Experiment 5:

Title: To understand and use SQL Sub-Query

Objective: To understand the use of sql subquery.

1. Create the following table.

Supplier-(scode,sname,scity,turnover)

```
mysql> CREATE TABLE SUPPLIER(
    -> SCODE INT PRIMARY KEY,
    -> SNAME VARCHAR(20) NOT NULL,
    -> SCITY VARCHAR(15) NOT NULL,
    -> TURNOVER INT);
Query OK, 0 rows affected (0.05 sec)
```

Part-(pcode,weigh,color,cost,sellingprice)

```
Database changed
mysql> CREATE TABLE PART(
    -> PCODE INT PRIMARY KEY,
    -> WEIGHT INT NOT NULL,
    -> COLOR VARCHAR(10),
    -> COST INT NOT NULL,
    -> SELLINGPRICE INT NOT NULL);
Query OK, 0 rows affected (0.05 sec)
```

Supplier_Part-(scode,pcode,qty)

```
mysql> CREATE TABLE SUPPLIER_PART(
    -> SCODE INT NOT NULL,
    -> PCODE INT NOT NULL,
    -> QTY INT NOT NULL,
    -> FOREIGN KEY(SCODE) REFERENCES SUPPLIER(SCODE),
    -> FOREIGN KEY(PCODE) REFERENCES PART(PCODE));
Query OK, 0 rows affected (0.09 sec)
```

2. Populate the table

```
mysql> SELECT * FROM SUPPLIER;
+-------+---------------+----------+----------+
| SCODE | SNAME         | SCITY    | TURNOVER |
+-------+---------------+----------+----------+
| 11000 | George_Company | Bangalore | 10000000 |
| 11011 | Himanshu_Int.  | Mumbai    |  5000000 |
| 11031 | L_n_T          | Hyderabad | 15000000 |
+-------+---------------+----------+----------+
3 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM PART;
+-------+--------+-------+------+--------------+
| PCODE | WEIGHT | COLOR | COST | SELLINGPRICE |
+-------+--------+-------+------+--------------+
| 12001 |     70 | NULL  | 1300 |         1000 |
| 12011 |     30 | NULL  |  850 |          750 |
| 12012 |     50 | NULL  | 1500 |         1200 |
+-------+--------+-------+------+--------------+
3 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM SUPPLIER_PART;
+-------+-------+-----+
| SCODE | PCODE | QTY |
+-------+-------+-----+
| 11000 | 12001 |  10 |
| 11031 | 12012 |  25 |
+-------+-------+-----+
2 rows in set (0.00 sec)
```

3. Write appropriate SQL Statement for the following:

1. Get the supplier number and part number in ascending order of supplier number.

```
mysql> SELECT SCODE, PCODE
    -> FROM SUPPLIER_PART
    -> ORDER BY SCODE ASC;
+-------+-------+
| SCODE | PCODE |
+-------+-------+
| 11000 | 12001 |
| 11031 | 12012 |
+-------+-------+
2 rows in set (0.00 sec)
```

2. Get the details of supplier who operate from Bombay with turnover 50.

```
mysql> SELECT * FROM SUPPLIER WHERE SCITY="Mumbai";
+-------+--------------+--------+----------+
| SCODE | SNAME        | SCITY  | TURNOVER |
+-------+--------------+--------+----------+
| 11011 | Himanshu_Int. | Mumbai |  5000000 |
+-------+--------------+--------+----------+
1 row in set (0.00 sec)
```

3. Get the total number of supplier.

```
mysql> SELECT COUNT(*) AS total_suppliers
    -> FROM SUPPLIER;
+-----------------+
| total_suppliers |
+-----------------+
|               3 |
+-----------------+
1 row in set (0.03 sec)
```

4. Get the part number weighing between 25 and 35.

```
mysql> SELECT PCODE
    -> FROM PART
    -> WHERE WEIGHT BETWEEN 25 AND 35;
+-------+
| PCODE |
+-------+
| 12011 |
+-------+
1 row in set (0.01 sec)
```

5. Get the supplier number whose turnover is null.

```
mysql> SELECT SNAME FROM SUPPLIER WHERE TURNOVER="NULL";
Empty set, 1 warning (0.01 sec)
```

6. Get the part number that cost 20, 30 or 40 rupees.

```
mysql> SELECT PCODE
    -> FROM PART
    -> WHERE COST IN (20, 30, 40);
Empty set (0.00 sec)
```

7. Get the total quantity of part 12012 that is supplied.

```
mysql> SELECT SUM(QTY) AS total_quantity
    -> FROM SUPPLIER_PART
    -> WHERE PCODE = 12012;
+----------------+
| total_quantity |
+----------------+
|             25 |
+----------------+
1 row in set (0.01 sec)
```

8. Get the name of supplier who supply part 2.

9. Get the part number whose cost is greater than the average cost.

```
mysql> SELECT PCODE
    -> FROM PART
    -> WHERE COST > (SELECT AVG(COST) FROM PART);
+-------+
| PCODE |
+-------+
| 12001 |
| 12012 |
+-------+
2 rows in set (0.01 sec)
```

10. Get the supplier number and turnover in descending order of turnover.

```
mysql> SELECT SCODE, TURNOVER FROM SUPPLIER ORDER BY TURNOVER DESC;
+-------+----------+
| SCODE | TURNOVER |
+-------+----------+
| 11031 | 15000000 |
| 11000 | 10000000 |
| 11011 |  5000000 |
+-------+----------+
3 rows in set (0.00 sec)
```