Implementation Internship – (Xeno)

Objective: Understand Your Thought Process on Client Data Handling

Name : Harsh Jain
Reg no. : 22BET10020
College : VIT Bhopal University
Profile : portfolio

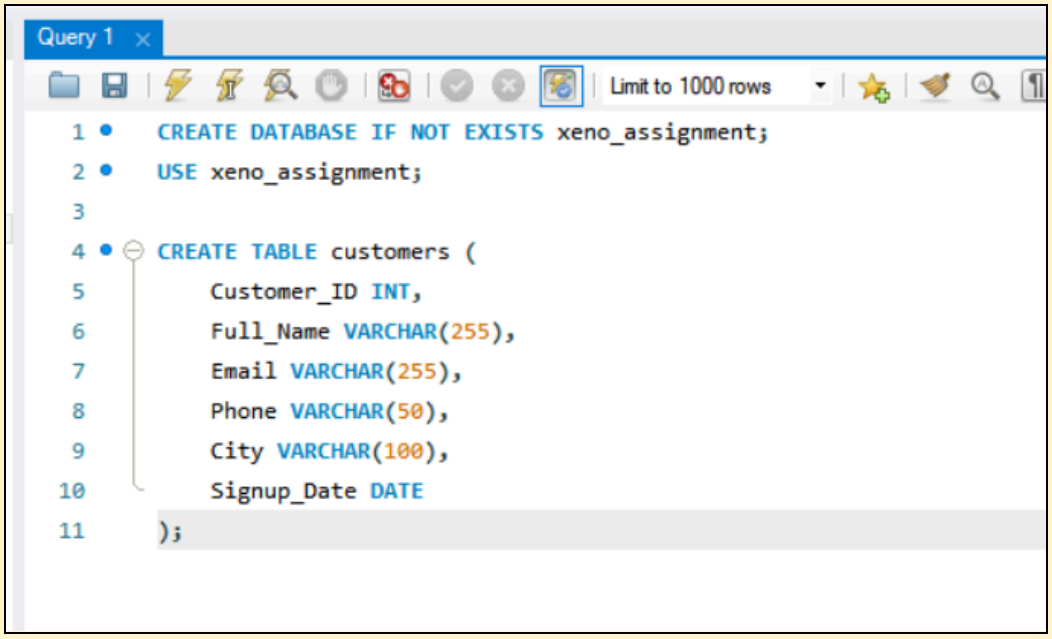
Assignment

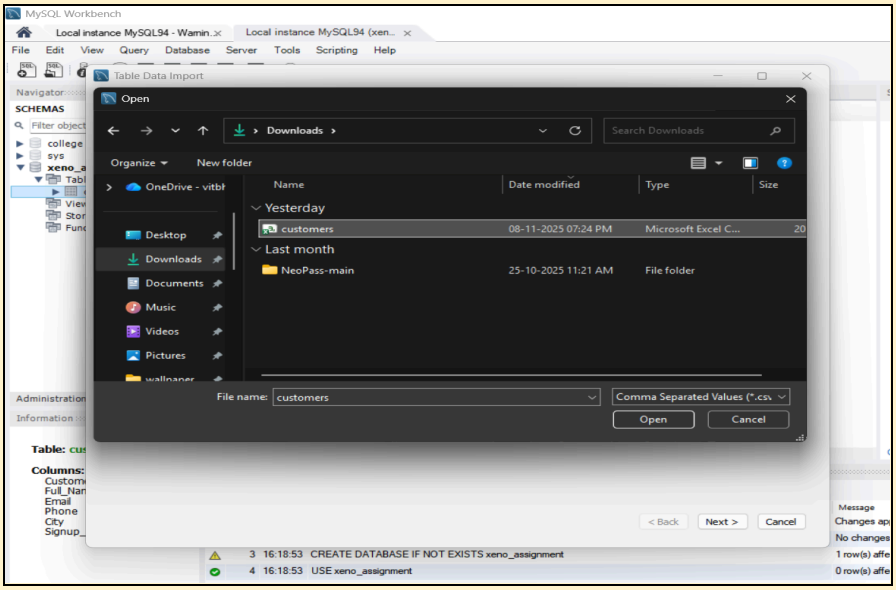
Assignment Tasks

Q1. SQL & Data Familiarity

1. **What steps would you take to review this data before importing it into a system? Please explain the process in not more than 3 lines.**
    - Before importing, we will review for any **inconsistency** in data types, **missing values** and verify the format of fields like email, phone and dates. We will identify **duplicates** with Customer ID and at last we check for **data integrity** like missing country code, typos.

2. **SQL Queries**
    - **Prerequisite**: Install MySQL and upload the data given in the attached file (Set the name of the table to be 'customers')
    - I will create a table and import the data.





1. **Task**: Write a query to display all customers from the city 'Delhi'.
   a. **Query**: SELECT * FROM customers WHERE city = 'Delhi';
   b. **Result**:
   c.

```
12 •  SELECT * FROM customers WHERE city = 'Delhi';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell

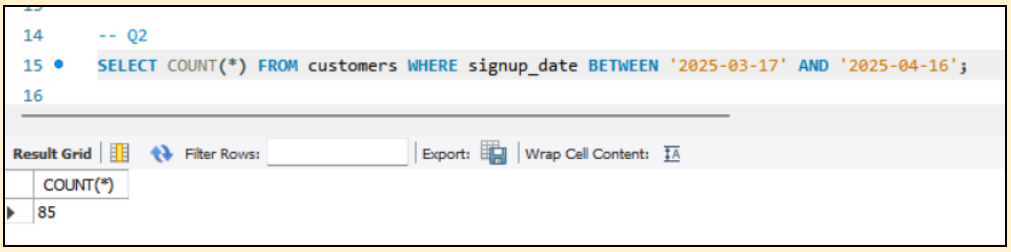
customer_id	full_name	email	phone_number	city	signup_date
131166	Samiha Batra		7150792111	Delhi	2023-12-24
156259	Hrishita Gopal	divijdora@hotmail.com	7344359134	Delhi	2024-08-28
159463	Divij Viswanathan	jwarrior@yahoo.com	9460578167	Delhi	2023-11-02
191011	Samarth Subramanian	jivin90@gmail.com	8340412987	Delhi	2023-09-10
196722	Adah Yogi	psoman@yahoo.com	8591581358	Delhi	2025-03-27
245018	Kabir Vaidya	nhanda@bal.org	8491731479	Delhi	2023-05-11
247923	Rania Bhattacharyya	devansh39@hotmail.com	8893549828	Delhi	2025-04-04
250616	Nishith Ramachandran	gbarman@chacko.net	7822335748	Delhi	2024-06-13
261495	Pihu Tailor	farhan12@chandra.com	7864389575	Delhi	2023-10-01
267289	Umang Kaur	hdara@gmail.com	8681162759	Delhi	2025-03-27
276828	Myra Sarna		9186475299	Delhi	2025-04-09
318287	Jiya Mann		9012500763	Delhi	2025-03-22
378194	Kavya Kadakia	golanayantara@gmail.com	7762967030	Delhi	2025-03-21
433818	Yasmin Dora	aborra@dhawan.com	9248283582	Delhi	2023-12-30
451453	Aarush Uppal	vritikabhargava@gmail....	8927631712	Delhi	2024-05-19
457035	Jhanvi Sehgal	darshitbose@sridhar.biz	8745916414	Delhi	2025-03-24

customers 2

2. **Task**: Count the number of signups in the last 30 days. Assume today to be 16th April 2025.
    a. **Query**:

```
SELECT COUNT(*)
FROM customers
WHERE signup_date
BETWEEN '2025-03-17' AND '2025-04-16';
```

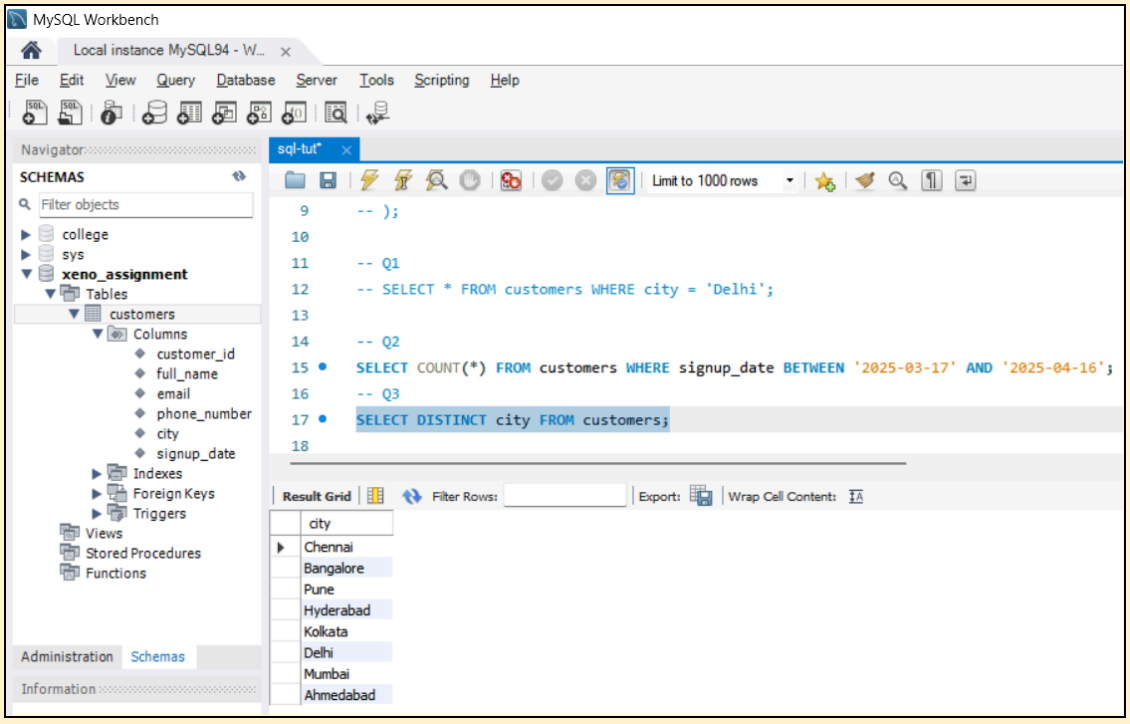
    b. **Result**: count = 85
    c.



3. **Task**: List unique cities where customers are based
    a. **Query**:

```
SELECT DISTINCT city
FROM customers;
```

    b. **Result**:
    c.



4. **Task**: List the top 3 cities by number of signups.
    a. **Query**:

    SELECT city, COUNT(*) AS signups

    FROM customers

    GROUP BY city ORDER BY signups DESC LIMIT 3;

    b. **Result**:

    c.

```
11    -- Q1
12    -- SELECT * FROM customers WHERE city = 'Delhi';
13
14    -- Q2
15    SELECT COUNT(*) FROM customers WHERE signup_date BETWEEN '2025-03-17' AND '2025-04-16';
16    -- Q3
17    SELECT DISTINCT city FROM customers;
18    -- Q4
19    SELECT city, COUNT(*) AS signups FROM customers GROUP BY city ORDER BY signups DESC LIMIT 3;
20
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

city	signups
Pune	44
Delhi	42
Bangalore	41

5. **Task**: Assume there's another table orders (customer_id, order_id, amount). How would you find customers who have never placed an order?
    a. **Query**:

```
SELECT * FROM customers
WHERE customerid NOT IN (SELECT customerid
FROM orders);
```

b. **Result**:

i. Created order table

```
-- Q5
CREATE TABLE orders (
    orderid INT PRIMARY KEY AUTO_INCREMENT,
    customer_id INT,
    orderdate DATE,
    amount DECIMAL(10,2),
    status VARCHAR(20),
    FOREIGN KEY (customer_id) REFERENCES customers(customer_id)
);
INSERT INTO orders (customer_id, orderdate, amount, status) VALUES
(251212, '2025-04-12', 749.99, 'Completed'),
(848133, '2025-04-10', 1200.50, 'Shipped'),
(511492, '2025-04-09', 450.00, 'Pending'),
(128097, '2025-03-28', 999.99, 'Completed'),
(905038, '2025-03-22', 545.60, 'Cancelled'),
(541502, '2025-03-21', 200.00, 'Completed'),
(866511, '2025-03-18', 319.50, 'Completed'),
(512203, '2025-03-15', 875.40, 'Shipped'),
(553987, '2025-03-14', 1120.00, 'Completed'),
(209910, '2025-03-13', 600.00, 'Pending');

SELECT * FROM customers WHERE customer_id NOT IN (SELECT customer_id FROM orders);
```

ii.

iii. Results

sql-tut*

Limit to 1000 rows

```
34    (905038, '2025-03-22', 545.60, 'Cancelled'),
35    (541502, '2025-03-21', 200.00, 'Completed'),
36    (866511, '2025-03-18', 319.50, 'Completed'),
37    (512203, '2025-03-15', 875.40, 'Shipped'),
38    (553987, '2025-03-14', 1120.00, 'Completed'),
39    (209910, '2025-03-13', 600.00, 'Pending');
40
41    SELECT * FROM customers WHERE customer_id NOT IN (SELECT customer_id FROM orders);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

customer_id	full_name	email	phone_number	city	signup_date
105156	Bhavin Kara		9617343581	Chennai	2025-04-04
105394	Eshani Kota	bhavinbhargava@hotmail.com	8151619403	Bangalore	2023-05-05
107397	Ojas Devan	stuvan83@gmail.com	9542793419	Bangalore	2025-04-08
109407	Miraya Sura	mannatraj@yahoo.com	9254427086	Pune	2023-05-11
110995	Mehul Balan	rajagopalankiara@hotmail.com	7202585038	Pune	2025-03-27
122197	Yuvraj Srinivas	tarini75@hotmail.com	7825021492	Hyderabad	2024-01-07
127033	Myra Keer	jayaramanamani@gmail.com	9258804475	Bangalore	2023-08-01
131166	Samiha Batra		7150792111	Delhi	2023-12-24
132742	Siya Shankar	omaharaj@grover.com	8027015672	Mumbai	2024-02-04
134669	Vedika Dube		7409997269	Chennai	2024-01-18
136596	Onkar Ahuja		9387224925	Hyderabad	2025-03-28
137770	Jayesh Bhandari	ritvikloyal@yahoo.com	7209194799	Hyderabad	2023-05-12
141005	Neelofar Gandhi		7350311370	Chennai	2023-05-02
143476	Pari Dutta		9929530775	Hyderabad	2023-12-04

2. Data Transformation & Enrichment

The operations teams want a few more details to ensure smooth operations.

1. **Task**: Add a new column is_gmail ('Yes'/'No').
    a. **Query**:

```
UPDATE customers
SET is_gmail = IF(COALESCE(email, '') LIKE '%@gmail.com', 'Yes',
'No')
WHERE customer_id IS NOT NULL;
```

        i. **"I had turned safe updates off."**
    b. **Result**:
        i.

```
47 • UPDATE customers
48   SET is_gmail = IF(COALESCE(email, '') LIKE '%@gmail.com', 'Yes', 'No')
49   WHERE customer_id IS NOT NULL;
50 • SET SQL_SAFE_UPDATES = 0;
51
52 • select * from customers
53
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

customer_id	full_name	email	phone_number	city	signup_date	is_gmail
105156	Bhavin Kara		9617343581	Chennai	2025-04-04	No
105394	Eshani Kota	bhavinbhargava@hotmail.com	8151619403	Bangalore	2023-05-05	No
107397	Ojas Devan	stuvan83@gmail.com	9542793419	Bangalore	2025-04-08	Yes
109407	Miraya Sura	mannatraj@yahoo.com	9254427086	Pune	2023-05-11	No
110995	Mehul Balan	rajagopalankiara@hotmail.com	7202585038	Pune	2025-03-27	No
122197	Yuvraj Srinivas	tarini75@hotmail.com	7825021492	Hyderabad	2024-01-07	No
127033	Myra Keer	jayaramanamani@gmail.com	9258804475	Bangalore	2023-08-01	Yes
128097	Jayesh Jha	urvi41@yahoo.com	9834699453	Kolkata	2025-03-24	No
131166	Samiha Batra		7150792111	Delhi	2023-12-24	No
132742	Siya Shankar	omaharaj@grover.com	8027015672	Mumbai	2024-02-04	No
134669	Vedika Dube		7409997269	Chennai	2024-01-18	No
136596	Onkar Ahuja		9387224925	Hyderabad	2025-03-28	No
137770	Jayesh Bhandari	rituvikdoyal@yahoo.com	7209194799	Hyderabad	2023-05-12	No

customers 9

2. **Task**: Extract the first_name from the full_name column.
   a. **Query**:

```
ALTER TABLE customers
ADD COLUMN first_name   VARCHAR(255);
UPDATE customers SET first_name =
SUBSTRING_INDEX(fullname, ' ', 1)
WHERE customer_id IS NOT NULL;
```

   b. **Result**:

```
52 •  ALTER TABLE customers ADD COLUMN first_name VARCHAR(255);
53 •  UPDATE customers SET first_name = SUBSTRING_INDEX(full_name, ' ', 1) WHERE customer_id IS NOT NULL;
54
55 •  select * from customers;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

customer_id	full_name	email	phone_number	city	signup_date	is_gmail	first_name
105156	Bhavin Kara		9617343581	Chennai	2025-04-04	No	Bhavin
105394	Eshani Kota	bhavinbhargava@hotmail.com	8151619403	Bangalore	2023-05-05	No	Eshani
107397	Ojas Devan	stuvan83@gmail.com	9542793419	Bangalore	2025-04-08	Yes	Ojas
109407	Miraya Sura	mannatraj@yahoo.com	9254427086	Pune	2023-05-11	No	Miraya
110995	Mehul Balan	rajagopalankiara@hotmail.com	7202585038	Pune	2025-03-27	No	Mehul
122197	Yuvraj Srinivas	tarini75@hotmail.com	7825021492	Hyderabad	2024-01-07	No	Yuvraj
127033	Myra Keer	jayaramanamani@gmail.com	9258804475	Bangalore	2023-08-01	Yes	Myra
128097	Jayesh Jha	urvi41@yahoo.com	9834699453	Kolkata	2025-03-24	No	Jayesh
131166	Samiha Batra		7150792111	Delhi	2023-12-24	No	Samiha
132742	Siya Shankar	omaharaj@grover.com	8027015672	Mumbai	2024-02-04	No	Siya
134669	Vedika Dube		7409997269	Chennai	2024-01-18	No	Vedika
136596	Onkar Ahuja		9387224925	Hyderabad	2025-03-28	No	Onkar
137770	Jayesh Bhandari	ritvikloyal@yahoo.com	7209194799	Hyderabad	2023-05-12	No	Jayesh
141005	Neelofar Gandhi		7350311370	Chennai	2023-05-02	No	Neelofar
143476	Pari Dutta		9929530775	Hyderabad	2023-12-04	No	Pari
156259	Hrishita Gopal	divijdora@hotmail.com	7344359134	Delhi	2024-08-28	No	Hrishita

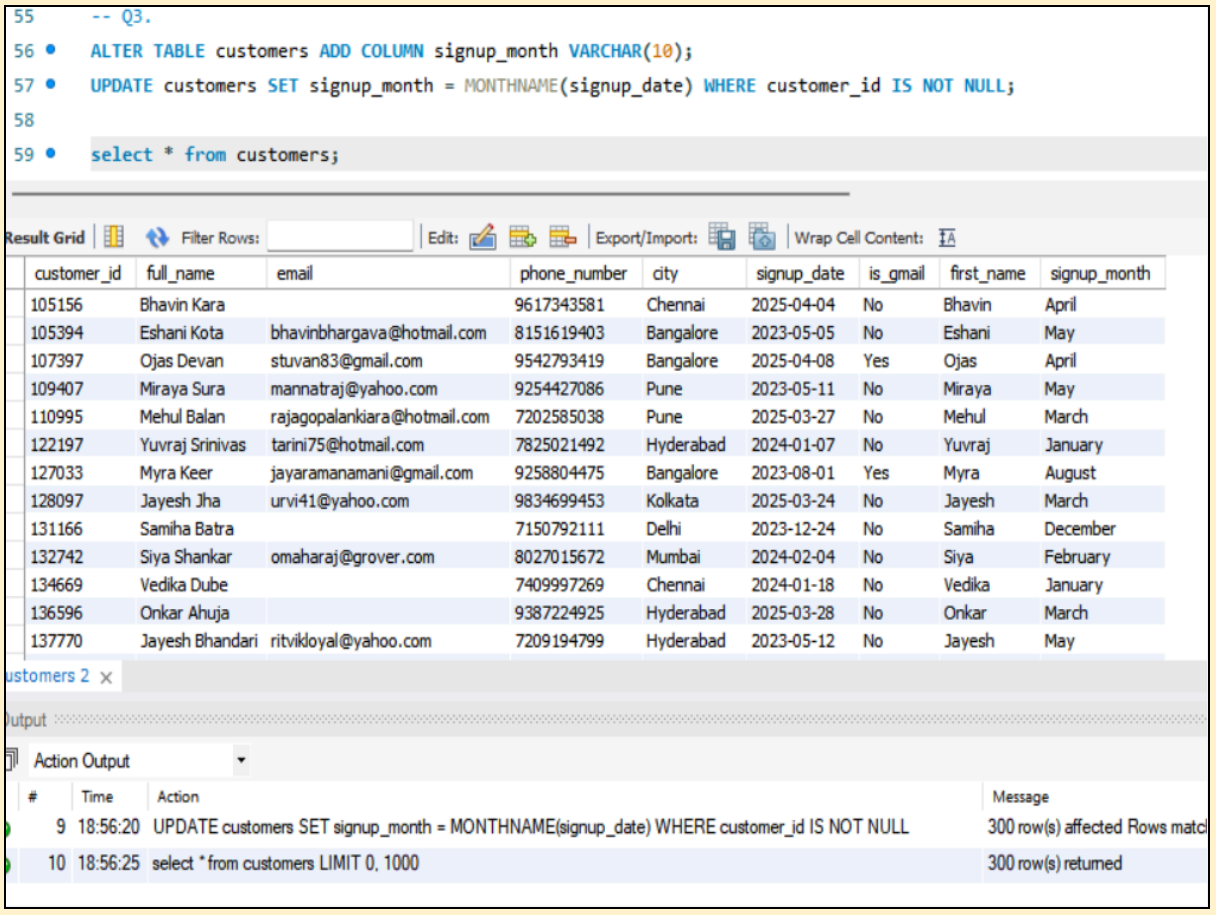
customers 1 ×

   c.

3. **Task**: Add a column signup_month.
   a. **Query**:

```
ALTER TABLE customers
ADD COLUMN signup_month VARCHAR(10);
UPDATE customers SET signup_month =
MONTHNAME(signup_date) WHERE customer_id IS NOT
NULL;
```

   b. **Result**:



```
55    -- Q3.
56 •  ALTER TABLE customers ADD COLUMN signup_month VARCHAR(10);
57 •  UPDATE customers SET signup_month = MONTHNAME(signup_date) WHERE customer_id IS NOT NULL;
58
59 •  select * from customers;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

customer_id	full_name	email	phone_number	city	signup_date	is_gmail	first_name	signup_month
105156	Bhavin Kara		9617343581	Chennai	2025-04-04	No	Bhavin	April
105394	Eshani Kota	bhavinbhargava@hotmail.com	8151619403	Bangalore	2023-05-05	No	Eshani	May
107397	Ojas Devan	stuvan83@gmail.com	9542793419	Bangalore	2025-04-08	Yes	Ojas	April
109407	Miraya Sura	mannatraj@yahoo.com	9254427086	Pune	2023-05-11	No	Miraya	May
110995	Mehul Balan	rajagopalankiara@hotmail.com	7202585038	Pune	2025-03-27	No	Mehul	March
122197	Yuvraj Srinivas	tarini75@hotmail.com	7825021492	Hyderabad	2024-01-07	No	Yuvraj	January
127033	Myra Keer	jayaramanamani@gmail.com	9258804475	Bangalore	2023-08-01	Yes	Myra	August
128097	Jayesh Jha	urvi41@yahoo.com	9834699453	Kolkata	2025-03-24	No	Jayesh	March
131166	Samiha Batra		7150792111	Delhi	2023-12-24	No	Samiha	December
132742	Siya Shankar	omaharaj@grover.com	8027015672	Mumbai	2024-02-04	No	Siya	February
134669	Vedika Dube		7409997269	Chennai	2024-01-18	No	Vedika	January
136596	Onkar Ahuja		9387224925	Hyderabad	2025-03-28	No	Onkar	March
137770	Jayesh Bhandari	ritvikloyal@yahoo.com	7209194799	Hyderabad	2023-05-12	No	Jayesh	May

ustomers 2 ×

Output

Action Output

#	Time	Action	Message
9	18:56:20	UPDATE customers SET signup_month = MONTHNAME(signup_date) WHERE customer_id IS NOT NULL	300 row(s) affected Rows matcl
10	18:56:25	select * from customers LIMIT 0, 1000	300 row(s) returned

   c.

4. **Task**: Create a report of GMAIL customers who signed up for each day of the week.
    a. **Query**:

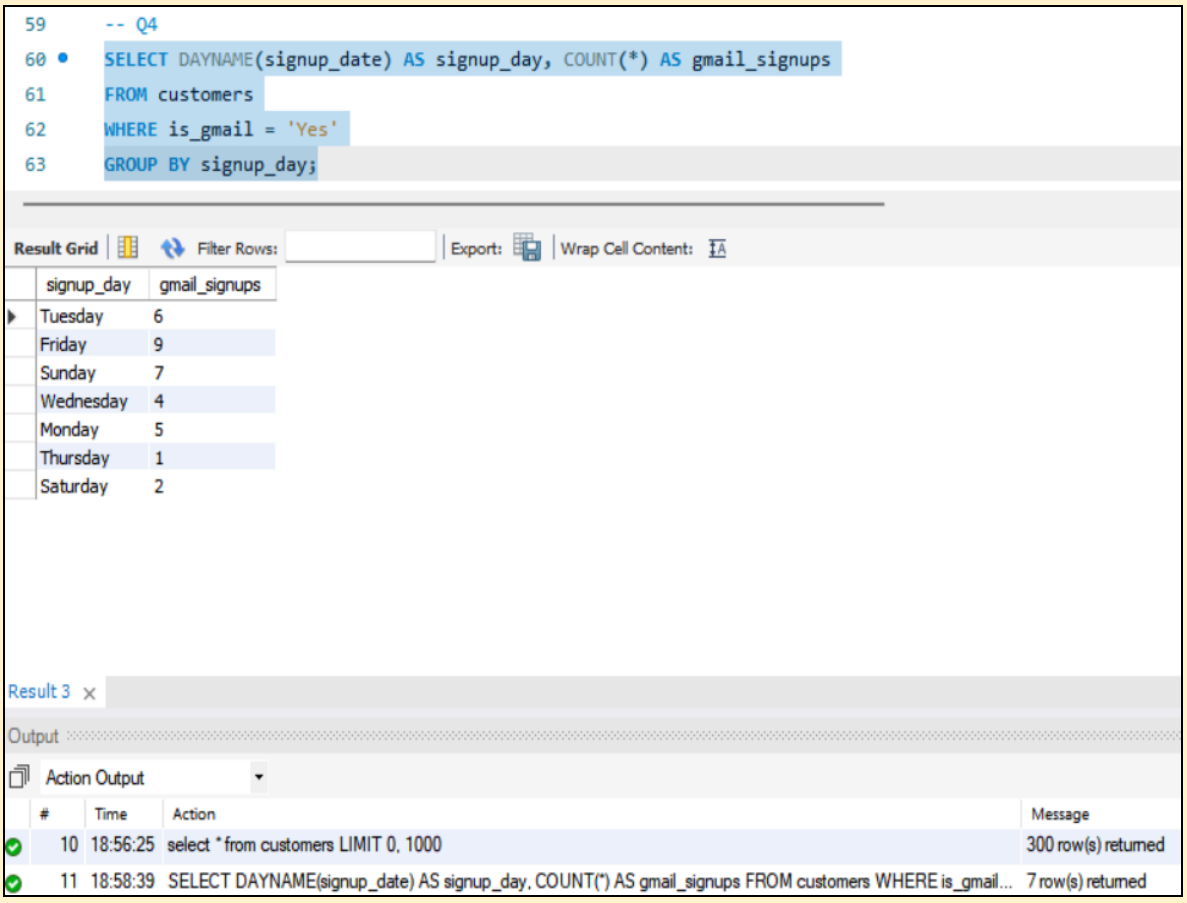
```
SELECT DAYNAME(signup_date) AS signup_day, COUNT(*)
AS gmail_signups

FROM customers

WHERE is_gmail = 'Yes'

GROUP BY signup_day;
```

    b. **Result**:



    c.

5. Task: Create a new table vip_customers (Delhi, Mumbai, Bangalore; signed up in the last 60 days from 16th April 2025).
   a. Query:

```
CREATE TABLE vip_customers AS
SELECT * FROM customers
WHERE city IN ('Delhi', 'Mumbai', 'Bangalore')
AND signup_date BETWEEN DATE_SUB('2025-04-16',
INTERVAL 60 DAY) AND '2025-04-16';
```

   b. Result:

   c.

```
65    -- Q5
66 •  CREATE TABLE vip_customers AS
67    SELECT * FROM customers
68    WHERE city IN ('Delhi', 'Mumbai', 'Bangalore')
69    AND signup_date BETWEEN DATE_SUB('2025-04-16', INTERVAL 60 DAY) AND '2025-04-16';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_id	full_name	email	phone_number	city	signup_date	is_gmail	first_name	signup_month
107397	Ojas Devan	stuvan83@gmail.com	9542793419	Bangalore	2025-04-08	Yes	Ojas	April
196722	Adah Yogi	psoman@yahoo.com	8591581358	Delhi	2025-03-27	No	Adah	March
224232	Divyansh Tripathi	jyohannan@datta.com	8647981998	Mumbai	2025-03-22	No	Divyansh	March
240679	Ahana Kumar	chirag63@sabharwal.com	8580016724	Mumbai	2025-03-18	No	Ahana	March
247923	Rania Bhattacharyya	devansh39@hotmail.com	8893549828	Delhi	2025-04-04	No	Rania	April
260398	Rasha Seshadri	tejasbhandari@yahoo.com	8496435124	Bangalore	2025-03-22	No	Rasha	March
266733	Adah Kamdar	gokul03@gmail.com	8643563127	Mumbai	2025-04-11	Yes	Adah	April
267289	Umang Kaur	hdara@gmail.com	8681162759	Delhi	2025-03-27	Yes	Umang	March
276828	Myra Sarna		9186475299	Delhi	2025-04-09	No	Myra	April
285490	Lakshay Loke	lcheema@sharaf.com	9040520243	Bangalore	2025-04-14	No	Lakshay	April
318287	Jiya Mann		9012500763	Delhi	2025-03-22	No	Jiya	March
340804	Sana Madan		9908406028	Bangalore	2025-04-14	No	Sana	April
378194	Kavya Kadakia	golanayantara@gmail.com	7762967030	Delhi	2025-03-21	Yes	Kavya	March

vip_customers 4

Output

Action Output

#	Time	Action	Message
12	19:02:01	CREATE TABLE vip_customers AS SELECT * FROM customers WHERE city IN ('Delhi', 'Mumbai', 'Bangalore') AN...	32 row(s) affected Records: 32 Duplicates: 0 Warnings: 0
13	19:02:16	select * from vip_customers LIMIT 0, 1000	32 row(s) returned

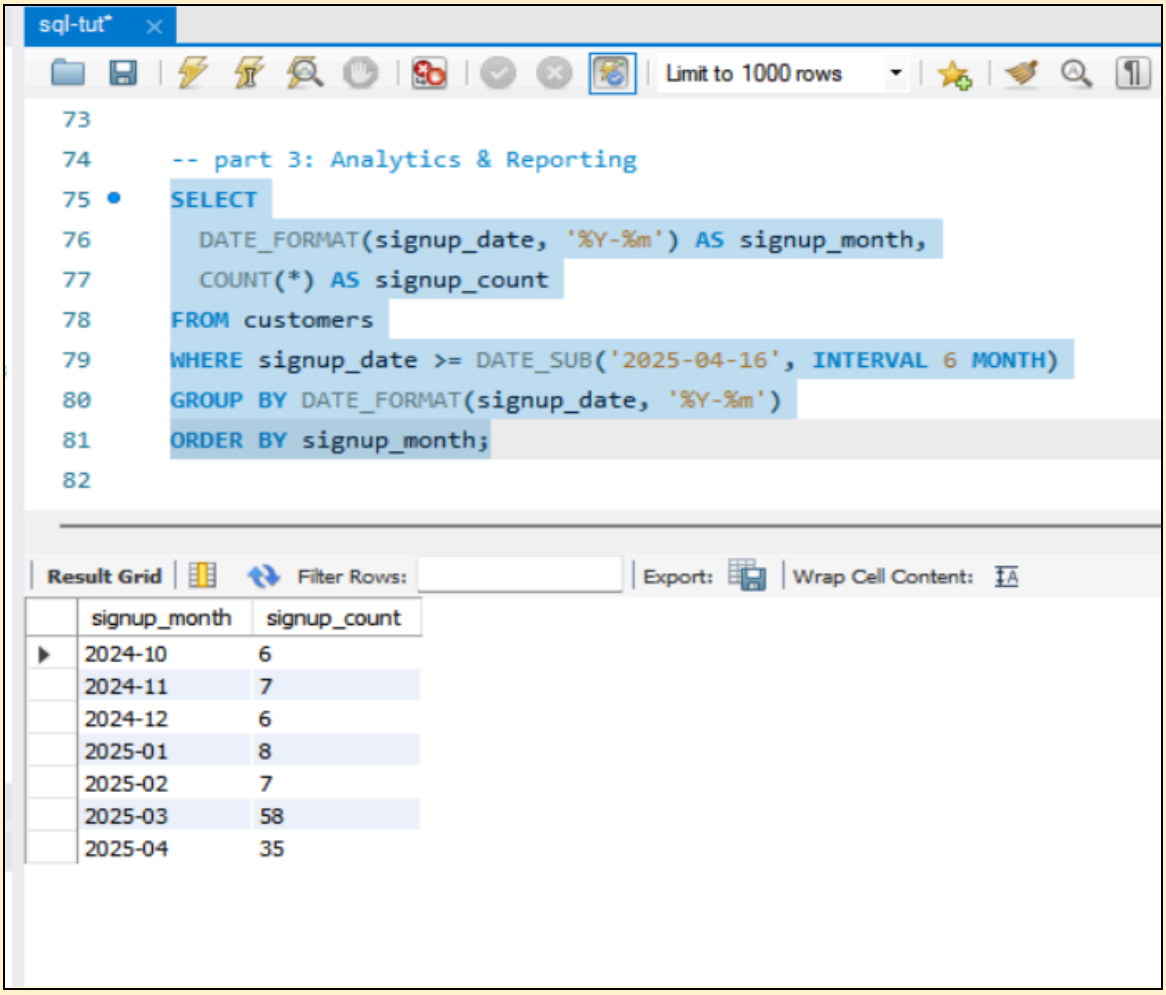
3. Analytics & Reporting

1. **Task**: Show a monthly signup count for the past 6 months (from 16th April 2025).
    a. **Query**:

```
SELECT
DATE_FORMAT(signup_date, '%Y-%m') AS signup_month,
COUNT(*) AS signup_count
FROM customers
WHERE signup_date >= DATE_SUB('2025-04-16', INTERVAL 6
MONTH)
GROUP BY DATE_FORMAT(signup_date, '%Y-%m')
ORDER BY signup_month;
```

    b. **Result**:

    c.



2. **Task**: Get a list of cities with more than 20 customers.
   a. **Query**:

```
SELECT city, COUNT(*) AS customer_count
FROM customers
GROUP BY city
HAVING customer_count > 20;
```

   b. **Result**:

```
81      ORDER BY signup_month;
82
83      -- q2.
84      SELECT city, COUNT(*) AS customer_count
85      FROM customers
86      GROUP BY city
87      HAVING customer_count > 20;
88
89
90
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

city	customer_count
Chennai	33
Bangalore	41
Pune	44
Hyderabad	39
Kolkata	41
Delhi	42
Mumbai	24
Ahmedabad	36

   c.

3. **Task**: Find the date with the highest number of signups.
   a. **Query**:

```
SELECT signup_date, COUNT(*) AS signup_count
FROM customers
GROUP BY signup_date
ORDER BY signup_count DESC
LIMIT 1;
```

   b. **Result**:

```
89    -- q3.
90    SELECT signup_date, COUNT(*) AS signup_count
91    FROM customers
92    GROUP BY signup_date
93    ORDER BY signup_count DESC
94    LIMIT 1;
95
96
```

Result Grid | Filter Rows: | Export: | Wrap Cell Co

signup_date	signup_count
2025-03-28	5

   c.

4. **Task**: Add a new column signup_day_name and find the day with the highest number of signups.
    a. **Query (Part A): Add a new column for signup day**

```
ALTER TABLE customers ADD COLUMN signup_day VARCHAR(20);
UPDATE customers SET signup_day = DAYNAME(signup_date)
WHERE customer_id IS NOT NULL;
SELECT *  FROM customers
```

    b. **Result (Part A):**

    c.

```
96    -- q4.
97    -- Step 1: Add a new column for signup day
98    ALTER TABLE customers ADD COLUMN signup_day VARCHAR(20);
99    UPDATE customers SET signup_day = DAYNAME(signup_date) WHERE customer_id IS NOT NULL;
100   select * from customers
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

customer_id	full_name	email	phone_number	city	signup_date	is_gmail	first_name	signup_month	signup_day
105156	Bhavin Kara		9617343581	Chennai	2025-04-04	No	Bhavin	April	Friday
105394	Eshani Kota	bhavinbhargava@hotmail.com	8151619403	Bangalore	2023-05-05	No	Eshani	May	Friday
107397	Ojas Devan	stuvan83@gmail.com	9542793419	Bangalore	2025-04-08	Yes	Ojas	April	Tuesday
109407	Miraya Sura	mannatraj@yahoo.com	9254427086	Pune	2023-05-11	No	Miraya	May	Thursday
110995	Mehul Balan	rajagopalankiara@hotmail.com	7202585038	Pune	2025-03-27	No	Mehul	March	Thursday
122197	Yuvraj Srinivas	tarini75@hotmail.com	7825021492	Hyderabad	2024-01-07	No	Yuvraj	January	Sunday
127033	Myra Keer	jayaramanamani@gmail.com	9258804475	Bangalore	2023-08-01	Yes	Myra	August	Tuesday
128097	Jayesh Jha	urvi41@yahoo.com	9834699453	Kolkata	2025-03-24	No	Jayesh	March	Monday
131166	Samiha Batra		7150792111	Delhi	2023-12-24	No	Samiha	December	Sunday
132742	Siya Shankar	omaharaj@grover.com	8027015672	Mumbai	2024-02-04	No	Siya	February	Sunday
134669	Vedika Dube		7409997269	Chennai	2024-01-18	No	Vedika	January	Thursday
136596	Onkar Ahuja		9387224925	Hyderabad	2025-03-28	No	Onkar	March	Friday
137770	Jayesh Bhandari	ritvikloyal@yahoo.com	7209194799	Hyderabad	2023-05-12	No	Jayesh	May	Friday

customers 8

Output

Action Output

#	Time	Action	Message
19	19:28:57	UPDATE customers SET signup_day = DAYNAME(signup_date) WHERE customer_id IS NOT NULL	300 row(s) affected Rows matched: 300 Changed: 300 W
20	19:29:23	select * from customers LIMIT 0, 1000	300 row(s) returned

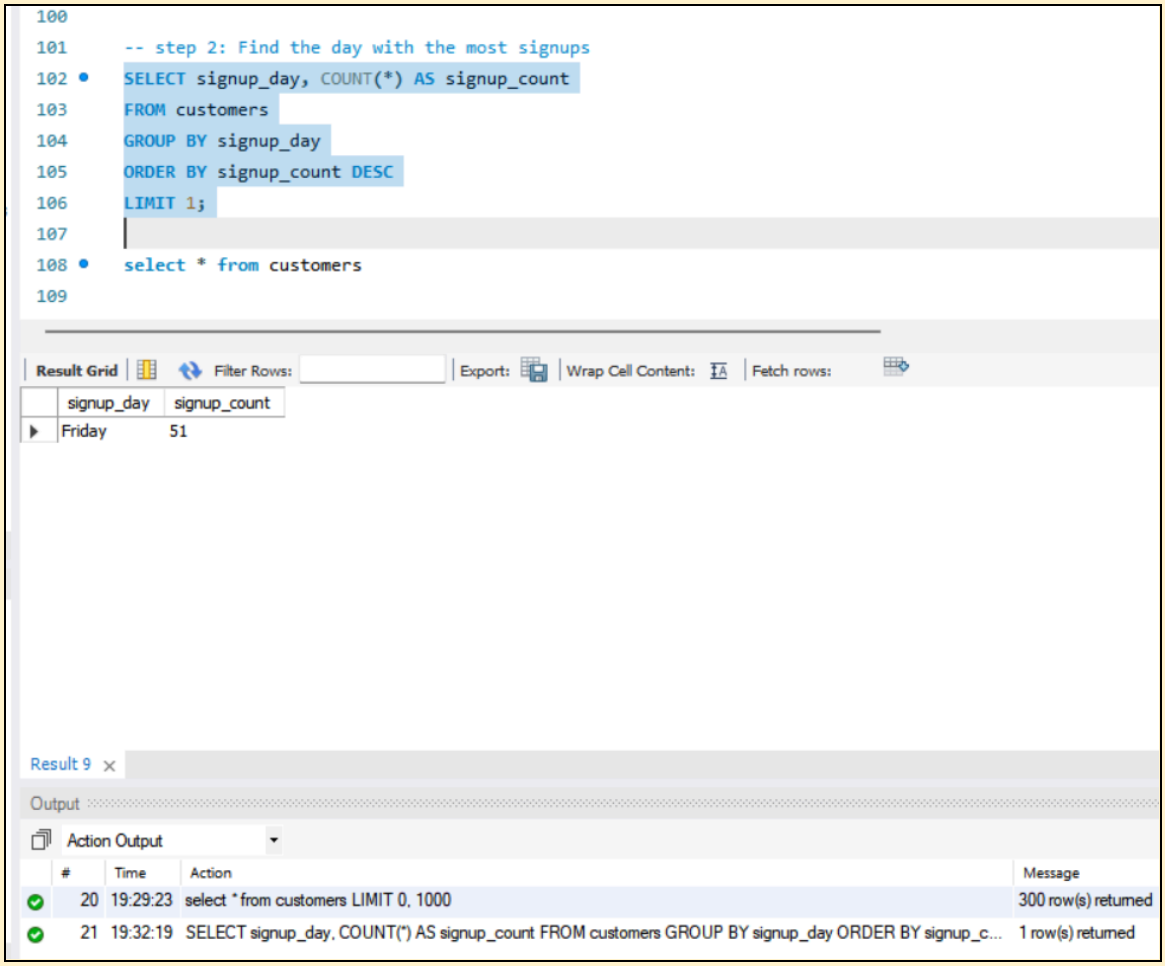
*

d. Query (Part B): Find the day with the most signups

```
SELECT signup_day, COUNT(*) AS signup_count
FROM customers
GROUP BY signup_day
ORDER BY signup_count DESC
LIMIT 1;
```

e. Result (Part B):

f.



4. My AI Usage Declaration

As requested, this section clarifies the use of AI in this assignment.

I used perplexity (an AI tool) to help structure this document, debug situations and to understand issues I faced while writing queries like "**where I had to turn safe updates off.**". All queries were run and screenshots captured from my local MySQL server.

5. Assignment Summary

Through this assignment, I learned a lot about SQL queries and practical data analysis in relational databases. The end-to-end process— from data cleaning and import to creating new columns, transforming values, and generating analytic reports—greatly improved my SQL proficiency and exposed me to important real-world data problems.

Resource Links i used:

- W3Schools SQL Tutorial
- SQLZoo
- GeeksforGeeks SQL Tutorial