SQL class 3:

While loading the data set, make sure to set the date columns as datetime and remove time from the format option.

Topics covered in today's class

1) **Aggregation**
2) **GROUP BY**
3) **HAVING Clause**
4) **Conditional Statements**
5) **CASE**

Aggregating Data

1. Exercise: Find the average monthly rate for each service type in service_packages. Introduce the ROUND function here to make result set neater

2. Exercise: Identify the customer who has used the most data in a single service_usage record. (covers ORDER BY and LIMIT that we did in last class)

3. Exercise: Calculate the total minutes used by all customers for mobile services.

4. Exercise: List the total number of feedback entries for each rating level.

The result set will show an empty record too because that is NULL

5. Exercise: Show the total amount due by each customer, but only for those who have a total amount greater than $100.(introducing HAVING clause here) Using billing table here

7. Exercise: Group feedback by service impacted and rating to count the number of feedback entries.

8. Exercise: Calculate the total data and minutes used per customer, per service type.

Conditional Expressions and CASE Statements

1. Exercise: Categorize customers based on their subscription date: ‘New’ for those subscribed after 2023-01-01, ‘Old’ for all others.

2. Exercise: Provide a summary of each customer’s billing status, showing ‘Paid’ if the payment_date is not null, and ‘Unpaid’ otherwise.

4. Exercise: In service_usage, label data usage as ‘High’ if above the average usage, ‘Low’ if below.

5. Exercise: For each feedback given, categorise the service_impacted into ‘Digital’ for ‘streaming’ or ‘broadband’ and ‘Voice’ for ‘mobile’.

6. Exercise: Update the discounts_applied field in billing to 10% of amount_due for bills with a payment_date past the due_date, otherwise set it to 5%.

7. Exercise: Classify each customer as ‘High Value’ if they have a total amount due greater than $500, or ‘Standard Value’ if not.

8. Exercise: Mark each feedback entry as ‘Urgent’ if the rating is 1 and the feedback text includes ‘outage’ or ‘down’.

9. Exercise: In billing, create a flag for each bill that is ‘Late’ if the payment_date is after the due_date, ‘On-Time’ if it’s the same, and ‘Early’ if before.