

Name : Mehnaz Shafeek

Reg No: 22BHI10211

Implementation Internship Assignment

1. SQL & Data Familiarity

- 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.

Ans. I would validate the structure of the dataset, encoding (UTF-8), and consistent delimiters and headers before importing. After which, I will validate the data types, formats (particularly for dates), and conformity to the target schema. I would identify, then resolve, missing, duplicate, or invalid records to maintain referential integrity. Finally, I would do a small test import to confirm that the mapping was successful and compatible with the system.

- Write a query to display all customers from the city 'Delhi'.

The screenshot shows the SSMS interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. Below the menu is a toolbar with various icons. The left pane is the Navigator, showing the SCHEMAS node expanded to show sys and xeno_assignment, with Tables selected. The main area is titled 'Query 1' and contains the following SQL code:

```
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
sys
xeno_assignment
Tables
Views
Stored Procedures
Functions
Query 1
47 •   SELECT * FROM customers LIMIT 5;
48 •   SELECT * FROM customers WHERE signup_date IS NULL;
49
50 •   SELECT * FROM customers ORDER BY signup_date DESC LIMIT 10;
51
52
53
```

The bottom half of the screen shows the 'Result Grid' tab selected, displaying the results of the last query:

customer_id	full_name	email	phone	city	signup_date
459709	Lakshit Raman	aradhayashere@shukla-singhal.com	7688582689	Mumbai	2025-04-14
285490	Lakshay Lake	lcheema@sharaf.com	9040520243	Bangalore	2025-04-14
851249	Manjari Batta	navyadhirna@kant.com	8669084740	Pune	2025-04-14
340804	Sana Madan		9908406028	Bangalore	2025-04-14
633702	Tejas Sanghvi	msolanki@gmail.com	9396172907	Kolkata	2025-04-14
715973	Rasha Sastry	avirk@hotmail.com	9393236358	Ahmedabad	2025-04-12
370147	Parinaz Sheth	qbal@gmail.com	7962001833	Pune	2025-04-12
335527	Indrans Jani	bhatpihu@hotmail.com	7377085388	Ahmedabad	2025-04-11
462199	Tara Dani	cbajaj@hotmail.com	9559381066	Delhi	2025-04-11
266733	Adah Kamdar	gokul03@gmail.com	8643563127	Mumbai	2025-04-11
*	HULL	HULL	HULL	HULL	HULL

- Count the number of signups in the last 30 days. Assume today to be 16th April 2025

The screenshot shows the SSMS interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. Below the menu is a toolbar with various icons. The left pane is the Navigator, showing the SCHEMAS node expanded to show sys and xeno_assignment, with Tables selected. The main area is titled 'Query 1' and contains the following SQL code:

```
50 •   SELECT * FROM customers ORDER BY signup_date DESC LIMIT 10;
51
52      -- Count signups from 2025-03-17 through 2025-04-16 (inclusive)
53 •   SELECT COUNT(*) AS signups_last_30_days
54     FROM customers
```

The bottom half of the screen shows the 'Result Grid' tab selected, displaying the results of the last query:

signups_last_30_days
85

2.3 List unique cities where customers are based

The screenshot shows a database interface with a results grid. The grid has a single column labeled 'city'. The data rows are: Ahmedabad, Bangalore, Chennai, Delhi, Hyderabad, Kolkata, Mumbai, and Pune. The rows for Bangalore, Chennai, Delhi, Hyderabad, Kolkata, Mumbai, and Pune are highlighted in blue, while Ahmedabad is not.

city
Ahmedabad
Bangalore
Chennai
Delhi
Hyderabad
Kolkata
Mumbai
Pune

2.4 List the top 3 cities by number of signups.

The screenshot shows a database interface with a results grid. The grid has two columns: 'city' and 'signup_count'. The data rows are: Pune (44), Delhi (42), and Bangalore (41). The row for Bangalore is highlighted in blue.

SQL Query:

```
61
62 •   SELECT TRIM(city) AS city, COUNT(*) AS signup_count
63   FROM customers
64   WHERE city IS NOT NULL AND TRIM(city) <> ''
65   GROUP BY TRIM(city)
66   ORDER BY signup_count DESC
67   LIMIT 3;
```

city	signup_count
Pune	44
Delhi	42
Bangalore	41

2.5 Assume there's another table orders (customer_id, order_id, amount). How would you find customers who have never placed an order?

This SQL query makes it possible to identify those customers who have never placed an order. It uses the NOT EXISTS condition to compare the customers table against the orders table. Results will include only those customers who do not have any matching records in the orders table.

```
SELECT 1
FROM orders o
WHERE o.customer_id = c.customer_id
```

2. Data Transformation and Enrichment

1. Add a new column to show if the email domain is ‘gmail.com’ or not. Fill it with ‘Yes’ or ‘No’.

The screenshot shows the MySQL Workbench interface. At the top, there is a code editor window with three SQL statements:

```
78
79 • ALTER TABLE customers ADD COLUMN is_gmail ENUM('Yes','No') DEFAULT 'No';
80 • ALTER TABLE customers ADD COLUMN first_name VARCHAR(100);
81 • ALTER TABLE customers ADD COLUMN signup_month VARCHAR(20);
```

Below the code editor is an "Output" pane showing the results of the executed statements:

#	Time	Action	Message	Duration / Fetch
6	09:59:07	ALTER TABLE customers ADD COLUMN is_gmail ENUM('Yes','No') DEFAULT 'No'	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
7	10:00:00	ALTER TABLE customers ADD COLUMN first_name VARCHAR(100)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
8	10:00:04	ALTER TABLE customers ADD COLUMN signup_month VARCHAR(20)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.032 sec

At the bottom of the output pane, there is a "Result Grid" table:

Field	Type	Null	Key	Default	Extra
customer_id	int	NO	PRI	HULL	
full_name	varchar(200)	YES		HULL	
email	varchar(255)	YES		HULL	
phone	varchar(50)	YES		HULL	
city	varchar(100)	YES		HULL	
signup_date	date	YES		HULL	
signup_date_dt	date	YES		HULL	
is_gmail	enum('Yes','No')	YES		No	

2. Extract the first name from the name column and store it in ‘first_name’

```
89      -- Extract first name
90 • UPDATE customers
91     SET first_name = NULLIF(TRIM(SUBSTRING_INDEX(TRIM(COALESCE(full_name,'')), ' ', 1)), '');
92 • SELECT
93     customer_id,
```

The screenshot shows the MySQL Workbench interface with the results of the UPDATE query:

customer_id	full_name	first_name	email	is_gmail
459709	Lakshit Raman	Lakshit	aradhyashere@shukla-singhal.com	No
851249	Manjari Batta	Manjari	navyadchingra@kant.com	No
285490	Lakshay Loke	Lakshay	lcheema@sharaf.com	No
340804	Sana Madan	Sana		No
633702	Tejas Sanghvi	Tejas	msolanki@gmail.com	Yes

3. Add a column ‘signup_month’ to capture the month name the customer signed up.

```
108 • SELECT
```

The screenshot shows the MySQL Workbench interface with the results of the SELECT query:

customer_id	full_name	signup_date	signup_month
459709	Lakshit Raman	2025-04-14	April
851249	Manjari Batta	2025-04-14	April
285490	Lakshay Loke	2025-04-14	April
340804	Sana Madan	2025-04-14	April
633702	Tejas Sanghvi	2025-04-14	April

4. Create a report which shows the no. of GMAIL customers who have signed up for each day of the week.

```
117 •   SELECT
118     DAYNAME(signup_date) AS weekday,
119     COUNT(*) AS gmail_signups
120   FROM customers
121   WHERE is_gmail = 'Yes' AND signup_date IS NOT NULL
122   GROUP BY weekday
123   ORDER BY FIELD(weekday, 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday');
```

weekday	gmail_signups
Monday	5
Tuesday	6
Wednesday	4
Thursday	1
Friday	9

5. Create a new table ‘vip_customers’ with customers from Delhi, Mumbai and Bangalore who have signed up in the last 60 days from 16th April 2025.

```
130      AND signup_date BETWEEN DATE_SUB('2025-04-16', INTERVAL 60 DAY) AND '2025-04-16';
131
132 •   SELECT * FROM vip_customers ORDER BY signup_date DESC;
```

customer_id	full_name	email	phone	city	signup_date	signup_date_dt	is_c
285490	Lakshay Loke	lcheema@sharaf.com	9040520243	Bangalore	2025-04-14	NULL	No
340804	Sana Madan		9908406028	Bangalore	2025-04-14	NULL	No
459709	Lakshit Raman	aradhyashere@shukla-singhal.com	7688582689	Mumbai	2025-04-14	NULL	No
266733	Adah Kamdar	gokul03@gmail.com	8643563127	Mumbai	2025-04-11	NULL	Yes

3. Analytics & Reporting

1. Show a monthly signup count for the past 6 months.

```
174 •   SELECT m.month_label, m.signups
175   FROM (
176     SELECT DATE_FORMAT(signup_date, '%Y-%m') AS month_label,
177           COUNT(*) AS signups
178     FROM customers
179     WHERE signup_date BETWEEN DATE_SUB('2025-04-16', INTERVAL 6 MONTH) AND '2025-04-16'
180     GROUP BY DATE_FORMAT(signup_date, '%Y-%m')
```

month_label	signups
2024-10	6
2024-11	7
2024-12	6
2025-01	8
2025-02	7

2. Get a list of cities with more than 20 customers.

```
135 •   SELECT TRIM(city) AS city, COUNT(*) AS customers_count  
136     FROM customers  
137     WHERE city IS NOT NULL AND TRIM(city) <> ''  
138     GROUP BY TRIM(city)
```

Result Grid		
	city	customers_count
▶	Pune	44
	Delhi	42
	Bangalore	41
	Kolkata	41
	Hyderabad	39

3. Find the date with the highest number of signups.

```
141  
142 •   SELECT signup_date, COUNT(*) AS signups  
143     FROM customers
```

Result Grid		
	signup_date	signups
▶	2025-04-14	5

4. Add a new column to show the day of the signup date. Find the day with the highest number of signups.

```
154  
155 •   SELECT signup_day, COUNT(*) AS signups
```

Result Grid		
	signup_day	signups
▶	Friday	51