

PERFORM USER REQUIREMENTS: SMART RIDE ANALYTICS SYSTEM

DQL (Data Query Language):

1. Retrieve all users

```
select * from users;
```

USER_ID	NAME	EMAIL	PHONE_NUMBER	REGISTRATION_DATE	LOCATION
29	100 Achutha_Raman	achusrinivasan77@gmail.com	7365842419	10-03-23	Pallavaram
30	101 Swetha	swethasubramian06@gmail.com	8220352454	03-04-22	Velacherry
31	102 Uma_Maheswari	umasrinivasan@gmail.co	7245845555	02-05-25	T.Nagar
32	103 Latha	lathasankar@gmail.com	9944572645	26-02-21	Meenampakkam
33	104 Kathir	kathirsankar@gmail.com	6235897412	03-01-18	Koyambedu
34	105 Sankar	sankaraarumuagam@gmail.com	8975461236	12-01-23	Kovilampakkam
35	106 Subathra	subathrakrish@gmail.com	9584756321	21-02-12	Thiruvaiyaru
36	107 Dhivya_Sri	divyasrikannan@gmail.com	7584963215	10-07-02	Sirucherry
37	108 Subramania_Sharma	subramaniasharma@gmail.com	7339658412	06-11-96	Medavakkam
38	109 Saipriya	saipriya@gmail.com	9645781235	19-09-07	Pammal
39	110 Venkatesh	venkateshiyar@gmail.com	7845631296	12-12-94	Maraimalai nagar
40	111 Sneka	snekasivakumar@gmail.com	8220356545	26-12-01	Kumbakonam
41	112 Sivakumar	sivakumarpillai2gmail.com	7863451248	05-09-80	Krishnapuram

2. Get the total number of rides completed

```
SELECT COUNT(*) AS total_rides FROM rides WHERE ride_status = 'Completed';
```

TOTAL RIDES
13

3. Find the highest fare paid for a ride

```
SELECT Max(total_fare) AS highest_fare FROM rides;
```

HIGHEST_FARE
1010

4. Retrieve all vehicle details for a specific type (e.g., Car)

```
SELECT * FROM vehicle_details WHERE vehicle_type = 'Car';
```

VEHICLE_ID	DRIVER_ID	VEHICLE_TYPE	VEHICLE_NO	MODEL	YEAR
4041	1001	Car	7836	Tesla	05-01-22
4045	1005	Car	9813	Ford	14-08-82
4049	1009	Car	6347	Nissan	08-08-09
4051	1011	Car	9343	Hyundai	10-05-85
4057	1017	Car	2579	Audi	06-11-96
4060	1020	Car	7755	BMW	03-04-08
4064	1024	Car	6565	Ferrari	07-07-15
4065	1025	Car	2048	Rolls-Royce	15-12-07
4070	1030	Car	6628	Cooper	12-04-78
4075	1035	Car	9944	Volvo	24-11-92

5. List rides with user details and driver details

```
SELECT r.ride_id, u.name AS user_name, d.name AS driver_name, r.ride_status, r.total_fare
FROM rides r
JOIN users u ON r.user_id = u.user_id
JOIN drivers d ON r.driver_id = d.driver_id;
```

RIDE_ID	USER_NAME	DRIVER_NAME	RIDE_STATUS	TOTAL_FARE
1	Achutha_Raman	Durai_Samy	Completed	248
2	Swetha	Kalidas	Cancelled	96
3	Uma_Maheswari	Marimuthu	Ongoing	498
4	Latha	Srinivasa_Iyar	Completed	641
5	Kathir	Suganya	Cancelled	88
6	Sankar	Kannan	Completed	349
7	Subathra	Mekala	Ongoing	680
8	Dhivya_Sri	Karuppasamy	Cancelled	781
9	Subramania_Sharma	Babu	Completed	94
10	Saipriya	Sundari	Completed	257
11	Venkatesh	Meenamma	Ongoing	647
12	Sneka	Kunasekaran	Ongoing	593
13	Sivakumar	Perumal	Completed	110
14	Lakshmi	Paneer_Selvam	Completed	100
15	Atchaya	Mukila	Cancelled	300

6. Calculate total earnings of each driver

```
SELECT d.driver_id, d.name, SUM(r.total_fare) AS total_earnings
FROM rides r
JOIN drivers d ON r.driver_id = d.driver_id
GROUP BY d.driver_id, d.name;
```

DRIVER_ID	NAME	TOTAL_EARNINGS
1001	Durai_Samy	248
1002	Kalidas	96
1005	Suganya	88
1025	Boopathi	310
1029	Veera	406
1014	Paneer_Selvam	100
1016	Balu	512
1020	Nirmala	35
1028	Manoj	872
1030	Vadivu	1010
1009	Babu	94

7. Find the average rating for each driver

```
SELECT d.driver_id, d.name, AVG(f.rating) AS avg_rating
FROM feedback f
JOIN rides r ON f.ride_id = r.ride_id
JOIN drivers d ON r.driver_id = d.driver_id
GROUP BY d.driver_id, d.name;
```

DRIVER_ID	NAME	AVG_RATING
1001	Durai_Samy	2
1002	Kalidas	5
1005	Suganya	4
1025	Boopathi	4
1029	Veera	3
1014	Paneer_Selvam	2
1016	Balu	5
1020	Nirmala	1
1028	Manoj	1
1030	Vadivu	4
1009	Babu	2
1018	Mohamad	4
1026	Rajes	2

8. Retrieve all rides for a specific user, ordered by ride date

```
SELECT * FROM rides
WHERE user_id = 108
ORDER BY ride_date;
```

RIDE_ID	USER_ID	DRIVER_ID	PICKUP_LOCATION	DROPOFF_LOCATION	RIDE_STATUS	RIDE_DATE	TOTAL_FARE
9	108	1009	valasaravakkam	Tharamani	Completed	04-10-12	94

9. Find the total revenue generated from rides

```
SELECT SUM(total_fare) AS total_revenue FROM rides;
```

TOTAL_REVENUE
13415

10. Find the drivers who haven't received any feedback

```
SELECT d.driver_id, d.name
FROM drivers d
LEFT JOIN rides r ON d.driver_id = r.driver_id
LEFT JOIN feedback f ON r.ride_id = f.ride_id
WHERE f.feedback_id IS NULL;
```

DRIVER_ID	NAME
1059	Naresh
1040	Radhika
1041	Ram_Prasath
1047	Saroja
1058	Prasanna
1051	Vijay
1042	Aravind
1056	Madhi
1050	Amirtha
1034	Karthi
1057	Thennarasu
1037	Kumara_Vel

11. List all rides where fare is above average fare

```
SELECT * FROM rides WHERE total_fare > (SELECT AVG(total_fare) FROM rides);
```

RIDE_ID	USER_ID	DRIVER_ID	PICKUP_LOCATION	DROPOFF_LOCATION	RIDE_STATUS	RIDE_DATE	TOTAL_FARE
3	102	1003	T.Nagar	Virugambakkam	Ongoing	15-04-12	498
4	103	1004	Meenampakkam	Koyambedu	Completed	02-05-98	641
7	106	1007	Thiruvaiyaru	BesantNagar	Ongoing	07-02-16	680
8	107	1008	Pammal	Porur	Cancelled	02-08-18	781
11	110	1011	Solinganallur	Alwarpet	Ongoing	28-03-25	647
12	111	1012	Kummidipoondi	Adyar	Ongoing	24-09-22	593
16	115	1016	Nungambakkam	Koyambedu	Ongoing	05-12-06	512
19	118	1019	Sirucherry	Perungudi	Cancelled	01-03-05	845
22	122	1022	Velacherry	Keeladi	Cancelled	06-09-22	980
23	123	1023	Krishnapuram	T.Nagar	Completed	04-12-14	786
24	124	1024	Maraimalainagar	Thuraipakkam	Completed	26-12-14	608
26	126	1026	Mambalam	BesantNagar	Cancelled	02-08-09	456

12. Count rides per day with TO_CHAR function

```
SELECT TO_CHAR(ride_date, 'YYYY-MM-DD') AS ride_day, COUNT(*) AS total_rides
FROM rides
GROUP BY TO_CHAR(ride_date, 'YYYY-MM-DD')
ORDER BY ride_day;
```

RIDE_DAY	TOTAL_RIDES
1997-04-30	1
1998-05-02	1
2000-05-16	1
2000-11-10	1
2001-06-03	1
2002-02-14	1
2002-10-28	1
2005-03-01	1
2005-03-11	1
2005-12-04	1
2006-04-13	1
2006-12-05	1
2008-07-13	1

13. Get the length of driver names using LENGTH function

```
SELECT name, LENGTH(name) AS name_length
FROM drivers
ORDER BY name_length DESC;
```

NAME	NAME_LENGTH
Kaliya_Perumal	14
Srinivasa_Iyar	14
Paneer_Selvam	13
Kamala_Devi	11
Edwin_Kumar	11
Karuppasamy	11
Kunasekaran	11
Ram_Prasath	11
Mariyappan	10
Kumara_Vel	10
Anna_Malai	10
Rathna_Vel	10
Durai_Samy	10

14. Count feedback ratings per ride status using NVL function

```
SELECT r.ride_status, COUNT(NVL(f.rating, 0)) AS total_feedbacks
FROM rides r
LEFT JOIN feedback f ON r.ride_id = f.ride_id
GROUP BY r.ride_status
ORDER BY total_feedbacks DESC;
```

RIDE_STATUS	TOTAL_FEEDBACKS
Completed	13
Cancelled	8
Ongoing	8
ongoing	1

15. Find users registered in the last 30 days using SYSDATE function

```
SELECT * FROM users
WHERE registration_date >= SYSDATE - 30
ORDER BY registration_date DESC;
```

USER_ID	NAME	EMAIL	PHONE_NUMBER	REGISTRATION_DATE	LOCATION
102	Uma_Maheswari	umasrinivasan@gmail.co	7245845555	02-05-25	T.Nagar
155	Bhuvana	bhuvanaselvam@gmail.com	9473558735	30-04-25	Ayanavaram

16. Get the lowest fare per ride status using MIN function

```
SELECT ride_status, MIN(total_fare) AS lowest_fare
FROM rides
GROUP BY ride_status
ORDER BY lowest_fare ASC;
```

RIDE_STATUS	LOWEST_FARE
1 Completed	35
2 Cancelled	88
3 Ongoing	211
4 ongoing	354

17. Get the highest and lowest ride fare using MAX & MIN functions

```
SELECT MAX(total_fare) AS highest_fare, MIN(total_fare) AS lowest_fare FROM rides;
```

HIGHEST_FARE	LOWEST_FARE
1010	35

18. Get payment details with INITCAP function to format payment methods

```
SELECT INITCAP(payment_method) AS formatted_method, SUM(amount) AS total_paid  
FROM transactions  
GROUP BY INITCAP(payment_method)  
ORDER BY total_paid DESC;
```

FORMATTED_METHOD	TOTAL_PAID
Cash	6561
Card	3299
Wallet	2890

19. Retrieve all transactions for a specific ride

```
SELECT * FROM transactions WHERE ride_id = 25;
```

PAYMENT_ID	RIDE_ID	PAYMENT_METHOD	AMOUNT	PAYMENT_DATE
2025	25	Card	854	17-02-13

20. Find Rides Taken Between Two Dates

```
SELECT * FROM rides  
WHERE ride_date BETWEEN TO_DATE('04-10-2012', 'DD-MM-YYYY') AND TO_DATE('23-06-2014', 'DD-MM-YYYY')  
ORDER BY ride_date;
```

RIDE_ID	USER_ID	DRIVER_ID	PICKUP_LOCATION	DROPOFF_LOCATION	RIDE_STATUS	RIDE_DATE	TOTAL_FARE
9	108	1009	valasaravakkam	Tharamani	Completed	04-10-12	94
5	104	1005	Koyambedu	Keeladi	Cancelled	23-06-14	88