MYSQL QUESTION 1

Q. Given table, Tutor, is shown below.

ID	NAME	AGE	CITY	FEE	PHONE
P1	SAMEER	34	DELHI	45000	9811076656
P2	ARYAN	35	NAGARKOT	54000	9911343989
P4	RAM	34	CHENNAI	45000	9810593578
P6	PREMLATA	36	BHOPAL	60000	9910139987
P7	SHIKHA	36	RAJKOT	34000	9912139456
P8	RADHA	33	DELHI	23000	8110668888

Write commands to do the following:

a) Display the name of those students in descending order whose age doesn't lie between 35 and 40.

```
mysql> SELECT NAME, AGE FROM TUTOR WHERE
    -> AGE NOT BETWEEN 35 AND 40 ORDER BY AGE DESC;
+--------+------+
| NAME   | AGE  |
+--------+------+
| SAMEER |   34 |
| RAM    |   34 |
| RADHA  |   33 |
+--------+------+
3 rows in set (0.00 sec)
```

b) List cities with their average fee in it.

```
mysql> SELECT CITY, AVG(FEE) FROM TUTOR GROUP BY CITY;
+----------+------------+
| CITY     | AVG(FEE)   |
+----------+------------+
| DELHI    | 34000.0000 |
| NAGARKOT | 54000.0000 |
| CHENNAI  | 45000.0000 |
| BHOPAL   | 60000.0000 |
| RAJKOT   | 34000.0000 |
+----------+------------+
5 rows in set (0.04 sec)
```

c) Decrease the fees of Shikha by 5%.

```
mysql> UPDATE TUTOR SET FEE=(FEE - 0.05*FEE) WHERE NAME='SHIKHA';
Query OK, 1 row affected (0.27 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> SELECT * FROM TUTOR;
+----+----------+------+----------+-------+------------+
| ID | NAME     | AGE  | CITY     | FEE   | PHONE      |
+----+----------+------+----------+-------+------------+
| P1 | SAMEER   |   34 | DELHI    | 45000 | 9811076656 |
| P2 | ARYAN    |   35 | NAGARKOT | 54000 | 9911343989 |
| P4 | RAM      |   34 | CHENNAI  | 45000 | 9810593578 |
| P6 | PREMLATA |   36 | BHOPAL   | 60000 | 9910139987 |
| P7 | SHIKHA   |   36 | RAJKOT   | 32300 | 9912139456 |
| P8 | RADHA    |   33 | DELHI    | 23000 | 8110668888 |
+----+----------+------+----------+-------+------------+
6 rows in set (0.00 sec)
```

d) Display cities where fees are maximum and minimum respectively.

```
mysql> SELECT CITY, FEE FROM TUTOR WHERE FEE=(SELECT MAX(FEE) FROM TUTOR)
    -> OR FEE=(SELECT MIN(FEE) FROM TUTOR);
+--------+-------+
| CITY   | FEE   |
+--------+-------+
| BHOPAL | 60000 |
| DELHI  | 23000 |
+--------+-------+
2 rows in set (0.00 sec)
```

e) Display the name and city of tutor who lives in a city having 'O' but not 'P'.

```
mysql> SELECT NAME, CITY FROM TUTOR WHERE
    -> CITY NOT LIKE '%P%' AND CITY LIKE '%O%';
+--------+----------+
| NAME   | CITY     |
+--------+----------+
| ARYAN  | NAGARKOT |
| SHIKHA | RAJKOT   |
+--------+----------+
2 rows in set (0.00 sec)
```

MYSQL QUESTION 2

Q. Consider the following tables, WATCHES and SALES, and write the SQL commands for subparts (i) to (v):

WatchId	Watch_Name	Price	Type	Qty_Store
W001	High Time	10000	Unisex	100
W002	Life Time	15000	Ladies	150
W003	Wave	20000	Gents	200
W004	High Fashion	7000	Unisex	250
W005	Golden Time	2500	Gents	100

WATCHES

WatchId	Qty_Sold	Quarter
W001	10	1
W003	5	1
W002	20	2
W003	10	2
W001	15	3
W002	20	3
W005	10	3
W003	15	4

SALES

a) To display watch name and their quantity sold in first quarter.

```
mysql> SELECT WATCH_NAME, QTY_SOLD, QUARTER FROM WATCHES W, SALES S
    -> WHERE W.WATCHID=S.WATCHID AND QUARTER=1;
+------------+----------+---------+
| WATCH_NAME | QTY_SOLD | QUARTER |
+------------+----------+---------+
| HIGH TIME  |       10 |       1 |
| WAVE       |        5 |       1 |
+------------+----------+---------+
2 rows in set (0.00 sec)
```

b) To display the details of those watches whose name ends with 'Time'.

```
mysql> SELECT * FROM WATCHES WHERE WATCH_NAME LIKE '%TIME';
+---------+-------------+-------+--------+-----------+
| WatchId | Watch_Name  | Price | Type   | Qty_Store |
+---------+-------------+-------+--------+-----------+
| W001    | HIGH TIME   | 10000 | UNISEX |       100 |
| W002    | LIFE TIME   | 15000 | LADIES |       150 |
| W005    | GOLDEN TIME |  2500 | GENTS  |       100 |
+---------+-------------+-------+--------+-----------+
3 rows in set (0.00 sec)
```

c) To display total quantity in store of Unisex type watches.

```
mysql> SELECT TYPE, SUM(QTY_STORE) FROM WATCHES
    -> WHERE TYPE='UNISEX';
+--------+----------------+
| TYPE   | SUM(QTY_STORE) |
+--------+----------------+
| UNISEX |            350 |
+--------+----------------+
1 row in set (0.00 sec)
```

d) To display watch's name and price of those watches which have price range in between 5000-15000.

```
mysql> SELECT WATCH_NAME, PRICE FROM WATCHES
    -> WHERE PRICE BETWEEN 5000 AND 15000;
+--------------+-------+
| WATCH_NAME   | PRICE |
+--------------+-------+
| HIGH TIME    | 10000 |
| LIFE TIME    | 15000 |
| HIGH FASHION |  7000 |
+--------------+-------+
3 rows in set (0.00 sec)
```

e) To display Quantity sold of all watches WatchId wise.

```
mysql> SELECT WATCHID, SUM(QTY_SOLD) FROM SALES
    -> GROUP BY WATCHID;
+---------+---------------+
| WATCHID | SUM(QTY_SOLD) |
+---------+---------------+
| W001    |            25 |
| W003    |            30 |
| W002    |            40 |
| W005    |            10 |
+---------+---------------+
4 rows in set (0.00 sec)
```