**SQL Exercise 4**

1. Display the minimum Status in the Supplier table.

**Ans : mysql> select min(status) from s;**

**+-------------+**

**| min(status) |**

**+-------------+**

**| 10 |**

**+-------------+**

1. Display the maximum Weight in the Parts table.

**Ans : mysql> select max(weight) from p;**

**+-------------+**

**| max(weight) |**

**+-------------+**

**| 1.5 |**

**+-------------+**

1. Display the average Weight of the Parts.

**Ans : mysql> select avg(weight) from p;**

**+--------------------+**

**| avg(weight) |**

**+--------------------+**

**| 0.9333333373069763 |**

**+--------------------+**

1. Display the total Quantity sold for part ‘P1’.

**Ans :**

**mysql> SELECT SUM(Qty) AS TotalQuantitySold**

**-> FROM spj**

**-> WHERE `P#` = 'P1';**

**+-------------------+**

**| TotalQuantitySold |**

**+-------------------+**

**| 100 |**

**+-------------------+**

1. Display the total Quantity sold for each part.

**Ans :**

**mysql> SELECT `P#`, SUM(Qty) AS TotalQuantitySold**

**-> FROM spj**

**-> GROUP BY `P#`;**

**+------+-------------------+**

**| P# | TotalQuantitySold |**

**+------+-------------------+**

**| P1 | 100 |**

**| P2 | 200 |**

**| P3 | 150 |**

1. Display the average Quantity sold for each part.

**Ans : mysql> SELECT `P#`, AVG(Qty) AS AverageQuantitySold**

**-> FROM spj**

**-> GROUP BY `P#`;**

**+------+---------------------+**

**| P# | AverageQuantitySold |**

**+------+---------------------+**

**| P1 | 100.0000 |**

**| P2 | 200.0000 |**

**| P3 | 150.0000 |**

**+------+---------------------+**

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

**Ans : mysql> SELECT `P#`, MAX(Qty) AS MaxQuantitySold**

**-> FROM spj**

**-> GROUP BY `P#`**

**-> HAVING MAX(Qty) > 800;**

**Empty set (0.00 sec)**

1. Display the Status and the count of Suppliers with that Status.

**Ans :**

**mysql> use classwork1;**

**Database changed**

**mysql> SELECT Status, COUNT(\*) AS SupplierCount**

**-> FROM s**

**-> GROUP BY Status;**

**+--------+---------------+**

**| Status | SupplierCount |**

**+--------+---------------+**

**| 30 | 1 |**

**| 20 | 1 |**

**| 10 | 1 |**

**| 25 | 1 |**

1. Display the count of Projects going on in different cities.

**Ans :**

**mysql> SELECT City, COUNT(\*) AS ProjectCount**

**-> FROM j**

**-> GROUP BY City;**

**+--------+--------------+**

**| City | ProjectCount |**

**+--------+--------------+**

**| Mumbai | 1 |**

**| Pune | 1 |**

**| Delhi | 1 |**

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

**Ans :**

**COUNT(\*)**

* **Counts all rows in the table.**
* **Includes rows even if they have NULL values.**
* **Used when you want the total row count.**

**COUNT(Status)**

* **Counts only rows where Status is not NULL.**
* **Skips rows with NULL in the Status column.**
* **Used when you want to count non-null values in a column.**

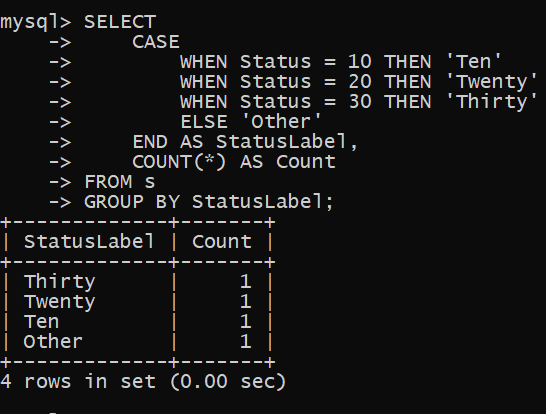
1. Display the Status and the Count of Suppliers with that Status in the following format as shown below:-

**Status Count**

Ten 1

Twenty 2

Thirty 3

**Ans : **