

SQL ETL CODE SNIPPET

-- This statement creates a new database named 'db_Churn'

```
CREATE DATABASE db_Churn;
```

-- This query checks distinct values and their counts for Gender

```
SELECT Gender, Count(Gender) AS TotalCount,  
       Count(Gender) * 100.0 / (SELECT Count(*) FROM stg_Churn) AS Percentage  
FROM stg_Churn  
GROUP BY Gender;
```

-- This query checks distinct values and their counts for Contract

```
SELECT Contract, Count(Contract) AS TotalCount,  
       Count(Contract) * 100.0 / (SELECT Count(*) FROM stg_Churn) AS Percentage  
FROM stg_Churn  
GROUP BY Contract;
```

-- This query checks distinct values and their counts, along with Total Revenue for Customer_Status

```
SELECT Customer_Status, Count(Customer_Status) AS TotalCount,  
       Sum(Total_Revenue) AS TotalRev,  
       Sum(Total_Revenue) * 100.0 / (SELECT Sum(Total_Revenue) FROM stg_Churn) * 100  
AS RevPercentage  
FROM stg_Churn  
GROUP BY Customer_Status;
```

-- This query checks distinct values and their counts for State, ordered by Percentage descending

```
SELECT State, Count(State) AS TotalCount,  
       Count(State) * 100.0 / (SELECT Count(*) FROM stg_Churn) AS Percentage  
FROM stg_Churn  
GROUP BY State  
ORDER BY Percentage DESC;
```

-- Data Exploration: Checking the NULL values in various columns

```
SELECT  
    SUM(CASE WHEN Customer_ID IS NULL THEN 1 ELSE 0 END) AS  
Customer_ID_Null_Count,  
    SUM(CASE WHEN Gender IS NULL THEN 1 ELSE 0 END) AS Gender_Null_Count,  
    SUM(CASE WHEN Age IS NULL THEN 1 ELSE 0 END) AS Age_Null_Count,  
    SUM(CASE WHEN Married IS NULL THEN 1 ELSE 0 END) AS Married_Null_Count,  
    SUM(CASE WHEN State IS NULL THEN 1 ELSE 0 END) AS State_Null_Count,  
    SUM(CASE WHEN Number_of_Referrals IS NULL THEN 1 ELSE 0 END) AS  
Number_of_Referrals_Null_Count,  
    SUM(CASE WHEN Tenure_in_Months IS NULL THEN 1 ELSE 0 END) AS  
Tenure_in_Months_Null_Count,  
    SUM(CASE WHEN Value_Deal IS NULL THEN 1 ELSE 0 END) AS  
Value_Deal_Null_Count,  
    SUM(CASE WHEN Phone_Service IS NULL THEN 1 ELSE 0 END) AS  
Phone_Service_Null_Count,  
    SUM(CASE WHEN Multiple_Lines IS NULL THEN 1 ELSE 0 END) AS  
Multiple_Lines_Null_Count,
```

SUM(CASE WHEN Internet_Service IS NULL THEN 1 ELSE 0 END) AS
Internet_Service_Null_Count,

SUM(CASE WHEN Internet_Type IS NULL THEN 1 ELSE 0 END) AS
Internet_Type_Null_Count,

SUM(CASE WHEN Online_Security IS NULL THEN 1 ELSE 0 END) AS
Online_Security_Null_Count,

SUM(CASE WHEN Online_Backup IS NULL THEN 1 ELSE 0 END) AS
Online_Backup_Null_Count,

SUM(CASE WHEN Device_Protection_Plan IS NULL THEN 1 ELSE 0 END) AS
Device_Protection_Plan_Null_Count,

SUM(CASE WHEN Premium_Support IS NULL THEN 1 ELSE 0 END) AS
Premium_Support_Null_Count,

SUM(CASE WHEN Streaming_TV IS NULL THEN 1 ELSE 0 END) AS
Streaming_TV_Null_Count,

SUM(CASE WHEN Streaming_Movies IS NULL THEN 1 ELSE 0 END) AS
Streaming_Movies_Null_Count,

SUM(CASE WHEN Streaming_Music IS NULL THEN 1 ELSE 0 END) AS
Streaming_Music_Null_Count,

SUM(CASE WHEN Unlimited_Data IS NULL THEN 1 ELSE 0 END) AS
Unlimited_Data_Null_Count,

SUM(CASE WHEN Contract IS NULL THEN 1 ELSE 0 END) AS Contract_Null_Count,

SUM(CASE WHEN Paperless_Billing IS NULL THEN 1 ELSE 0 END) AS
Paperless_Billing_Null_Count,

SUM(CASE WHEN Payment_Method IS NULL THEN 1 ELSE 0 END) AS
Payment_Method_Null_Count,

SUM(CASE WHEN Monthly_Charge IS NULL THEN 1 ELSE 0 END) AS
Monthly_Charge_Null_Count,

SUM(CASE WHEN Total_Charges IS NULL THEN 1 ELSE 0 END) AS
Total_Charges_Null_Count,

SUM(CASE WHEN Total_Refunds IS NULL THEN 1 ELSE 0 END) AS
Total_Refunds_Null_Count,

SUM(CASE WHEN Total_Extra_Data_Charges IS NULL THEN 1 ELSE 0 END) AS
Total_Extra_Data_Charges_Null_Count,

```

SUM(CASE WHEN Total_Long_Distance_Charges IS NULL THEN 1 ELSE 0 END) AS
Total_Long_Distance_Charges_Null_Count,

SUM(CASE WHEN Total_Revenue IS NULL THEN 1 ELSE 0 END) AS
Total_Revenue_Null_Count,

SUM(CASE WHEN Customer_Status IS NULL THEN 1 ELSE 0 END) AS
Customer_Status_Null_Count,

SUM(CASE WHEN Churn_Category IS NULL THEN 1 ELSE 0 END) AS
Churn_Category_Null_Count,

SUM(CASE WHEN Churn_Reason IS NULL THEN 1 ELSE 0 END) AS
Churn_Reason_Null_Count

FROM stg_Churn;

```

-- Remove NULL values and insert the new data into the Prod table

```

SELECT
    Customer_ID,
    Gender,
    Age,
    Married,
    State,
    Number_of_Referrals,
    Tenure_in_Months,
    ISNULL(Value_Deal, 'None') AS Value_Deal,
    Phone_Service,
    ISNULL(Multiple_Lines, 'No') AS Multiple_Lines,
    Internet_Service,
    ISNULL(Internet_Type, 'None') AS Internet_Type,
    ISNULL(Online_Security, 'No') AS Online_Security,

```

```
ISNULL(Online_Backup, 'No') AS Online_Backup,
ISNULL(Device_Protection_Plan, 'No') AS Device_Protection_Plan,
ISNULL(Premium_Support, 'No') AS Premium_Support,
ISNULL(Streaming_TV, 'No') AS Streaming_TV,
ISNULL(Streaming_Movies, 'No') AS Streaming_Movies,
ISNULL(Streaming_Music, 'No') AS Streaming_Music,
ISNULL(Unlimited_Data, 'No') AS Unlimited_Data,
Contract,
Paperless_Billing,
Payment_Method,
Monthly_Charge,
Total_Charges,
Total_Refunds,
Total_Extra_Data_Charges,
Total_Long_Distance_Charges,
Total_Revenue,
Customer_Status,
ISNULL(Churn_Category, 'Others') AS Churn_Category,
ISNULL(Churn_Reason, 'Others') AS Churn_Reason
INTO [db_Churn].[dbo].[prod_Churn]
FROM [db_Churn].[dbo].[stg_Churn];
```

-- Create view for Power BI: vw_ChurnData

```
CREATE VIEW vw_ChurnData AS  
  
    SELECT *  
  
    FROM prod_Churn  
  
    WHERE Customer_Status IN ('Churned', 'Stayed');  
  
GO
```

-- Create view for Power BI: vw_JoinData

```
CREATE VIEW vw_JoinData AS  
  
    SELECT *  
  
    FROM prod_Churn  
  
    WHERE Customer_Status = 'Joined';  
  
GO
```