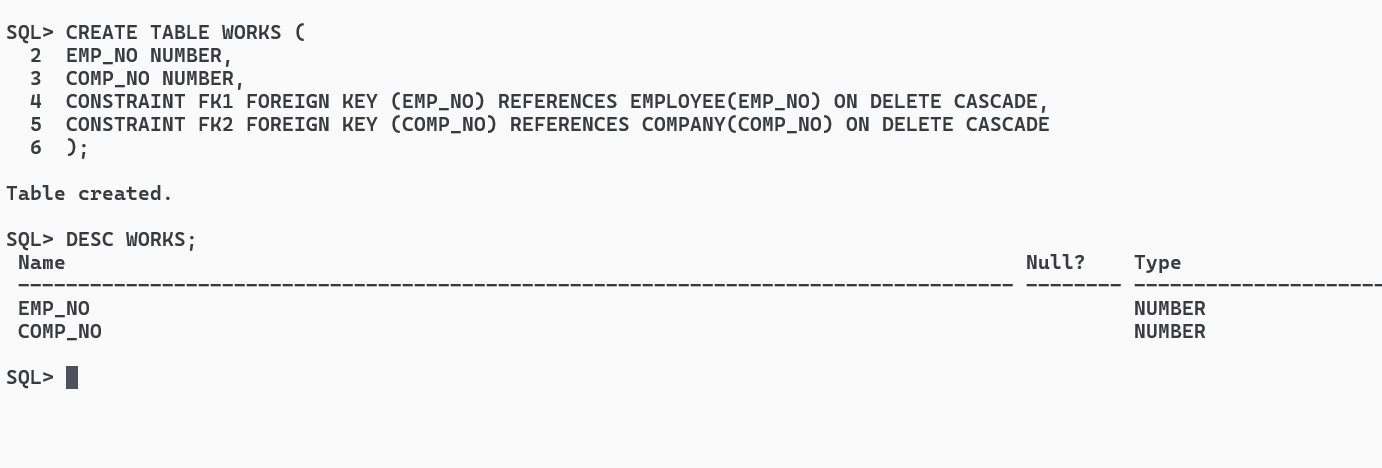
ADDRESS VARCHAR2(20) NOT NULL);



CREATE TABLE WORKS (

EMP\_NO VARCHAR2(8) NOT NULL,COMP\_NO VARCHAR2(8) NOT NULL,

CONSTRAINT FK1 FOREIGN KEY (EMP\_NO) REFERENCES EMPLOYEE(EMP\_NO) ON DELETE CASCADE,CONSTRAINT FK2 FOREIGN KEY (COMP\_NO) REFERENCES COMPANY(COMP\_NO) ON DELETE CASCADE);



INSERT ALL

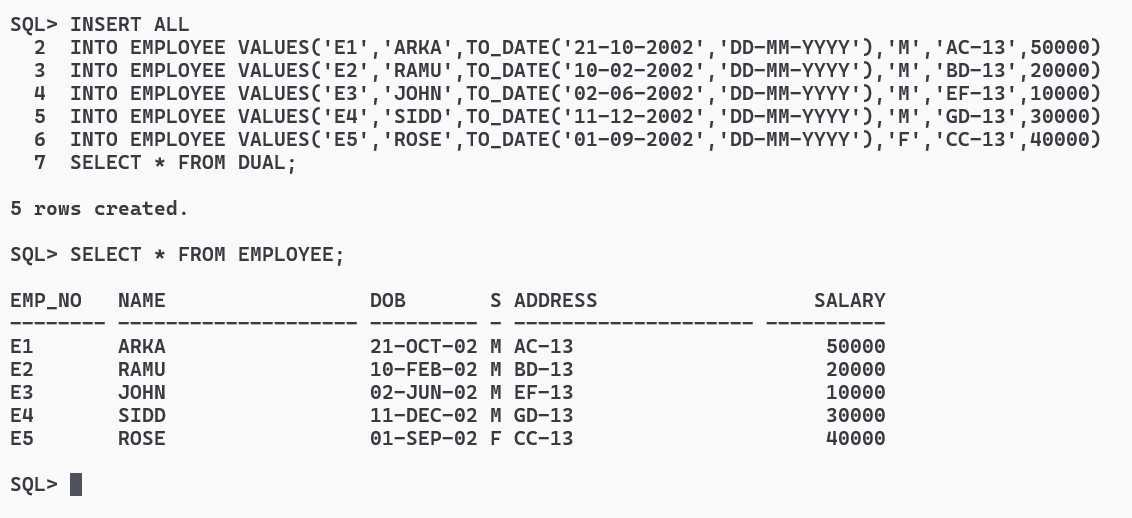
INTO EMPLOYEE VALUES('E1','ARKA',TO\_DATE('21-10-2002','DD-MM-YYYY'),'M','AC-13',50000)

INTO EMPLOYEE VALUES('E2','RAMU',TO\_DATE('10-02-2002','DD-MM-YYYY'),'M','BD-13',20000)

INTO EMPLOYEE VALUES('E3','JOHN',TO\_DATE('02-06-2002','DD-MM-YYYY'),'M','EF-13',10000)

INTO EMPLOYEE VALUES('E4','SIDD',TO\_DATE('11-12-2002','DD-MM-YYYY'),'M','GD-13',30000)

INTO EMPLOYEE VALUES('E5','ROSE',TO\_DATE('01-09-2002','DD-MM-YYYY'),'F','CC-13',40000)

SELECT \* FROM DUAL;

INSERT ALL

INTO COMPANY VALUES('C00001','ABC','SD-21')

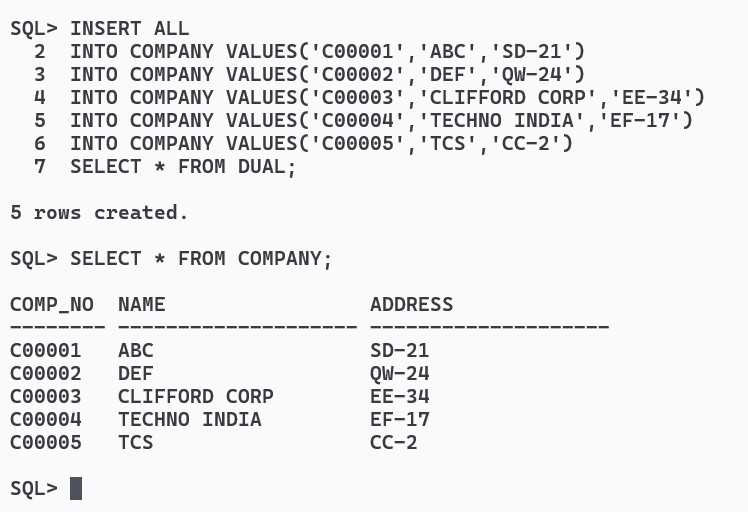
INTO COMPANY VALUES('C00002','DEF','QW-24')

INTO COMPANY VALUES('C00003','CLIFFORD CORP','EE-34')

INTO COMPANY VALUES('C00004','TECHNO INDIA','EF-17')

INTO COMPANY VALUES('C00005','TCS','CC-2')

SELECT \* FROM DUAL;



INSERT ALL

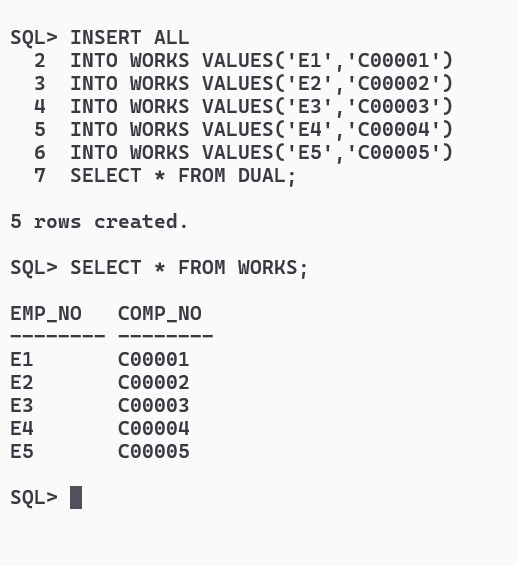
INTO WORKS VALUES('E1','C00001')

INTO WORKS VALUES('E2','C00002')

INTO WORKS VALUES('E3','C00003')

INTO WORKS VALUES('E4','C00004')

INTO WORKS VALUES('E5','C00005')

SELECT \* FROM DUAL;

**1.List the employees who work for company ‘C00002’**

SELECT \* FROM EMPLOYEE E WHERE E.EMP\_NO IN (SELECT EMP\_NO FROM WORKS WHERE COMP\_NO = 'C00002');

**2.List the employees who work for company ‘C00004’**

SELECT \* FROM EMPLOYEE E WHERE E.EMP\_NO IN (SELECT EMP\_NO FROM WORKS WHERE COMP\_NO = 'C00004');

**3.List the employees who work for Clifford Corp**

SELECT \* FROM EMPLOYEE E WHERE E.EMP\_NO IN (

SELECT EMP\_NO FROM WORKS W WHERE W.COMP\_NO IN (SELECT COMP\_NO FROM COMPANY WHERE NAME = 'CLIFFORD CORP'));

**4.List the employees whose name ends with ‘a’**

SELECT NAME FROM EMPLOYEE WHERE NAME LIKE '%A';

**5.List the employees born between 1999 and 2011**

SELECT NAME FROM EMPLOYEE WHERE EXTRACT(YEAR FROM DOB) BETWEEN '1999' AND '2011';

