

## DBMS LAB-6 AGGREGATE FUNCTION

NAME: GAURAV MAHAJAN

SRN: PES1UG20CS150

1. Find the average distance between subsequent stations for every train

```
MariaDB [PES1UG20CS150_RRS_UPDATED]> select t.train_no,avg(distance) from train as t natural join route_info as r where
r.to_station_no - 1 = r.from_station_no group by train_no;
```

train_no	avg(distance)
25260	175.5000
25261	160.3333
58450	168.0000
58451	167.6667
62620	90.2500
62621	90.5000

6 rows in set (0.001 sec)

2. Find the average distance between subsequent stations for every train and display them in descending order of distance

```
MariaDB [PES1UG20CS150_RRS_UPDATED]> select t.train_no ,avg(distance) from train as t natural join route_info as r where
r.to_station_no - 1=r.from_station_no group by train_no order by avg(distance) desc;
```

train_no	avg(distance)
25260	175.5000
58450	168.0000
58451	167.6667
25261	160.3333
62621	90.5000
62620	90.2500

6 rows in set (0.036 sec)

3. Display the list of train numbers and the total distance travelled by each in descending order of the distance travelled

```
MariaDB [PES1UG20CS150_RRS_UPDATED]> select train_no, distance from train natural join route_info where source=from_station_name and destination=to_station_name group by train_no order by distance desc;
```

train_no	distance
58450	504
58451	503
25260	481
25261	481
62621	362
62620	361

6 rows in set (0.001 sec)

```
MariaDB [PES1UG20CS150_RRS_UPDATED]> select train_no, distance from train natural join route_info where source=from_station_name and destination=to_station_name group by train_no order by distance desc;
```

train_no	distance
58450	504
58451	503
25260	481
25261	481
62621	362
62620	361

```
6 rows in set (0.001 sec)
```

4. List those trains that have maximum and minimum number compartments and also display number of compartments they have. (2 queries one to find max and other to find min)

```
MariaDB [PES1UG20CS150_RRS_UPDATED]> create view compartment_number as select train_name, count(*) as count from train, compartment where train_no=train_number group by train_name order by count;
```

Query OK, 0 rows affected (0.007 sec)

```
MariaDB [PES1UG20CS150_RRS_UPDATED]> select train_name from compartment_number where count=(select max(count) from compartment_number);
```

train_name
BEN-CHE Shatabdi

```
1 row in set (0.004 sec)
```

```
MariaDB [PES1UG20CS150_RRS_UPDATED]> select train_name from compartment_number where count=(select min(count) from compartment_number);
```

train_name
CHE-BEN Shatabdi
BEN-MAN Express

```
2 rows in set (0.001 sec)
```

```
MariaDB [PES1UG20CS150_RRS_UPDATED]>
```

5. Display the number of phone numbers corresponding to the user\_id(s) ADM\_001, USR\_006, USR\_10

```
MariaDB [PES1UG20CS150_RRS_UPDATED]> select user_id, count(user_id) as no_of_phone from user_phone where user_id in ('ADM_001','USR_006','USR_010') group by user_id;
```

user_id	no_of_phone
ADM_001	2
USR_006	2
USR_010	2

```
3 rows in set (0.008 sec)
```

6. Find the average fare per km for each train type specified and display the train type and corresponding average fare per km as 'Avg\_Fare' in decreasing order of Avg\_Fare

```

MariaDB [PES1UG20CS150_RRS_UPDATED]> select train_type, avg(fare_per_km) as avg_fare from fare group by train_type order
by avg(fare_per_km) desc;
+-----+-----+
| train_type | avg_fare |
+-----+-----+
| Fast      | 2.0000   |
| Superfast | 2.0000   |
| Mail      | 1.3333   |
+-----+-----+
3 rows in set (0.001 sec)

MariaDB [PES1UG20CS150_RRS_UPDATED]>

```

7. Retrieve all details of the oldest passenger.

```

MariaDB [PES1UG20CS150_RRS_UPDATED]> select * from ticket_passenger where age = (select max(age) from ticket_passenger);
+-----+-----+-----+-----+
| seat_no | name   | age | pnr |
+-----+-----+-----+-----+
| F01-13 | Ramya R | 45 | PNR012 |
+-----+-----+-----+-----+
1 row in set (0.007 sec)

```

8. Count the number of passengers whose name consists of 'Ulla'. (Hint: Use the LIKE operator)

```

MariaDB [PES1UG20CS150_RRS_UPDATED]> select count(name) from ticket_passenger where name like '%Ulla%';
+-----+
| count(name) |
+-----+
| 4 |
+-----+
1 row in set (0.001 sec)

```