Global Academy of Technology
Growing Ahead Of Time....
Autonomous Institute, Affiliated To VTU

Department of Artificial Intelligence & Machine Learning

AML23403

Database Management System

VASUGI I

ASSISTANT PROFESSOR

DEPARTMENT OF AI & ML

Exercise-5

Consider the schema for Company Database:

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

DNo) DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate)

DLOCATION (DNo,DLoc)

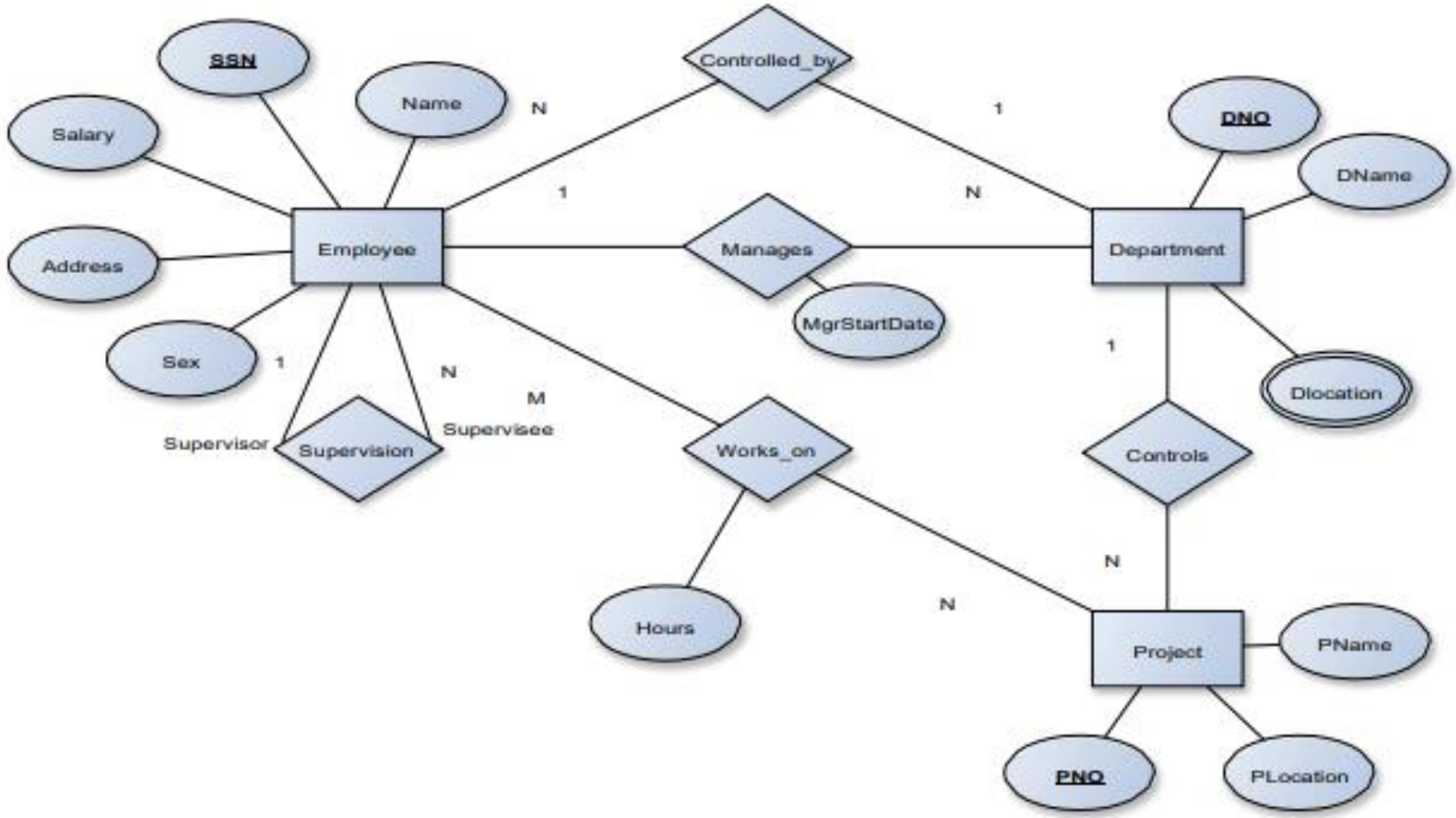
PROJECT (PNo, PName, PLocation,

DNo) WORKS_ON (SSN, PNo, Hours)

Write SQL queries to

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.
2. Show the resulting salaries if every employee working on the ‘IoT’ project is given a 10 percent raise.
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
4. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator).
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.

Entity-Relationship Diagram



Schema Diagram

Table Creation

1. CREATE TABLE DEPARTMENT (DNO VARCHAR2 (20) PRIMARY KEY, DNAME VARCHAR2 (20), MGRSTARTDATE DATE);
2. CREATE TABLE EMPLOYEE (SSN VARCHAR2 (20) PRIMARY KEY, FNAME VARCHAR2 (20), LNAME VARCHAR2 (20), ADDRESS VARCHAR2 (20), SEX CHAR (1), SALARY INTEGER, SUPERSSN REFERENCES EMPLOYEE (SSN), DNO REFERENCES DEPARTMENT (DNO));
3. ALTER TABLE DEPARTMENT ADD MGRSSN REFERENCES EMPLOYEE (SSN);
4. CREATE TABLE DLOCATION (DLOC VARCHAR2 (20), DNO REFERENCES DEPARTMENT (DNO), PRIMARY KEY (DNO, DLOC));

Table Creation

5. CREATE TABLE PROJECT (PNO INTEGER PRIMARY KEY, PNAME VARCHAR2 (20), PLOCATION VARCHAR2 (20), DNO REFERENCES DEPARTMENT (DNO));

6. CREATE TABLE WORKS_ON (HOURS INTEGER, SSN REFERENCES EMPLOYEE (SSN), PNO REFERENCES PROJECT(PNO), PRIMARY KEY (SSN, PNO));

Table Description

- DESC DEPARTMENT;
- DESC EMPLOYEE;
- DESC DLOCATION;
- DESC PROJECT;
- DESC WORKS_ON;

- Insertion of Values to Tables –
  DEPARTMENT;
  EMPLOYEE;
  DLOCATION;
  PROJECT;
  WORKS_ON;

Query-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 P.PNO
FROM PROJECT P, DEPARTMENT D, EMPLOYEE E
WHERE E.DNO=D.DNO
    AND D.MGRSSN=E.SSN
   AND E.LNAME=‘SCOTT’)
UNION
(SELECT DISTINCT P1.PNO
 FROM PROJECT P1, WORKS_ON W, EMPLOYEE E1
 WHERE P1.PNO=W.PNO
    AND E1.SSN=W.SSN
    AND E1.LNAME=‘SCOTT’);
```

Query-2

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

```
SELECT E.FNAME, E.LNAME, 1.1*E.SALARY AS INCR_SAL
FROM EMPLOYEE E, WORKS_ON W, PROJECT P
WHERE E.SSN=W.SSN
    AND W.PNO=P.PNO
   AND P.PNAME=‘IOT’;
```

Query-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),
            MAX (E.SALARY),
            MIN (E.SALARY),
           AVG (E.SALARY)
FROM EMPLOYEE E, DEPARTMENT D
WHERE E.DNO=D.DNO
     AND D.DNAME=‘ACCOUNTS’;
```

Query-4

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

```
SELECT E.FNAME, E.LNAME
FROM EMPLOYEE E
WHERE NOT EXISTS(
        (SELECT PNO
         FROM PROJECT
         WHERE DNO='5')
        MINUS
       (SELECT PNO
        FROM WORKS_ON
        WHERE E.SSN=SSN)
 );
```

Query-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.DNO, COUNT (*)
FROM DEPARTMENT D, EMPLOYEE E
WHERE D.DNO=E.DNO
    AND E.SALARY>600000
   AND D.DNO IN
                (SELECT E1.DNO
                 FROM EMPLOYEE E1 GROUP BY E1.DNO
                                          HAVING COUNT (*)>5)
 GROUP BY D.DNO;
```