ASSIGNMENT: 12.2

A. Consider the schema for Company Database:

EMPLOYEE (SSN, Name, Address, Sex, Salary, SuperSSN, DNo)

DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate)

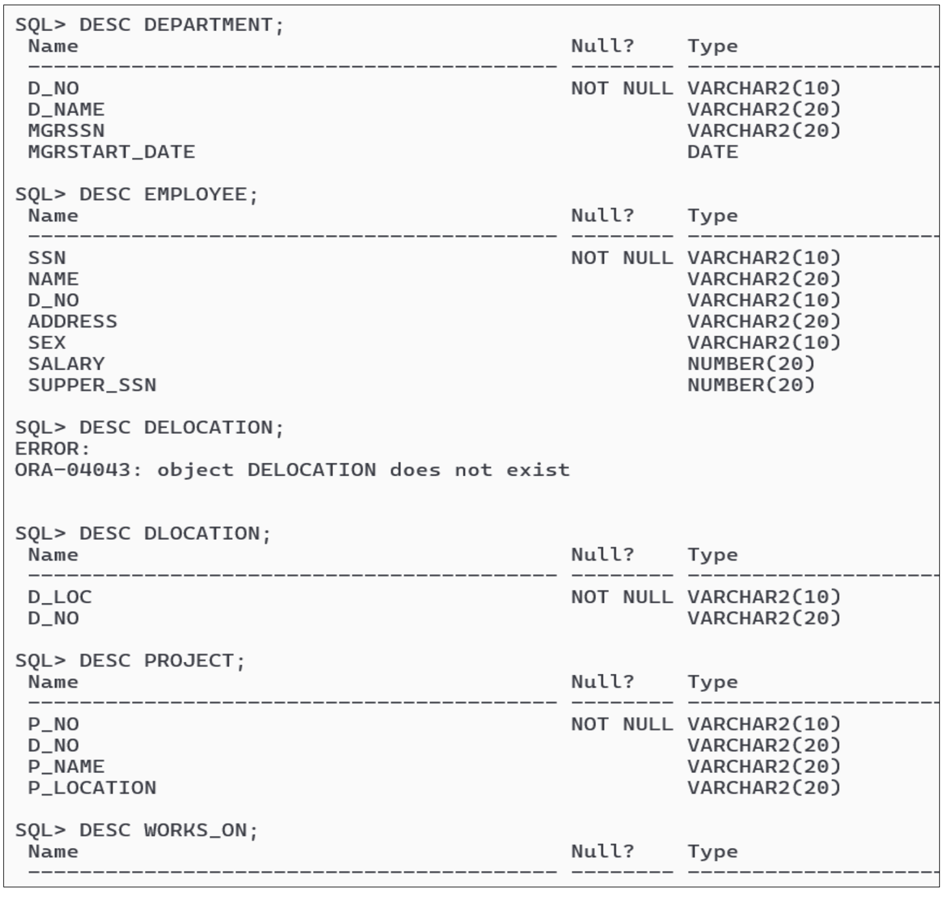
DLOCATION (DNo,DLoc)

PROJECT (PNo, PName, PLocation, DNo)

WORKS_ON (SSN, PNo, Hours)

TABLE CREATES:

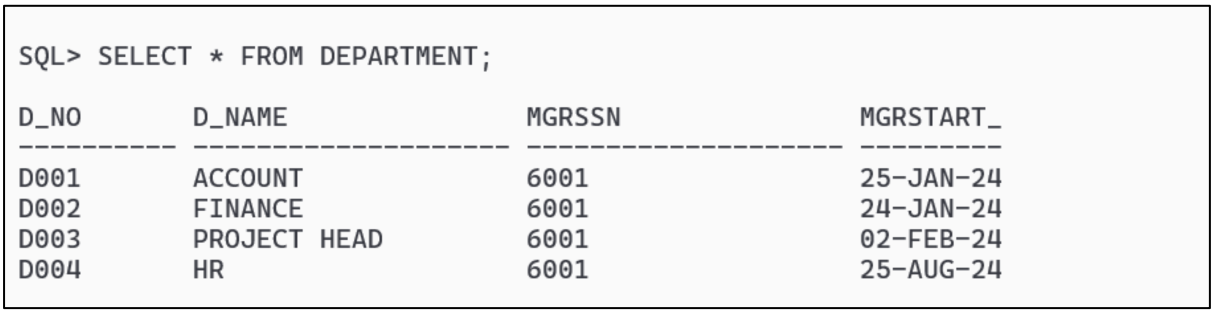
```
CREATE TABLE DEPARTMENT(
        D_NO VARCHAR2(10) PRIMARY KEY,
        D_NAME VARCHAR2(20),
        MGRSSN VARCHAR2(20),
        MGRSTART_DATE DATE
);
CREATE TABLE EMPLOYEE(
        SSN VARCHAR2(10) PRIMARY KEY,
        NAME VARCHAR2(20),
        D_NO VARCHAR2(10),
        ADDRESS VARCHAR2(20),
        SEX VARCHAR2(10),
        SALARY NUMBER(20),
        SUPPER_SSN NUMBER(20),
        FOREIGN KEY (D_NO) REFERENCES DEPARTMENT(D_NO) ON DELETE CASCADE
);
CREATE TABLE DLOCATION(
        D_LOC VARCHAR2(10) PRIMARY KEY,
        D_NO VARCHAR2(20),
        FOREIGN KEY (D_NO) REFERENCES DEPARTMENT(D_NO) ON DELETE CASCADE
);
CREATE TABLE PROJECT(
        P_NO VARCHAR2(10) PRIMARY KEY,
        D_NO VARCHAR2(20),
        P_NAME VARCHAR2(20),
        P_LOCATION VARCHAR2(20),
        FOREIGN KEY (D_NO) REFERENCES DEPARTMENT(D_NO) ON DELETE CASCADE
);
CREATE TABLE WORKS_ON(
        SSN VARCHAR2(10),
        P_NO VARCHAR2(10),
        HOURS VARCHAR2(20),
        FOREIGN KEY (P_NO) REFERENCES PROJECT(P_NO) ON DELETE CASCADE,
        FOREIGN KEY (SSN) REFERENCES EMPLOYEE(SSN) ON DELETE CASCADE
);
```



```
SQL> DESC DEPARTMENT;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 D_NO                                      NOT NULL VARCHAR2(10)
 D_NAME                                             VARCHAR2(20)
 MGRSSN                                             VARCHAR2(20)
 MGRSTART_DATE                                      DATE

SQL> DESC EMPLOYEE;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 SSN                                       NOT NULL VARCHAR2(10)
 NAME                                               VARCHAR2(20)
 D_NO                                               VARCHAR2(10)
 ADDRESS                                            VARCHAR2(20)
 SEX                                                VARCHAR2(10)
 SALARY                                             NUMBER(20)
 SUPPER_SSN                                         NUMBER(20)

SQL> DESC DELOCATION;
ERROR:
ORA-04043: object DELOCATION does not exist


SQL> DESC DLOCATION;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 D_LOC                                     NOT NULL VARCHAR2(10)
 D_NO                                               VARCHAR2(20)

SQL> DESC PROJECT;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
 P_NO                                      NOT NULL VARCHAR2(10)
 D_NO                                               VARCHAR2(20)
 P_NAME                                             VARCHAR2(20)
 P_LOCATION                                         VARCHAR2(20)

SQL> DESC WORKS_ON;
 Name                                      Null?    Type
 ----------------------------------------- -------- --------------------
```

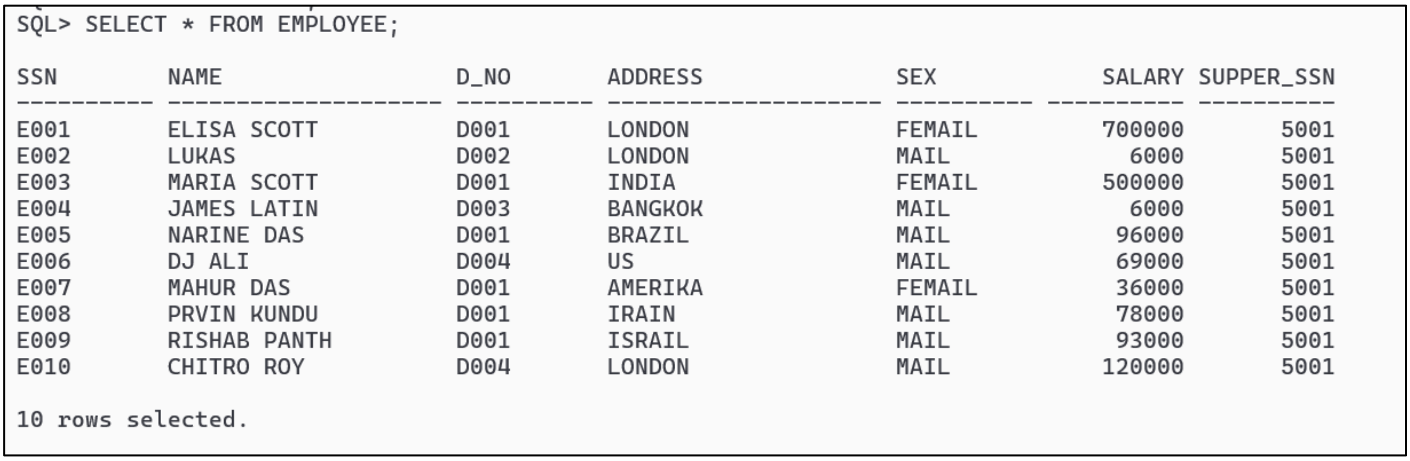
VALUE INSERT:

INSERT INTO DEPARTMENT VALUES('D001','ACCOUNT','6001','25-JAN-2024');
INSERT INTO DEPARTMENT VALUES('D002','FINANCE','6001','24-JAN-2024');
INSERT INTO DEPARTMENT VALUES ('D003','PROJECT HEAD','6001','2-FEB-2024');
INSERT INTO DEPARTMENT VALUES('D004','HR','6001','25-AUG-2024');



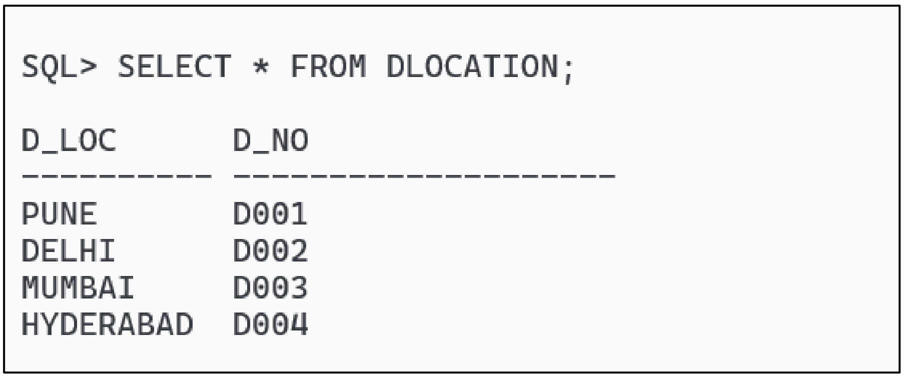
```
SQL> SELECT * FROM DEPARTMENT;

D_NO       D_NAME               MGRSSN               MGRSTART_
---------- -------------------- -------------------- ---------
D001       ACCOUNT              6001                 25-JAN-24
D002       FINANCE              6001                 24-JAN-24
D003       PROJECT HEAD         6001                 02-FEB-24
D004       HR                   6001                 25-AUG-24
```

INSERT INTO EMPLOYEE VALUES('E001','ELISA SCOTT','D001','LONDON','FEMAIL',700000,5001);
INSERT INTO EMPLOYEE VALUES('E002','LUKAS','D002','LONDON','MAIL',6000,5001);
INSERT INTO EMPLOYEE VALUES('E003','MARIA SCOTT','D001','INDIA','FEMAIL',500000,5001);
INSERT INTO EMPLOYEE VALUES('E004','JAMES LATIN','D003','BANGKOK','MAIL',6000,5001);
INSERT INTO EMPLOYEE VALUES('E005','NARINE DAS','D001','BRAZIL','MAIL',96000,5001);
INSERT INTO EMPLOYEE VALUES('E006','DJ ALI','D004','US','MAIL',69000,5001);
INSERT INTO EMPLOYEE VALUES('E007','MAHUR DAS','D001','AMERIKA','FEMAIL',36000,5001);
INSERT INTO EMPLOYEE VALUES('E008','PRVIN KUNDU','D001','IRAIN','MAIL',78000,5001);
INSERT INTO EMPLOYEE VALUES('E009','RISHAB PANTH','D001','ISRAIL','MAIL',93000,5001);
INSERT INTO EMPLOYEE VALUES('E010','CHITRO ROY','D004','LONDON','MAIL',120000,5001);



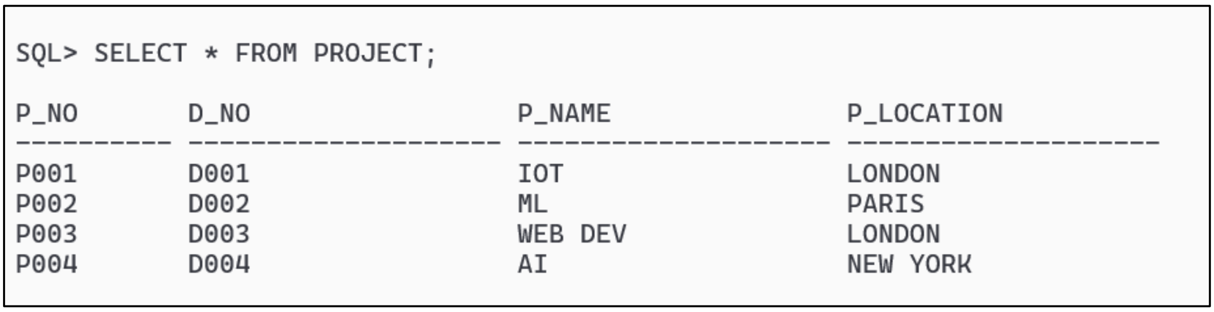
```
SQL> SELECT * FROM EMPLOYEE;

SSN        NAME                 D_NO       ADDRESS              SEX            SALARY SUPPER_SSN
---------- -------------------- ---------- -------------------- ---------- ---------- ----------
E001       ELISA SCOTT          D001       LONDON               FEMAIL         700000       5001
E002       LUKAS                D002       LONDON               MAIL             6000       5001
E003       MARIA SCOTT          D001       INDIA                FEMAIL         500000       5001
E004       JAMES LATIN          D003       BANGKOK              MAIL             6000       5001
E005       NARINE DAS           D001       BRAZIL               MAIL            96000       5001
E006       DJ ALI               D004       US                   MAIL            69000       5001
E007       MAHUR DAS            D001       AMERIKA              FEMAIL          36000       5001
E008       PRVIN KUNDU          D001       IRAIN                MAIL            78000       5001
E009       RISHAB PANTH         D001       ISRAIL               MAIL            93000       5001
E010       CHITRO ROY           D004       LONDON               MAIL           120000       5001

10 rows selected.
```

```
INSERT INTO DLOCATION VALUES('PUNE','D001');
INSERT INTO DLOCATION VALUES('DELHI','D002');
INSERT INTO DLOCATION VALUES('MUMBAI','D003');
INSERT INTO DLOCATION VALUES('HYDERABAD','D004');
```



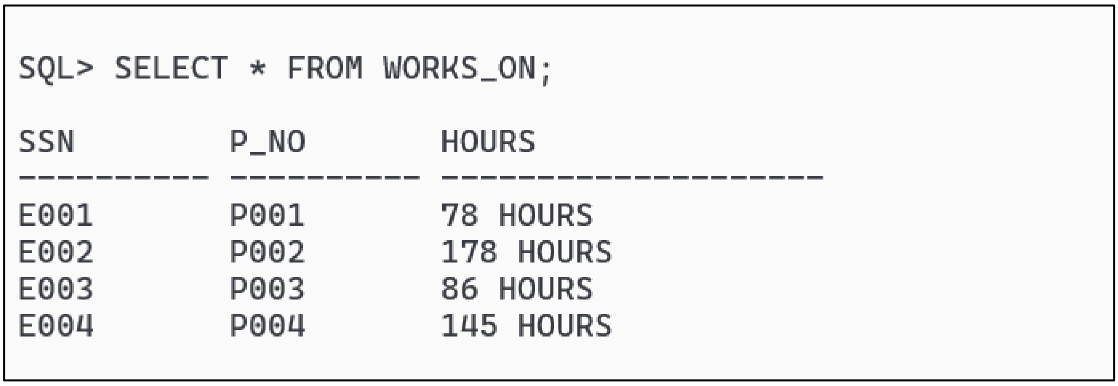
```
SQL> SELECT * FROM DLOCATION;

D_LOC      D_NO
---------- --------------------
PUNE       D001
DELHI      D002
MUMBAI     D003
HYDERABAD  D004
```

```
INSERT INTO PROJECT VALUES('P001','D001','IOT','LONDON');
INSERT INTO PROJECT VALUES('P002','D002','ML','PARIS');
INSERT INTO PROJECT VALUES('P003','D003','WEB DEV','LONDON');
INSERT INTO PROJECT VALUES('P004','D004','AI','NEW YORK');
```



```
SQL> SELECT * FROM PROJECT;

P_NO       D_NO                 P_NAME               P_LOCATION
---------- -------------------- -------------------- --------------------
P001       D001                 IOT                  LONDON
P002       D002                 ML                   PARIS
P003       D003                 WEB DEV              LONDON
P004       D004                 AI                   NEW YORK
```

```
INSERT INTO WORKS_ON VALUES('E001','P001','78 HOURS');
INSERT INTO WORKS_ON VALUES('E002','P002','178 HOURS');
INSERT INTO WORKS_ON VALUES('E003','P003','86 HOURS');
INSERT INTO WORKS_ON VALUES('E004','P004','145 HOURS');
```

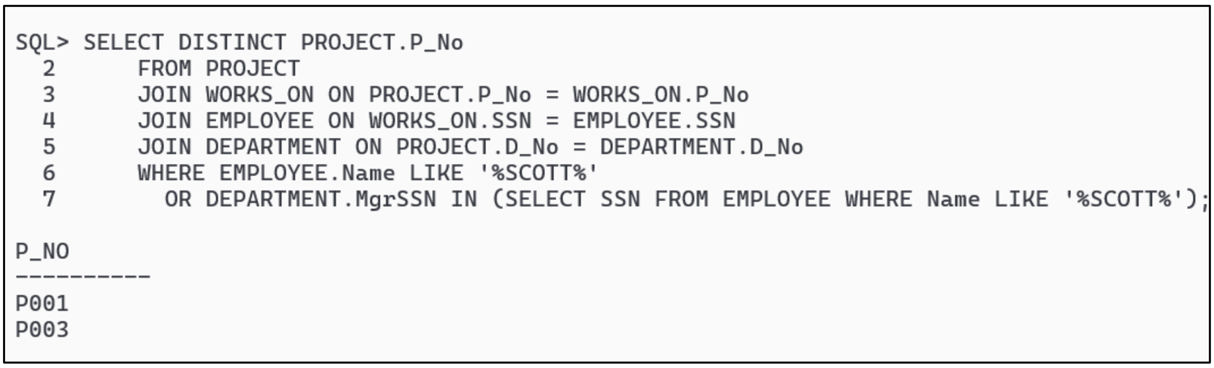


```
SQL> SELECT * FROM WORKS_ON;

SSN        P_NO       HOURS
---------- ---------- --------------------
E001       P001       78 HOURS
E002       P002       178 HOURS
E003       P003       86 HOURS
E004       P004       145 HOURS
```

Write SQL queries:

1. **Make a list of all project numbers for projects that involve an employee whose last name is ‘Scott’, either as a worker or as a manager of the department that controls the project.**

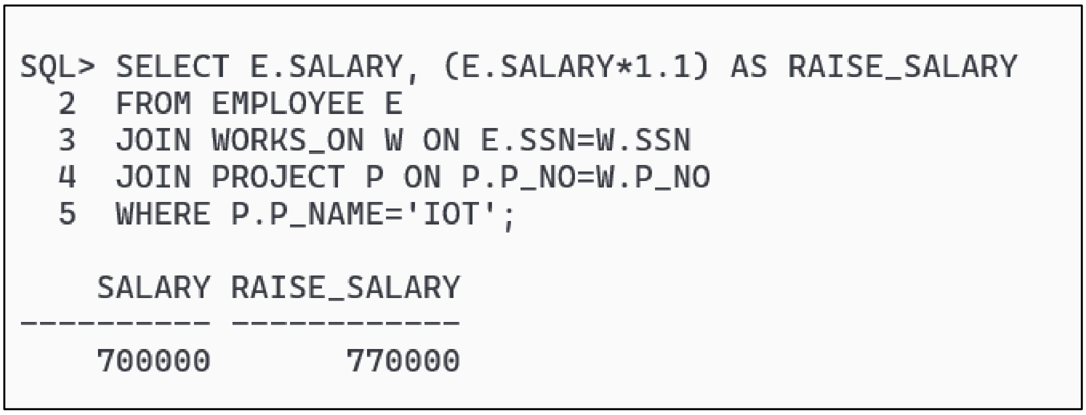
```
SELECT DISTINCT PROJECT.P_No
   FROM PROJECT
   JOIN WORKS_ON ON PROJECT.P_No = WORKS_ON.P_No
   JOIN EMPLOYEE ON WORKS_ON.SSN = EMPLOYEE.SSN
   JOIN DEPARTMENT ON PROJECT.D_No = DEPARTMENT.D_No
   WHERE EMPLOYEE.Name LIKE '%SCOTT%'
     OR DEPARTMENT.MgrSSN IN (SELECT SSN FROM EMPLOYEE WHERE Name LIKE
'%SCOTT%');
```



```
SQL> SELECT DISTINCT PROJECT.P_No
         FROM PROJECT
         JOIN WORKS_ON ON PROJECT.P_No = WORKS_ON.P_No
         JOIN EMPLOYEE ON WORKS_ON.SSN = EMPLOYEE.SSN
         JOIN DEPARTMENT ON PROJECT.D_No = DEPARTMENT.D_No
         WHERE EMPLOYEE.Name LIKE '%SCOTT%'
           OR DEPARTMENT.MgrSSN IN (SELECT SSN FROM EMPLOYEE WHERE Name LIKE '%SCOTT%');

P_NO
----------
P001
P003
```

2. **Show the resulting salaries if every employee working on the ‘IoT’ project is given a 10 percent raise.**

```
SELECT E.SALARY, (E.SALARY*1.1) AS RAISE_SALARY
FROM EMPLOYEE E
JOIN WORKS_ON W ON E.SSN=W.SSN
JOIN PROJECT P ON P.P_NO=W.P_NO
WHERE P.P_NAME='IOT';
```



```
SQL> SELECT E.SALARY, (E.SALARY*1.1) AS RAISE_SALARY
  2  FROM EMPLOYEE E
  3  JOIN WORKS_ON W ON E.SSN=W.SSN
  4  JOIN PROJECT P ON P.P_NO=W.P_NO
  5  WHERE P.P_NAME='IOT';

    SALARY RAISE_SALARY
---------- ------------
    700000       770000
```

3. **Find the sum of the salaries of all employees of the ‘Accounts’ department, as well as the maximum salary, the minimum salary, and the average salary in this department**

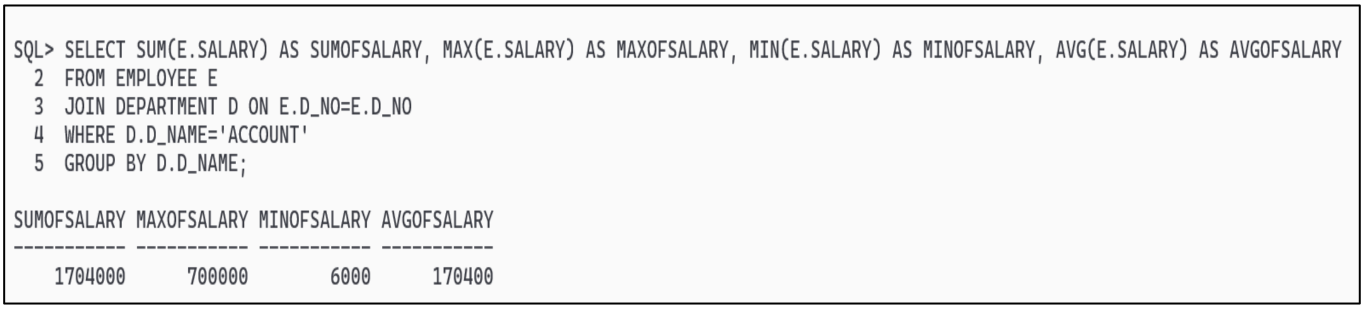
SELECT SUM(E.SALARY) AS SUMOFSALARY, MAX(E.SALARY) AS MAXOFSALARY, MIN(E.SALARY) AS MINOFSALARY, AVG(E.SALARY) AS AVGOFSALARY

FROM EMPLOYEE E

JOIN DEPARTMENT D ON E.D_NO=E.D_NO

WHERE D.D_NAME='ACCOUNT'

GROUP BY D.D_NAME;

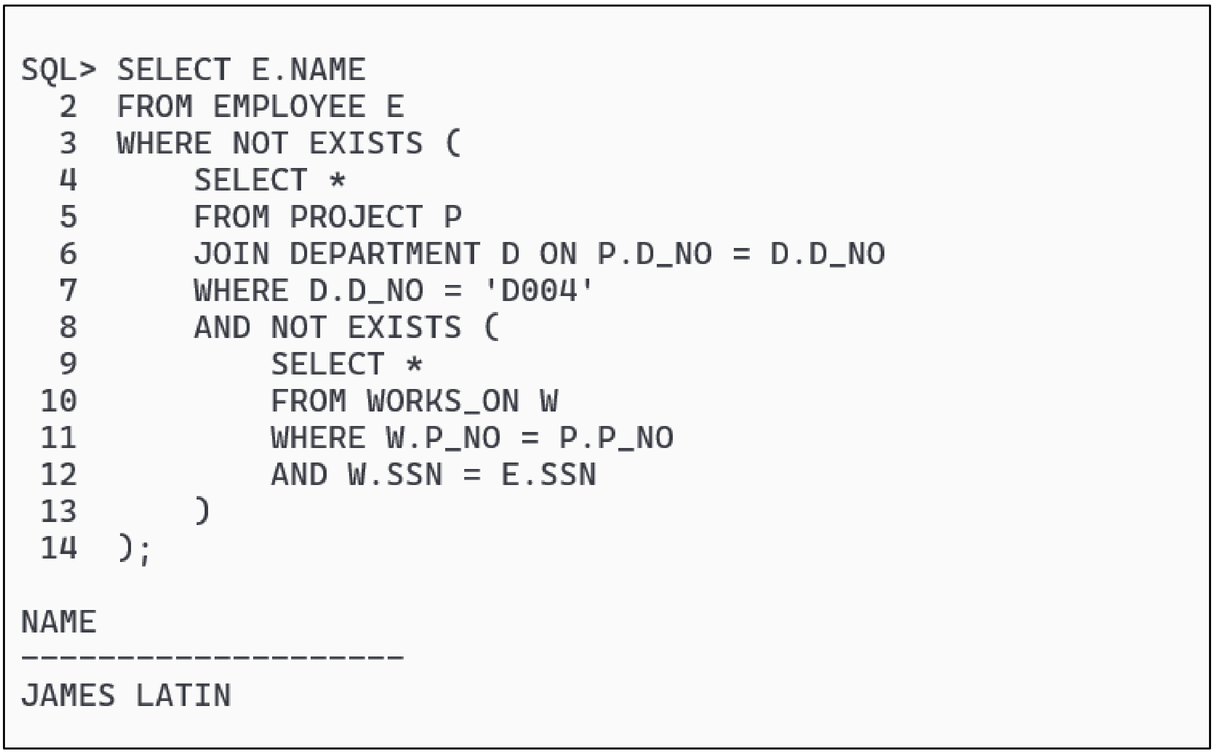


```
SQL> SELECT SUM(E.SALARY) AS SUMOFSALARY, MAX(E.SALARY) AS MAXOFSALARY, MIN(E.SALARY) AS MINOFSALARY, AVG(E.SALARY) AS AVGOFSALARY
  2  FROM EMPLOYEE E
  3  JOIN DEPARTMENT D ON E.D_NO=E.D_NO
  4  WHERE D.D_NAME='ACCOUNT'
  5  GROUP BY D.D_NAME;

SUMOFSALARY MAXOFSALARY MINOFSALARY AVGOFSALARY
----------- ----------- ----------- -----------
    1704000      700000        6000      170400
```

4. **Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator).**

```
SELECT E.NAME
FROM EMPLOYEE E
WHERE NOT EXISTS (SELECT *
   FROM PROJECT P
   JOIN DEPARTMENT D ON P.D_NO = D.D_NO
   WHERE D.D_NO = 'D004'
   AND NOT EXISTS ( SELECT *
          FROM WORKS_ON W
```

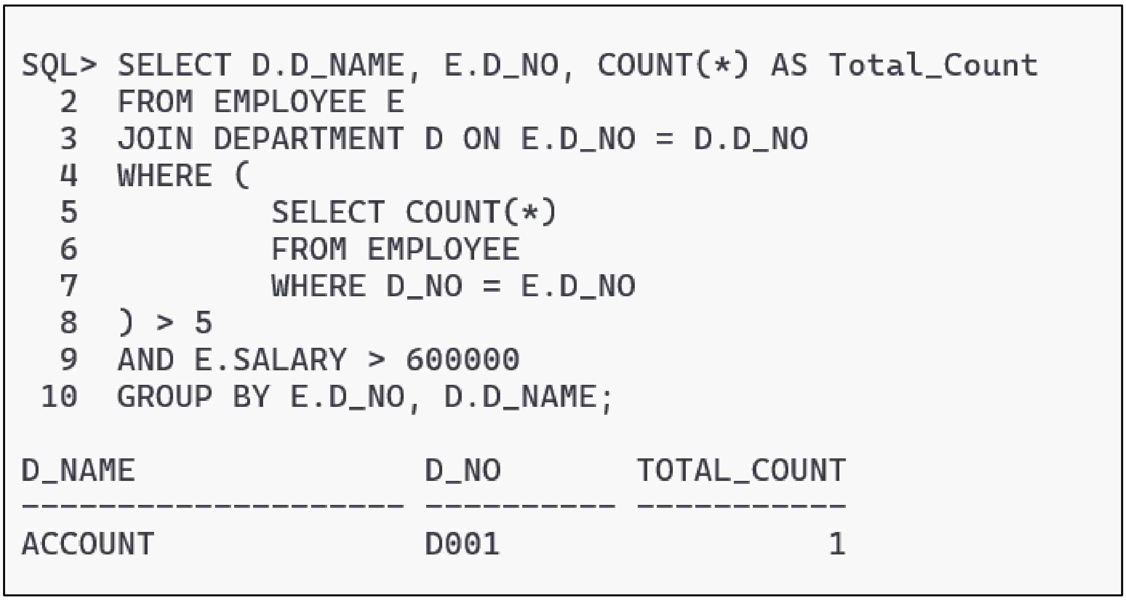
```
            WHERE W.P_NO = P.P_NO
            AND W.SSN = E.SSN
        )
    );
```



```
SQL> SELECT E.NAME
  2  FROM EMPLOYEE E
  3  WHERE NOT EXISTS (
  4      SELECT *
  5      FROM PROJECT P
  6      JOIN DEPARTMENT D ON P.D_NO = D.D_NO
  7      WHERE D.D_NO = 'D004'
  8      AND NOT EXISTS (
  9          SELECT *
 10          FROM WORKS_ON W
 11          WHERE W.P_NO = P.P_NO
 12          AND W.SSN = E.SSN
 13      )
 14  );

NAME
--------------------
JAMES LATIN
```

5. **For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.**

```
SELECT D.D_NAME, E.D_NO, COUNT(*) AS Total_Count
FROM EMPLOYEE E
JOIN DEPARTMENT D ON E.D_NO = D.D_NO
WHERE (
    SELECT COUNT(*)
    FROM EMPLOYEE
    WHERE D_NO = E.D_NO
) > 5
AND E.SALARY > 600000
GROUP BY E.D_NO, D.D_NAME;
```

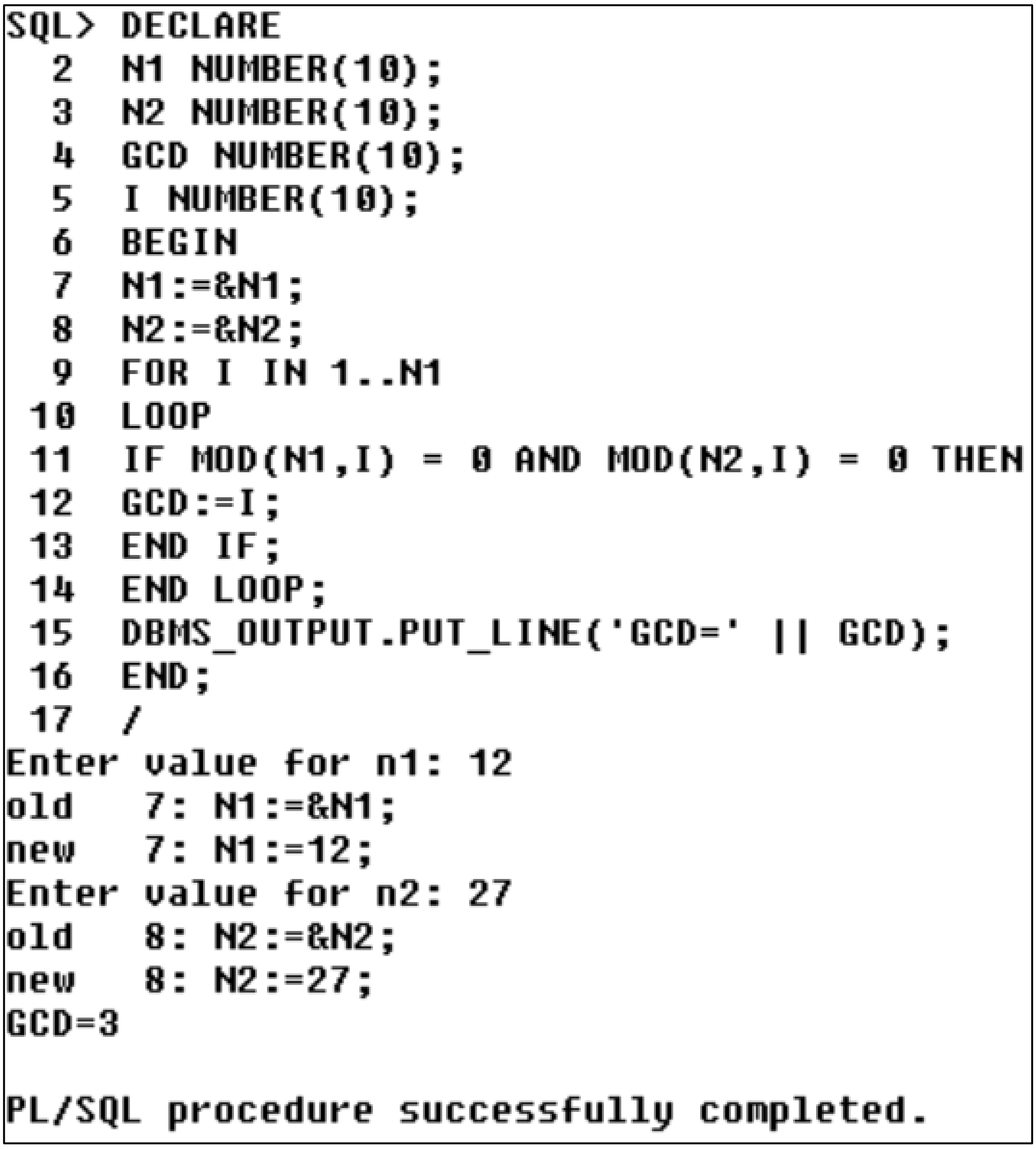


```
SQL> SELECT D.D_NAME, E.D_NO, COUNT(*) AS Total_Count
  2  FROM EMPLOYEE E
  3  JOIN DEPARTMENT D ON E.D_NO = D.D_NO
  4  WHERE (
  5          SELECT COUNT(*)
  6          FROM EMPLOYEE
  7          WHERE D_NO = E.D_NO
  8  ) > 5
  9  AND E.SALARY > 600000
 10  GROUP BY E.D_NO, D.D_NAME;

D_NAME                D_NO       TOTAL_COUNT
--------------------- ---------- -----------
ACCOUNT               D001                 1
```

PL/SQL queries:

B. Write a program in PL/SQL to create a procedure to displays the GCD of nos.

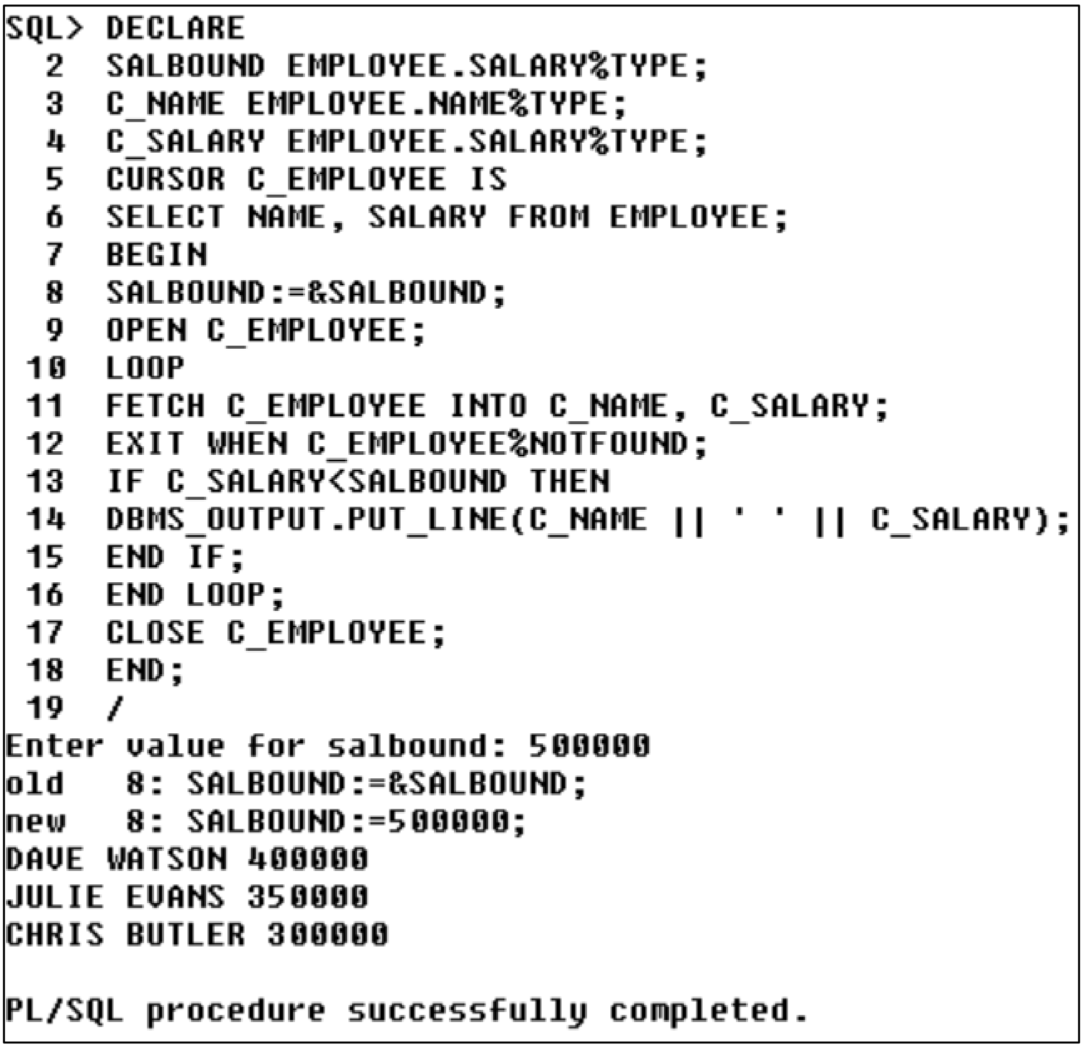
```
DECLARE
N1 NUMBER(10);
N2 NUMBER(10);
GCD NUMBER(10);
I NUMBER(10);
BEGIN
N1:=&N1;
N2:=&N2;
FOR I IN 1..N1
LOOP
IF MOD(N1,I) = 0 AND MOD(N2,I) = 0 THEN
GCD:=I;
END IF;
END LOOP;
DBMS_OUTPUT.PUT_LINE('GCD=' || GCD);
END;
```



```
SQL> DECLARE
  2  N1 NUMBER(10);
  3  N2 NUMBER(10);
  4  GCD NUMBER(10);
  5  I NUMBER(10);
  6  BEGIN
  7  N1:=&N1;
  8  N2:=&N2;
  9  FOR I IN 1..N1
 10  LOOP
 11  IF MOD(N1,I) = 0 AND MOD(N2,I) = 0 THEN
 12  GCD:=I;
 13  END IF;
 14  END LOOP;
 15  DBMS_OUTPUT.PUT_LINE('GCD=' || GCD);
 16  END;
 17  /
Enter value for n1: 12
old   7: N1:=&N1;
new   7: N1:=12;
Enter value for n2: 27
old   8: N2:=&N2;
new   8: N2:=27;
GCD=3

PL/SQL procedure successfully completed.
```

C. Write a program in PL/SQL to create a cursor displays the name and salary of each employee in the EMPLOYEES table whose salary is less than that specified by a passed-in parameter value.

```
DECLARE
SALBOUND EMPLOYEE.SALARY%TYPE;
C_NAME EMPLOYEE.NAME%TYPE;
C_SALARY EMPLOYEE.SALARY%TYPE;
CURSOR C_EMPLOYEE IS
SELECT NAME, SALARY FROM EMPLOYEE;
BEGIN
SALBOUND:=&SALBOUND;
OPEN C_EMPLOYEE;
LOOP
FETCH C_EMPLOYEE INTO C_NAME, C_SALARY;
EXIT WHEN C_EMPLOYEE%NOTFOUND;
IF C_SALARY<SALBOUND THEN
DBMS_OUTPUT.PUT_LINE(C_NAME || ' ' || C_SALARY);
END IF;
END LOOP;
CLOSE C_EMPLOYEE;
END;
/
```



```
SQL> DECLARE
  2  SALBOUND EMPLOYEE.SALARY%TYPE;
  3  C_NAME EMPLOYEE.NAME%TYPE;
  4  C_SALARY EMPLOYEE.SALARY%TYPE;
  5  CURSOR C_EMPLOYEE IS
  6  SELECT NAME, SALARY FROM EMPLOYEE;
  7  BEGIN
  8  SALBOUND:=&SALBOUND;
  9  OPEN C_EMPLOYEE;
 10  LOOP
 11  FETCH C_EMPLOYEE INTO C_NAME, C_SALARY;
 12  EXIT WHEN C_EMPLOYEE%NOTFOUND;
 13  IF C_SALARY<SALBOUND THEN
 14  DBMS_OUTPUT.PUT_LINE(C_NAME || ' ' || C_SALARY);
 15  END IF;
 16  END LOOP;
 17  CLOSE C_EMPLOYEE;
 18  END;
 19  /
Enter value for salbound: 500000
old   8: SALBOUND:=&SALBOUND;
new   8: SALBOUND:=500000;
DAVE WATSON 400000
JULIE EVANS 350000
CHRIS BUTLER 300000

PL/SQL procedure successfully completed.
```