**SQL 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)**

**2. Create the table SDEPT with the following structure:-**

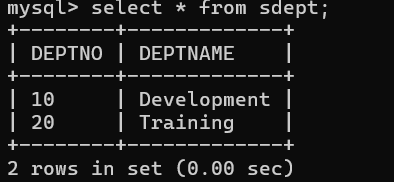
**DEPTNO CHAR(2)**

**DEPTNAME CHAR(15)**

**3. Insert into the SDEPT table the following values:-**

**10, Development**

**20, Training**

****

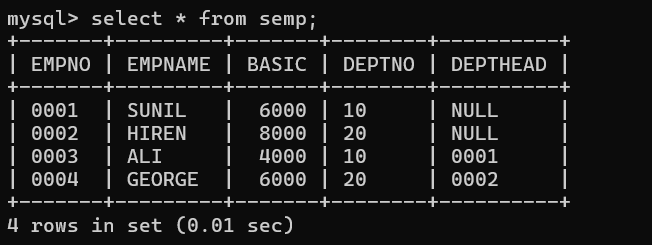
**4. Insert into the SEMP table the following values:-**

**0001, SUNIL, 6000, 10**

**0002, HIREN, 8000, 20**

**0003, ALI, 4000, 10, 0001**

**0004, GEORGE, 6000, 0002**

****

**Create S, P, J, SPJ tables as specified below and insert a few rows in each table:-**

**SUPPLIER - S**

**(S#, Sname, Status, City)**

**PARTS - P**

**(P#, Pname, Color, Weight, City)**

**PROJECTS - J**

**(J#, Jname, City)**

**SUPPLIER-PARTS-PROJECT - SPJ**

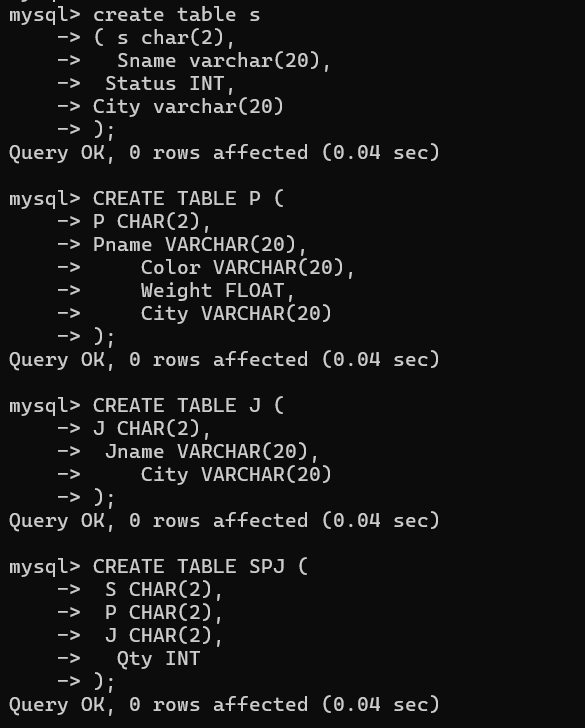
**(S#, P#, J#, Qty)**

**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.**

****

**-- Insert data into the SUPPLIER table**

**INSERT INTO S VALUES ('S1', 'Supplier1', 10, 'London');**

**INSERT INTO S VALUES ('S2', 'Supplier2', 20, 'Paris');**

**INSERT INTO S VALUES ('S3', 'Supplier3', 30, 'Athens');**

**INSERT INTO S VALUES ('S4', 'Supplier4', 20, 'New York');**

**-- Insert data into the PARTS table**

**INSERT INTO P VALUES ('P1', 'Part1', 'Red', 12.5, 'London');**

**INSERT INTO P VALUES ('P2', 'Part2', 'Blue', 15.2, 'Paris');**

**INSERT INTO P VALUES ('P3', 'Part3', 'Green', 13.8, 'London');**

**INSERT INTO P VALUES ('P4', 'Part4', 'Yellow', 9.5, 'Athens');**

**-- Insert data into the PROJECTS table**

**INSERT INTO J VALUES ('J1', 'Project1', 'London');**

**INSERT INTO J VALUES ('J2', 'Project2', 'Athens');**

**INSERT INTO J VALUES ('J3', 'Project3', 'Paris');**

**-- Insert data into the SUPPLIER-PARTS-PROJECT table**

**INSERT INTO SPJ VALUES ('S1', 'P1', 'J1', 100);**

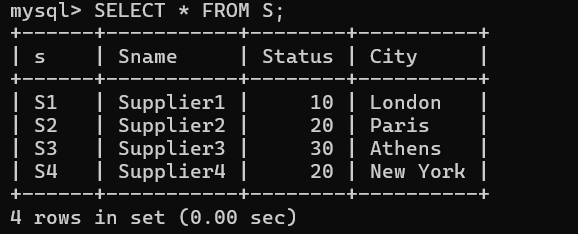
**INSERT INTO SPJ VALUES ('S2', 'P2', 'J3', 150);**

**INSERT INTO SPJ VALUES ('S3', 'P3', 'J2', 200);**

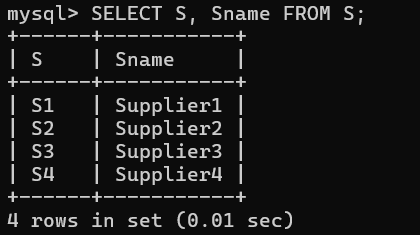
**INSERT INTO SPJ VALUES ('S4', 'P4', 'J2', 300);**

**Write the SELECT queries to do the following:-**

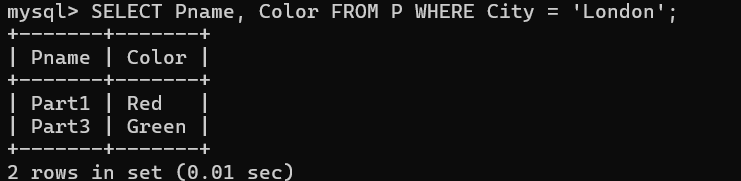
**5. Display all the data from the S table.**

****

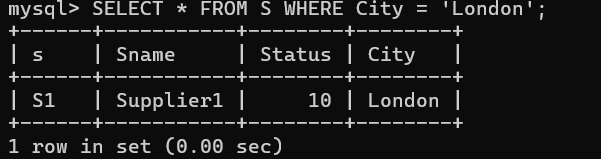
**6. Display only the S# and SNAME fields from the S table.**

****

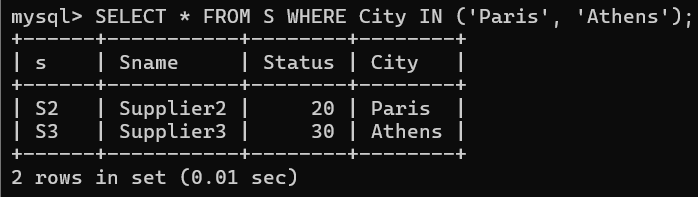
**7. Display the PNAME and COLOR from the P table for the CITY=”London”.**

****

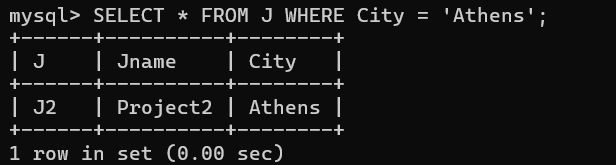
**8. Display all the Suppliers from London.**

****

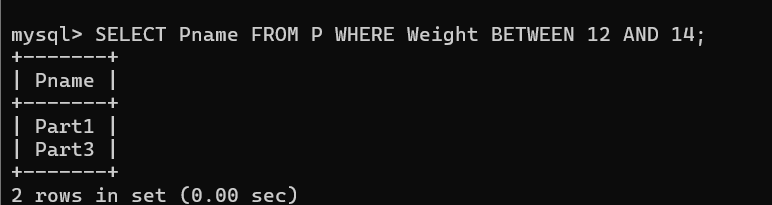
**9. Display all the Suppliers from Paris or Athens.**

****

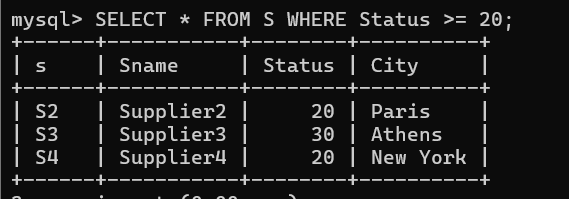
**10. Display all the Projects in Athens.**

****

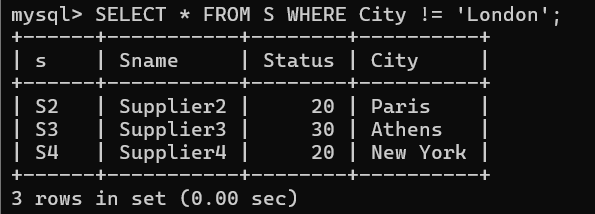
**11. Display all the Partnames with the weight between 12 and 14 (inclusive of both).**

****

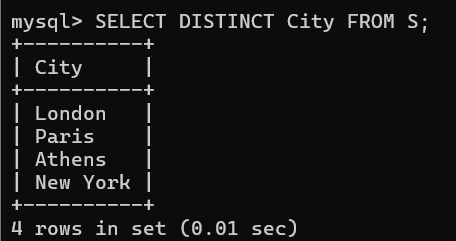
**12. Display all the Suppliers with a Status greater than or equal to 20.**

****

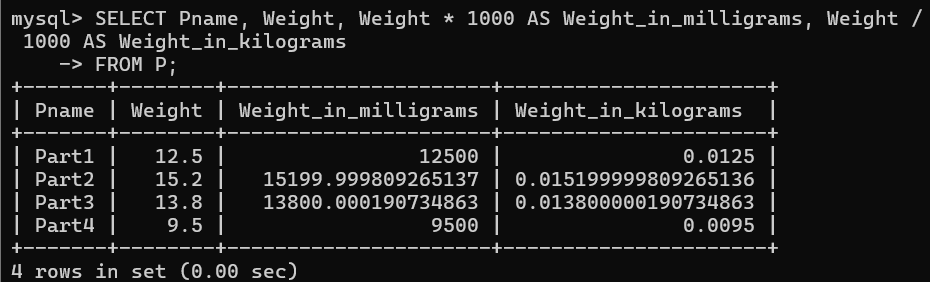
**13. Display all the Suppliers except the Suppliers from London.**

****

**14. Display only the Cities from where the Suppliers come from.**

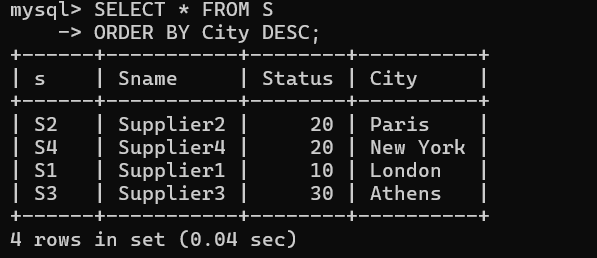
****

**15. Assuming that the Part Weight is in GRAMS, display the same in MILLIGRAMS and KILOGRAMS.**

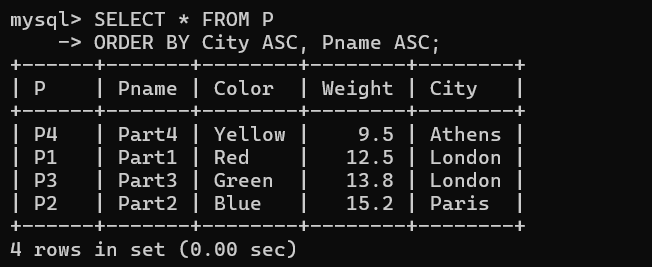
****

**SQL Exercise 2**

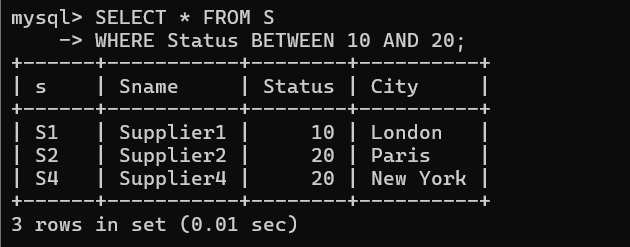
**1. Display the Supplier table in the descending order of CITY.**

****

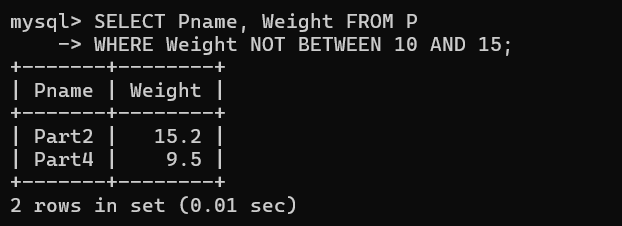
**2. Display the Part Table in the ascending order of CITY and within the city in the ascending order of Part names.**

****

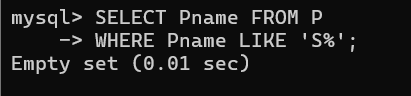
**3. Display all the Suppliers with a status between 10 and 20.**

****

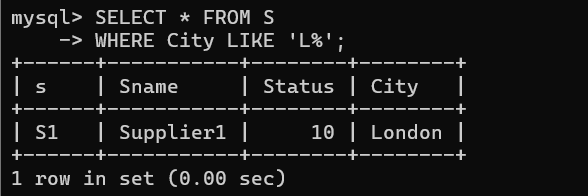
**4. Display all the Parts and their Weight, which are not in the range of 10 and 15.**

****

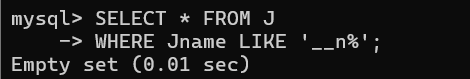
**5. Display all the Part names starting with the letter ‘S’.**

****

**6. Display all the Suppliers, belonging to cities starting with the letter ‘L’.**

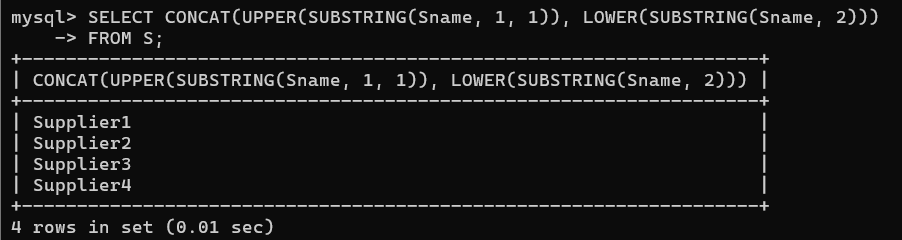
****

**7. Display all the Projects, with the third letter in JNAME as ‘n’.**

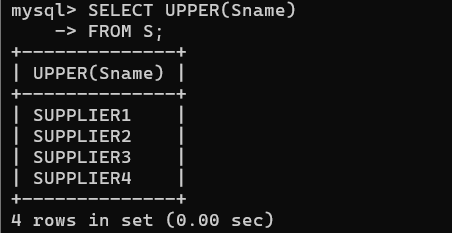
****

**SQL Exercise 3**

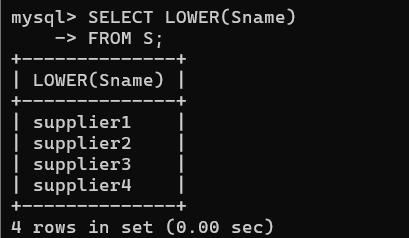
**1. Display all the Supplier names with the initial letter capital.**

****

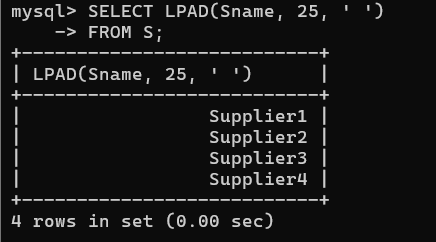
**2. Display all the Supplier names in upper case.**

****

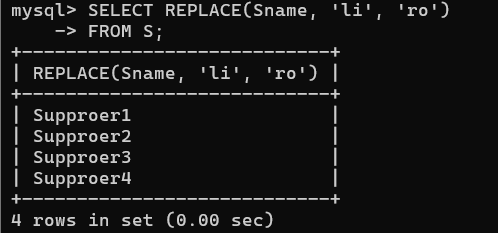
**3. Display all the Supplier names in lower case.**

****

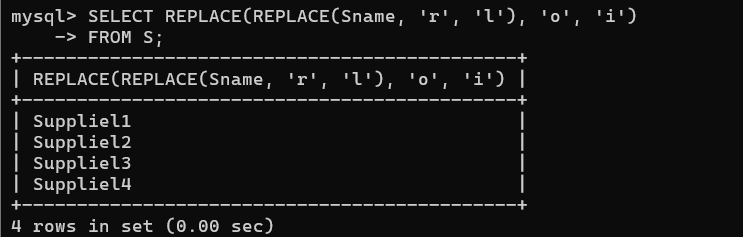
**4. Display all the Supplier names padded to 25 characters, with spaces on the left.**

****

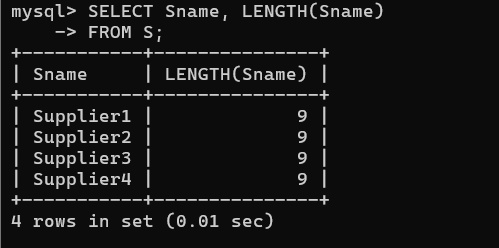
**5. Display all the Supplier names (with ‘la’ replaced by ‘ro’). HINT: REPLACE.**

****

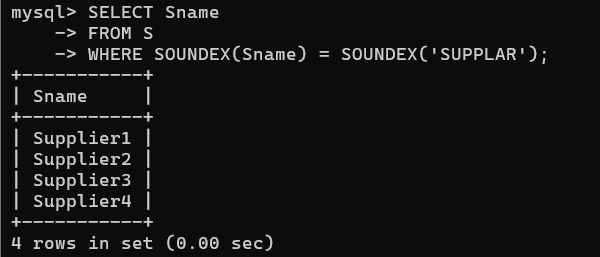
**6. Implement the above command such that ‘l’ is replaced with ‘r’ and ‘a’ is replaced with ‘o’.**

****

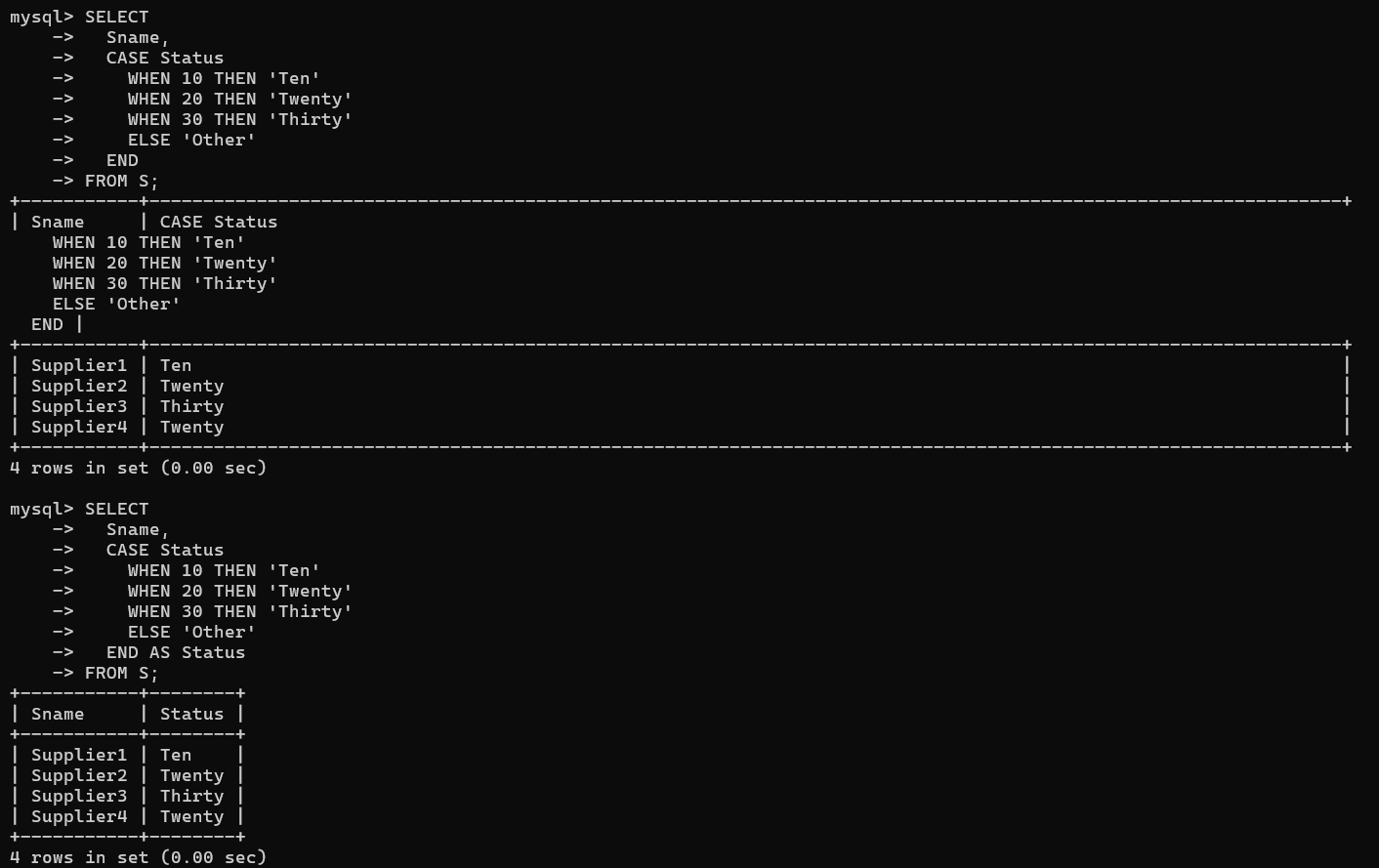
**7. Display the Supplier names and the lengths of the names.**

****

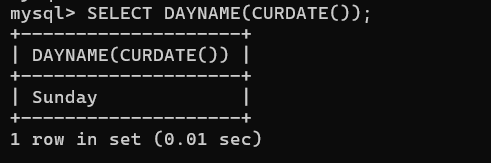
**8. Use the soundex function to search for a supplier by the name of ‘BLOKE’.**

****

**9. Display the Supplier name and the status (as Ten, Twenty, Thirty, etc.).**

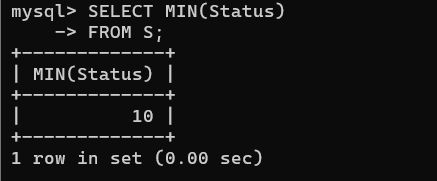
****

**10. Display the current day (e.g. Thursday).**

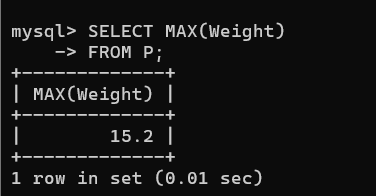
****

**SQL Exercise 4**

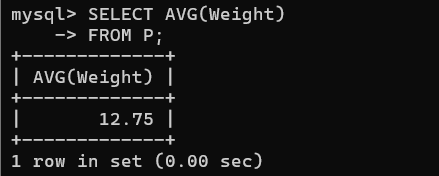
**1. Display the minimum Status in the Supplier table.**

****

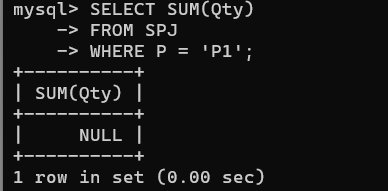
**2. Display the maximum Weight in the Parts table.**

****

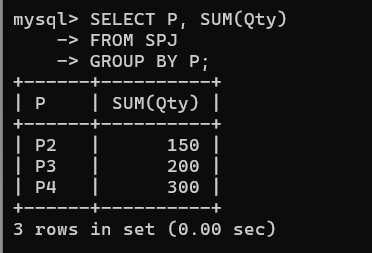
**3. Display the average Weight of the Parts.**

****

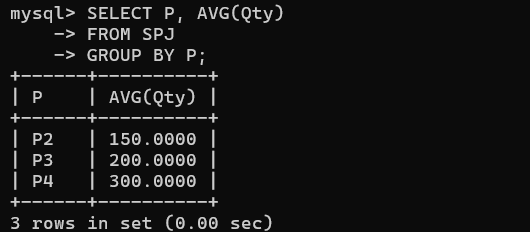
**4. Display the total Quantity sold for part ‘P1’.**

****

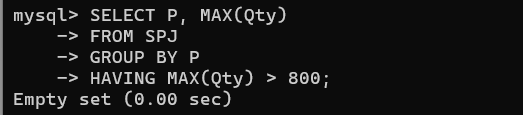
**5. Display the total Quantity sold for each part.**

****

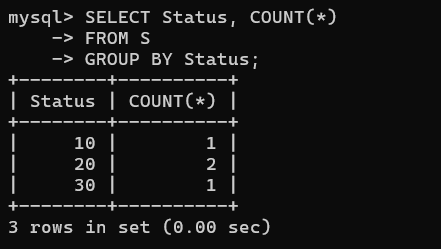
**6. Display the average Quantity sold for each part.**

****

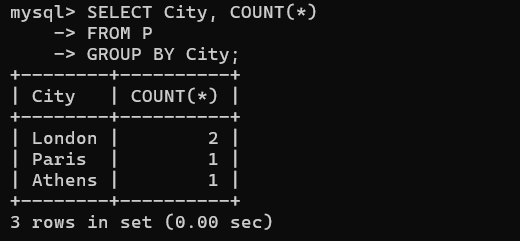
**7. Display the maximum Quantity sold for each part, provided the maximum Quantity is greater than 800.**

****

**8. Display the Status and the count of Suppliers with that Status.**

****

**9. Display the count of Projects going on in different cities.**

****

**10. What is the difference between COUNT(Status) and COUNT(\*) ?**

✅ SELECT COUNT(Status) FROM table;

→ Returns 3 (it skips the NULL in row 3)

✅ SELECT COUNT(\*) FROM table;

→ Returns 4 (counts all rows)

****

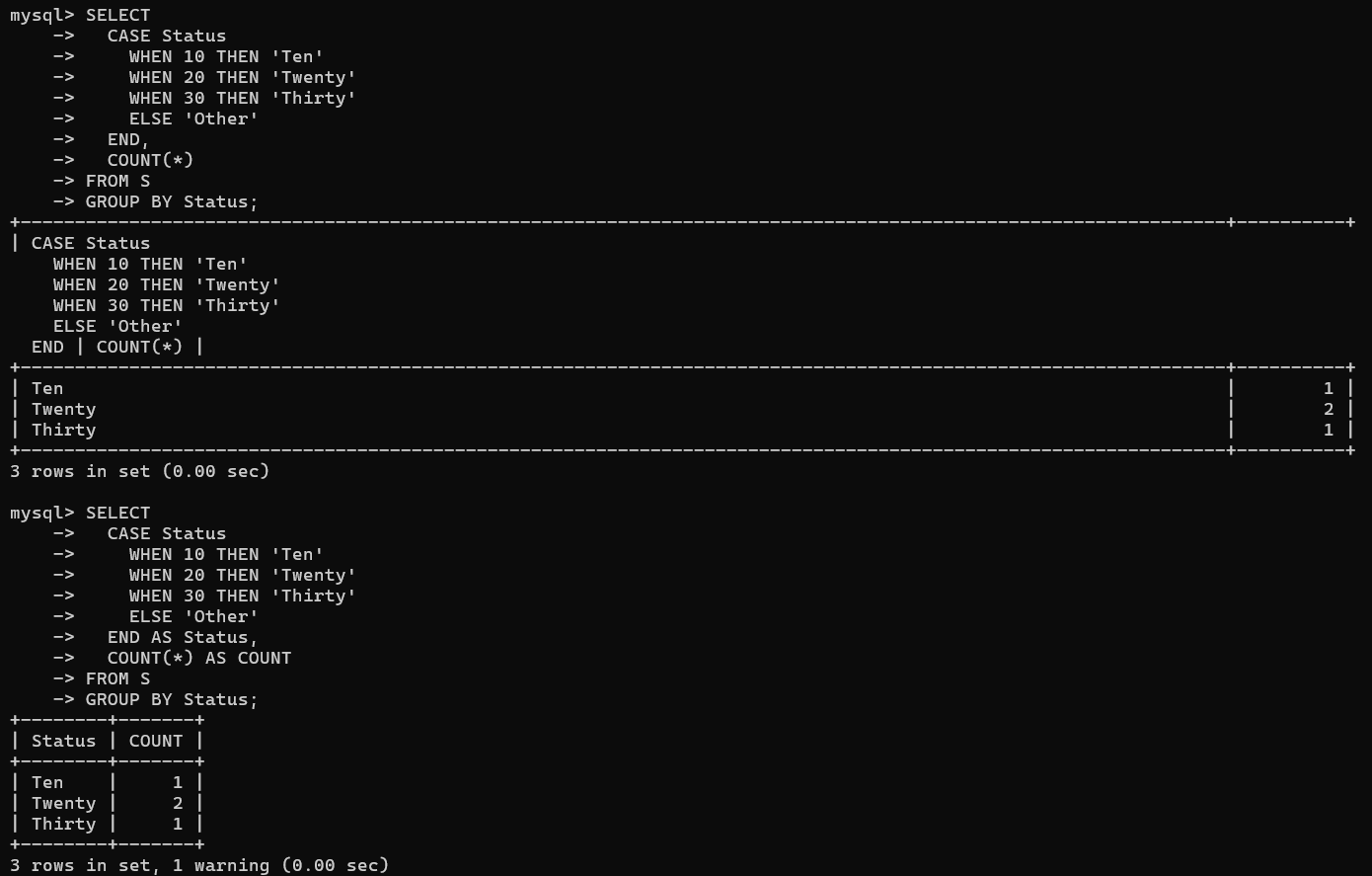
**11. Display the Status and the Count of Suppliers with that Status in the following format as shown below:-**

**Status Count**

**Ten 1**

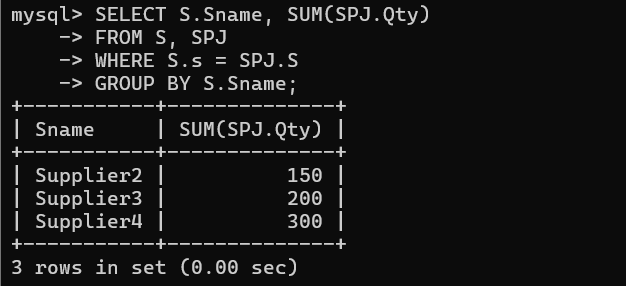
**Twenty 2**

**Thirty 3**

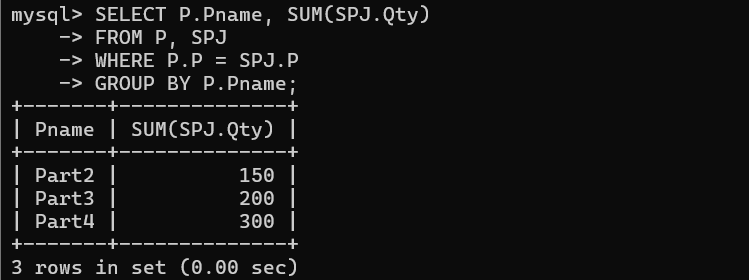
****

**SQL Exercise 5**

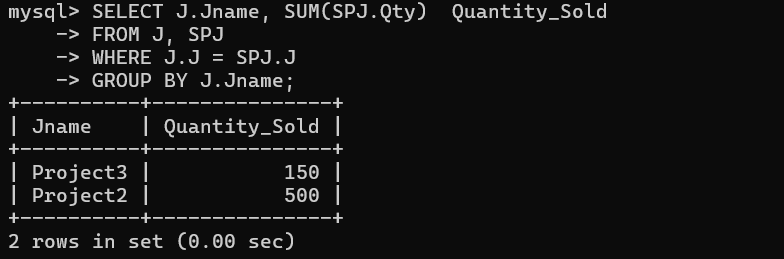
**1. Display the Supplier name and the Quantity sold.**

****

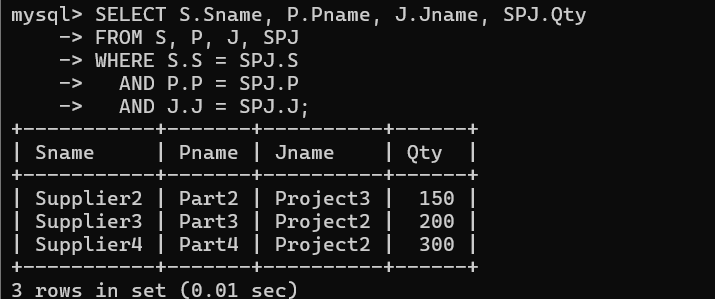
**2. Display the Part name and Quantity sold.**

****

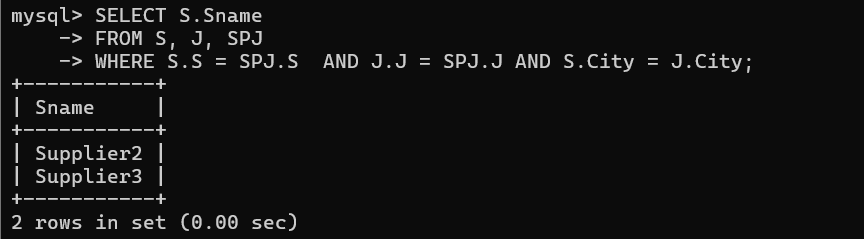
**3. Display the Project name and Quantity sold.**

****

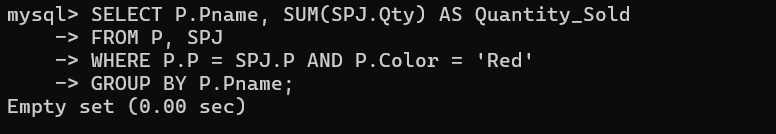
**4. Display the Supplier name, Part name, Project name and Quantity sold.**

****

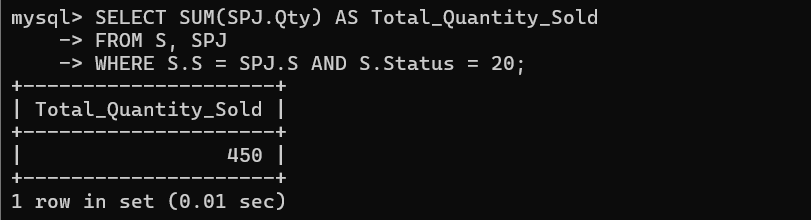
**5. Display the Supplier name, Supplying Parts to a Project in the same City.**

****

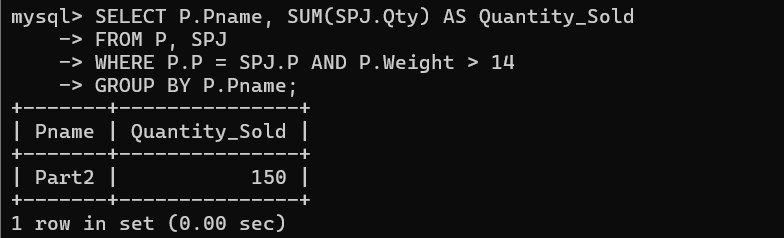
**6. Display the Part name that is ‘Red’ is color, and the Quantity sold.**

****

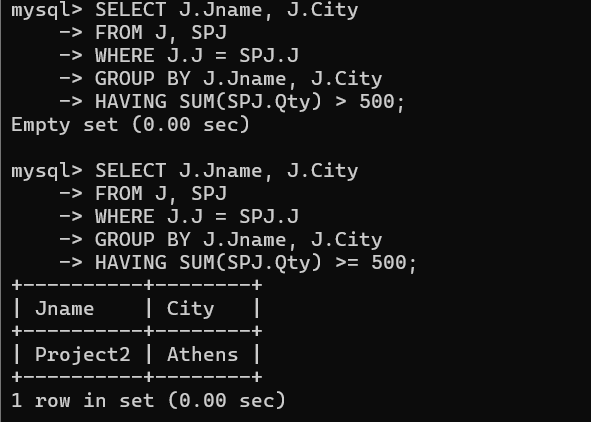
**7. Display all the Quantity sold by Suppliers with the Status = 20.**

****

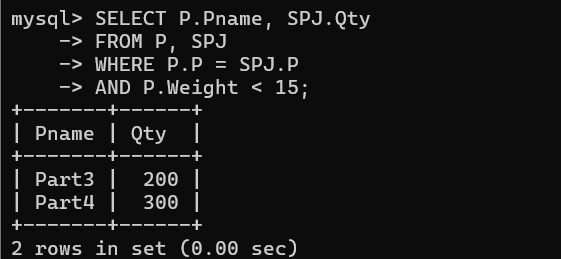
**8. Display all the Parts and Quantity with a Weight > 14.**

****

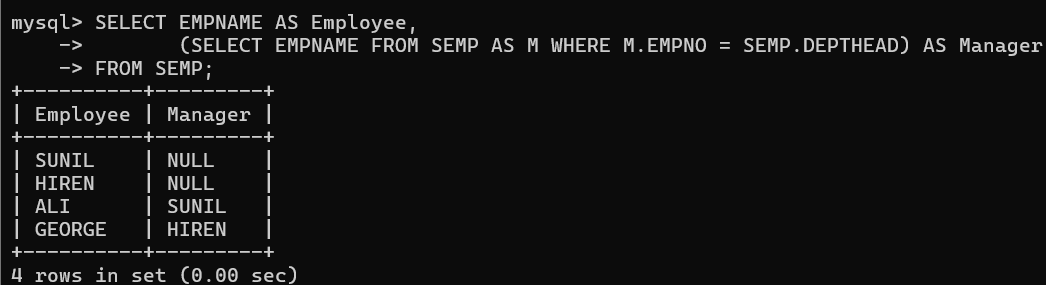
**9. Display all the Project names and City, which has bought more than 500 Parts.**

****

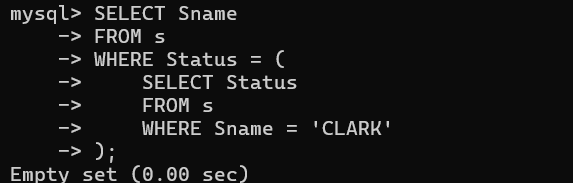
**10. Display all the Part names and Quantity sold that have a Weight less than 15.**

****

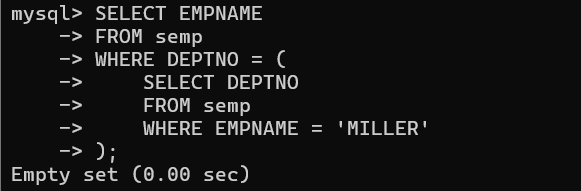
**11. Display all the Employee names and the name of their Managers.**

****

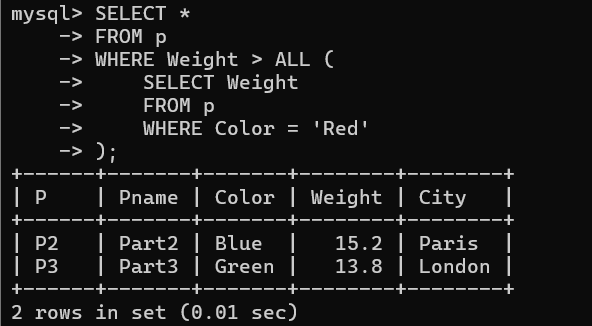
**SQL Exercise 6**

**1. Display all the Suppliers with the same Status as the supplier, ‘CLARK’. **

**2. Display all the Employees in the same department as the employee ‘MILLER’.**

****

**3. Display all the Parts which have more Weight than all the Red parts.**

****

**4. Display all the Projects going on in the same city as the project ‘TAPE’.**

SELECT \*

FROM Projects

WHERE city = (

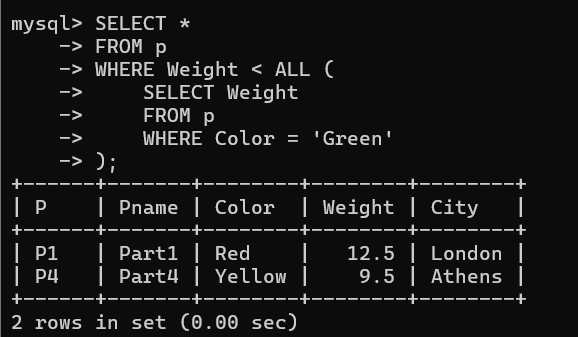
SELECT city

FROM Projects

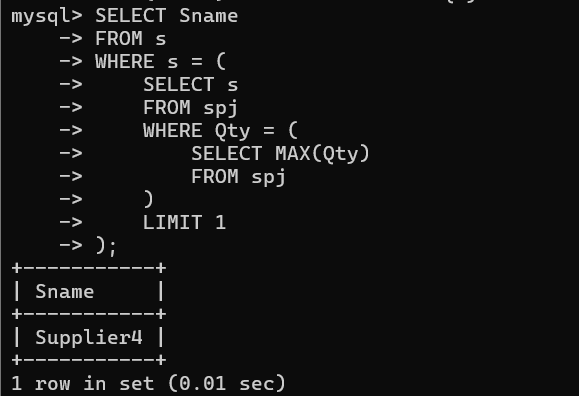
WHERE project\_name = 'TAPE'

);

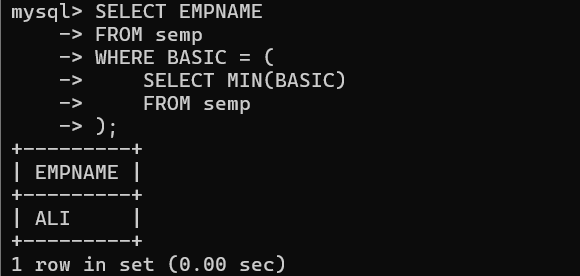
**5. Display all the Parts with Weight less than all the Green parts.**

****

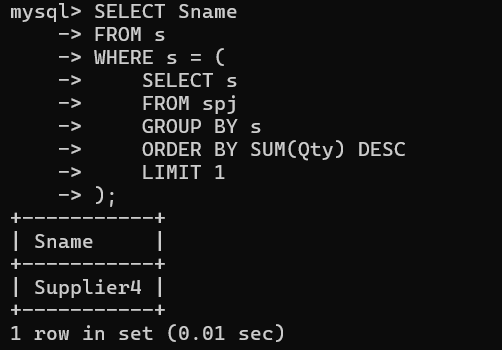
**6. Display the name of the Supplier who has sold the maximum Quantity (in one sale).**

****

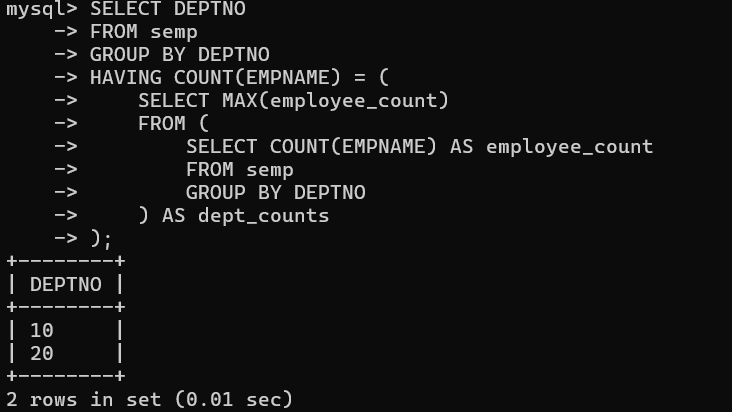
**7. Display the name of the Employee with the minimum Salary.**

****

**8. Display the name of the Supplier who has sold the maximum overall Quantity (sum of Sales).**

****

**9. Display the name of the Department with the maximum number of Employees.**

****