❖EXERCISE 1:

1) Create the table SEMP with the following structure:-
   EMPNO CHAR(4)
   EMPNAME CHAR(20)
   BASIC FLOAT
   DEPTNO CHAR(2)
   DEPTHEAD CHAR(4)

⇒ CREATE TABLE SEMP (

EMPNO CHAR(4) PRIMARY KEY,

EMPNAME CHAR(20),

BASIC FLOAT,

DEPTNO CHAR(2),

DEPTHEAD CHAR(4)

);

2) Create the table SDEPT with the following structure:-
   DEPTNO CHAR(2)
   DEPTNAME CHAR(15)

⇒ CREATE TABLE SDEPT (

DEPTNO CHAR(2) PRIMARY KEY,

DEPTNAME CHAR(15)

);

3) Insert into the SDEPT table the following values:-
   10, Development
   20, Training

⇒ INSERT INTO SDEPT (DEPTNO, DEPTNAME) VALUES ('10', 'Development');

INSERT INTO SDEPT (DEPTNO, DEPTNAME) VALUES ('20', 'Training');

4) Insert into the SEMP table the following values:-
0001, SUNIL, 6000, 10
0002, HIREN, 8000, 20
0003, ALI, 4000, 10, 0001
0004, GEORGE, 6000, 0002

⇒ INSERT INTO SEMP (EMPNO, EMPNAME, BASIC, DEPTNO, DEPTHEAD) VALUES ('0001', 'SUNIL', 6000, '10', NULL);

INSERT INTO SEMP (EMPNO, EMPNAME, BASIC, DEPTNO, DEPTHEAD) VALUES ('0002', 'HIREN', 8000, '20', NULL);

INSERT INTO SEMP (EMPNO, EMPNAME, BASIC, DEPTNO, DEPTHEAD) VALUES ('0003', 'ALI', 4000, '10', '0001');

INSERT INTO SEMP (EMPNO, EMPNAME, BASIC, DEPTNO, DEPTHEAD) VALUES ('0004', 'GEORGE', 6000, '20', '0002');

5) Create S, P, J, SPJ tables as specified below and insert a few rows in each table:-
SUPPLIER (S#, Sname, Status, City) –
S PARTS (P#, Pname, Color, Weight, City) –
P PROJECTS (J#, Jname, City) –
J SUPPLIER-
PARTS-
PROJECT (S#, P#, J#, Qty) –
SPJ Sample data for
S# column:- ‘S1’, ‘S2’, ‘S3’, etc.
Sample data for P# column:- ‘P1’, ‘P2’, ‘P3’, etc.
Sample data for J# column:- ‘J1’, ‘J2’, ‘J3’, etc.
Sample data for Status column:- 10, 20, 30, etc.

⇒ CREATE TABLE S (

`S#` CHAR(2) PRIMARY KEY,

```
    `Sname` CHAR(20),
    `Status` INT,
    `City` CHAR(20)
);
CREATE TABLE `P` ( `P#` CHAR(2) PRIMARY KEY,
`Pname` CHAR(20), `Color` CHAR(20), `Weight` FLOAT,
`City` CHAR(20) );
CREATE TABLE `J` ( `J#` CHAR(2) PRIMARY KEY,
`Jname` CHAR(20), `City` CHAR(20) );
CREATE TABLE `SPJ` ( `S#` CHAR(2), `P#` CHAR(2),
`J#` CHAR(2), `Qty` INT, PRIMARY KEY (`S#`, `P#`,
`J#`), FOREIGN KEY (`S#`) REFERENCES `S`(`S#`),
FOREIGN KEY (`P#`) REFERENCES `P`(`P#`), FOREIGN
KEY (`J#`) REFERENCES `J`(`J#`) );
INSERT INTO `S` (`S#`, `Sname`, `Status`, `City`) VALUES
('S1', 'Supplier A', 10, 'New York'),
('S2', 'Supplier B', 20, 'Los Angeles'),
('S3', 'Supplier C', 30, 'Chicago');
INSERT INTO `P` (`P#`, `Pname`, `Color`, `Weight`, `City`)
VALUES
('P1', 'Part X', 'Red', 10.5, 'New York'),
('P2', 'Part Y', 'Blue', 15.0, 'Los Angeles'),
('P3', 'Part Z', 'Green', 7.8, 'Chicago');
INSERT INTO `J` (`J#`, `Jname`, `City`) VALUES
('J1', 'Project Alpha', 'New York'),
('J2', 'Project Beta', 'Los Angeles'),
('J3', 'Project Gamma', 'Chicago');
```

```
INSERT INTO `SPJ` (`S#`, `P#`, `J#`, `Qty`) VALUES
('S1', 'P1', 'J1', 100),
('S2', 'P2', 'J2', 200),
('S3', 'P3', 'J3', 150),
('S1', 'P2', 'J3', 50);
```

6) Display all the data from the S table.

```
⇒ SELECT * FROM `S`;
```

7) Display only the S# and SNAME fields from the S table.

```
⇒SELECT `S#`, `Sname` FROM `S`;
```

8) Display the PNAME and COLOR from the P table for the CITY=”London”.

```
⇒ SELECT `Pname`, `Color` FROM `P` WHERE `City` =
  'London';
```

9) Display all the Suppliers from London.

```
⇒ SELECT * FROM `S` WHERE `City` = 'London';
```

10) Display all the Suppliers from Paris or Athens.

```
⇒ SELECT * FROM `S` WHERE `City` IN ('Paris',
  'Athens');
```

11) Display all the Projects in Athens.

```
⇒ SELECT * FROM `J` WHERE `City` = 'Athens';
```

12) Display all the Partnames with the weight between 12 and 14 (inclusive of both).

```
⇒  SELECT `Pname` FROM `P` WHERE `Weight`
  BETWEEN 12 AND 14;
```

13) Display all the Suppliers with a Status greater than or equal to 20.

⇒ SELECT * FROM `S` WHERE `Status` >= 20;

14) Display all the Suppliers except the Suppliers from London.

⇒ SELECT * FROM `S` WHERE `City` NOT IN ('London');

15) Display only the Cities from where the Suppliers come from.

⇒ SELECT DISTINCT `City` FROM `S`;

16) Assuming that the Part Weight is in GRAMS, display the same in MILLIGRAMS and KILOGRAMS.

⇒ SELECT `Pname`, `Weight` AS `Weight_in_grams`,

`Weight` * 1000 AS `Weight_in_milligrams`,

`Weight` / 1000 AS `Weight_in_kilograms` FROM `P`;