**MySQL Queries which Were used in the Churn Data Analysis**

CREATE SCHEMA customerchurn;

USE customerchurn;

CREATE TABLE customer\_churn\_new #Table creation using create table command

(Customer\_ID VARCHAR(255),

Gender VARCHAR(255),

Age INT,

Married VARCHAR(255),

Number\_of\_Dependents INT,

City VARCHAR(255),

Number\_of\_Referrals INT,

Tenure\_in\_Months INT,

Offer VARCHAR(255),

Phone\_Service VARCHAR(255),

Avg\_Monthly\_Long\_Distance\_Charges DOUBLE,

Multiple\_Lines VARCHAR(255),

Internet\_Service VARCHAR(255),

Internet\_Type VARCHAR(255),

Online\_Security VARCHAR(255),

Online\_Backup VARCHAR(255),

Device\_Protection\_Plan VARCHAR(255),

Premium\_Tech\_Support VARCHAR(255),

Streaming\_TV VARCHAR(255),

Streaming\_Movies VARCHAR(255),

Streaming\_Music VARCHAR(255),

Unlimited\_Data VARCHAR(255),

Contract VARCHAR(255),

Paperless\_Billing VARCHAR(255),

Payment\_Method VARCHAR(255),

Monthly\_Charge DOUBLE,

Total\_Charges DOUBLE,

Total\_Refunds INT,

Total\_Extra\_Data\_Charges INT,

Total\_Long\_Distance\_Charges DOUBLE,

Total\_Revenue DOUBLE,

Customer\_Status VARCHAR(255),

Churn\_Category VARCHAR(255),

Churn\_Reason VARCHAR(255)

);

LOAD DATA INFILE'customer\_churn new.csv' #Loading data by Infile command

INTO TABLE customer\_churn\_new

FIELDS TERMINATED BY ','

ENCLOSED BY '"'

LINES TERMINATED BY '\r\n'

IGNORE 1 ROWS;

SELECT \* FROM customer\_churn\_new;

# Identification of customers with high total charges who have churned

SET @row\_count = (SELECT COUNT(\*) FROM customer\_churn\_new WHERE Customer\_Status='Churned'); # Calculate the total number of churned customers

SET @row\_num = 0; # Initialize a row number variable

CREATE TEMPORARY TABLE ordered\_customers AS # Create a temporary table with ordered churned customers and row numbers

SELECT Customer\_ID, Total\_Charges, Customer\_Status, (@row\_num := @row\_num + 1) AS rownumber

FROM customer\_churn\_new

WHERE Customer\_Status = 'Churned'

ORDER BY Total\_Charges;

SELECT Customer\_ID, Total\_Charges, Customer\_Status # Filter the result to get customers with total charges greater than the 75th percentile

FROM ordered\_customers

WHERE rownumber > 0.75 \* @row\_count;

#Calculation of the total charges distribution for churned and non-churned customers

SELECT # Basic Statistics for Total Charges

Customer\_Status,

COUNT(\*) AS Total\_Customers,

AVG(Total\_Charges) AS Average\_Total\_Charges,

MIN(Total\_Charges) AS Minimum\_Total\_Charges,

MAX(Total\_Charges) AS Maximum\_Total\_Charges,

STDDEV(Total\_Charges) AS stddev\_total\_charges

FROM customer\_churn\_new

GROUP BY Customer\_Status;

SELECT # Distribution by Charge Range

Customer\_Status,

CASE

WHEN Total\_Charges < 50 THEN '0-50'

WHEN Total\_Charges < 100 THEN '50-100'

WHEN Total\_Charges < 150 THEN '100-150'

WHEN Total\_Charges < 200 THEN '150-200'

WHEN Total\_Charges < 250 THEN '200-250'

WHEN Total\_Charges < 300 THEN '250-300'

ELSE '300+'

END AS Charge\_Range,

COUNT(\*) AS Total\_Customers\_In\_Range

FROM customer\_churn\_new

GROUP BY Customer\_Status, Charge\_Range

ORDER BY Customer\_Status, Charge\_Range;

# Identification of the average total charges for customers grouped by gender and marital status

SELECT

Gender,

Married,

AVG(Total\_Charges) AS Average\_Charges

FROM

customer\_churn\_new

GROUP BY Gender,Married;

# Calculation of the average monthly charges for different age groups among churned customers

SELECT

CASE

WHEN Age BETWEEN 0 AND 20 THEN '0-20'

WHEN Age BETWEEN 21 AND 30 THEN '21-30'

WHEN Age BETWEEN 31 AND 40 THEN '31-40'

WHEN Age BETWEEN 41 AND 50 THEN '41-50'

WHEN Age BETWEEN 51 AND 60 THEN '51-60'

WHEN Age BETWEEN 61 AND 70 THEN '61-70'

WHEN Age BETWEEN 71 AND 80 THEN '71-80'

ELSE '81-100'

END AS Age\_Group,

AVG(Monthly\_Charge) AS Average\_Monthly\_Charge

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned'

GROUP BY

Age\_Group

ORDER BY

Age\_Group;

# Determination of the average age and total charges for customers with multiple lines and online backup

SELECT

AVG(Age) AS Average\_age,

SUM(Total\_Charges) AS Total\_charges

FROM

customer\_churn\_new

WHERE

Multiple\_Lines = 'Yes'

AND Online\_Backup = 'Yes';

# Identification of the contract types with the highest churn rate among senior citizens (age 65 and over)

SELECT

Contract,

COUNT(CASE WHEN Customer\_Status = 'Churned' THEN 1 END) \* 1.0 / COUNT(\*) AS churn\_rate #calculates the churn rate as the number of churned customers divided by the total number of customers for each contract type.

FROM

customer\_churn\_new

WHERE

age >= 65

GROUP BY

Contract

ORDER BY

churn\_rate DESC;

# Calculation of the average monthly charges for customers who have multiple lines and streaming TV

SELECT

AVG(Monthly\_Charge) AS Average\_Monthly\_Charge

FROM customer\_churn\_new

WHERE Multiple\_Lines = 'Yes'

AND Streaming\_TV = 'Yes';

# Identification of the customers who have churned and used the most online services

SELECT # Filter Churned Customers and Count Online Services

Customer\_ID,

(CASE WHEN Online\_Security = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Online\_Backup = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Device\_Protection\_Plan = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Premium\_Tech\_Support = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Streaming\_TV = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Streaming\_Movies = 'Yes' THEN 1 ELSE 0 END) AS Online\_Services\_Count

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned';

SELECT # Identify the Maximum Number of Online Services Used

MAX(Online\_Services\_Count) AS Max\_Online\_Services

FROM

(SELECT

Customer\_ID,

(CASE WHEN Online\_Security = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Online\_Backup = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Device\_Protection\_Plan = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Premium\_Tech\_Support = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Streaming\_TV = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Streaming\_Movies = 'Yes' THEN 1 ELSE 0 END) AS Online\_Services\_Count

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned') AS Subquery;

SELECT # Find Customers with the Maximum Online Services Count

Customer\_ID

FROM

(SELECT

Customer\_ID,

(CASE WHEN Online\_Security = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Online\_Backup = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Device\_Protection\_Plan= 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Premium\_Tech\_Support = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Streaming\_TV = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Streaming\_Movies = 'Yes' THEN 1 ELSE 0 END) AS Online\_Services\_Count

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned') AS Subquery

WHERE

Online\_Services\_Count = (SELECT

MAX(Online\_Services\_Count)

FROM

(SELECT

Customer\_ID,

(CASE WHEN Online\_Security = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Online\_Backup = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Device\_Protection\_Plan = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Premium\_Tech\_Support = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Streaming\_TV = 'Yes' THEN 1 ELSE 0 END +

CASE WHEN Streaming\_Movies = 'Yes' THEN 1 ELSE 0 END) AS online\_services\_count

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned') AS Max\_Subquery);

# Identification of the gender distribution among customers who have churned and are on yearly contracts

SELECT

Gender,

COUNT(\*) AS Count

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned'

AND Contract = 'One\_Year'

GROUP BY

Gender;

# Calculation of the average monthly charges and total charges for customers who have churned, grouped by contract type and internet service type

SELECT

Contract,

Internet\_Type,

AVG(Monthly\_Charge) AS Average\_Monthly\_Charge,

SUM(Total\_Charges) AS Total\_Charges

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned'

GROUP BY

Contract, Internet\_Type;

# To Find the customers who have churned and are not using online services, and their average total charges

SELECT

AVG(Total\_Charges) AS Average\_Total\_Charges

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned'

AND Online\_Security = 'No'

AND Online\_Backup = 'No'

AND Device\_Protection\_Plan = 'No'

AND Premium\_Tech\_Support = 'No'

AND Streaming\_TV = 'No'

AND Streaming\_Movies = 'No'

AND Streaming\_Music = 'No';

# Calculation of the average monthly charges and total charges for customers who have churned, grouped by the number of dependents

SELECT

Number\_of\_Dependents,

AVG(Monthly\_charge) AS Average\_Monthly\_Charge,

SUM(Total\_Charges) AS Total\_Charges

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned'

GROUP BY

Number\_of\_Dependents;

# Identification of the customers who have churned, and their contract duration in months (for monthly contracts)

SELECT

Customer\_ID,

Tenure\_in\_Months,

Customer\_Status

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned'

AND Contract = 'Month-to-Month';

# Determination of the average age and total charges for customers who have churned, grouped by internet service and phone service

SELECT

Internet\_Service,

Phone\_Service,

AVG(Age) AS Average\_Age,

AVG(Total\_Charges) AS Average\_Total\_Charges

FROM

customer\_churn\_new

WHERE

Customer\_Status = 'Churned'

AND Internet\_Service = 'Yes'

AND Phone\_Service ='Yes'

GROUP BY

Internet\_Service,Phone\_Service;