

# **MySQL Capstone Project**

## **E-Commerce Customer Churn Analysis**

### **Project Steps and Objectives:**

#### **1) Data Cleaning :-**

- => Handling Missing & Outliers
- => Dealing with Inconsistencies

#### **2) Data Transformation :-**

- => Column Renaming
- => Creating New Columns
- => Column Dropping

#### **3) Data Exploration and Analysis :-**

- => Retrieving
- => Data Exploring
- => Calculating
- => Creating New Table
- => Analysing etc.,

#### **4) Key Insights for E-commerce Project :-**

- => Customer Churn Analysis
- => Customer Behaviour, Performance and Preferences
- => Demographic Insights
- => Business Recommendations

## 1) Data Cleaning:-

### Handling Missing Values & Outliers:

**Question 1 :** Imputing Mean (or) average and Rounding off for the following columns

- a) WarehouseToHome
- b) HourSpendOnApp
- c) OrderAmountHikeFromlastYear
- d) DaySinceLastOrder

#### **a) WarehouseToHome (average) :-**

Turning safe updates off : -

```
set sql_safe_updates=0;
```

Creating variable to WarehouseToHome mean

```
set @WarehouseToHome_average = (select round  
(avg(WarehouseToHome)) from customer_churn);
```

**Average Value is : 16**

-- Imputing variable to missing values in the WarehouseToHome column--

```
Update customer_churn  
set WarehouseToHome=@WarehouseToHome_average  
where WarehouseToHome IS NULL;  
SELECT WarehouseToHome FROM customer_churn;
```

The variable is successfully imputed as average value where the value is "Null"

#### **B) HourSpendOnApp (mean/average) :-**

Creating variable to HourSpendOnApp mean

```
set @HourSpendOnApp= (select round(avg(HourSpendOnApp))  
from customer_churn);
```

**Average Value is : 3**

Imputing variable to missing values in the HourSpendOnApp column

```
Update customer_churn  
set HourSpendOnApp= @HourSpendOnApp  
where HourSpendOnApp IS NULL;
```

The variable is successfully imputed as average value where the value is "Null"

### C ) OrderAmountHikeFromlastYear (Mean/Average) :-

Creating variable to OrderAmountHikeFromlastYear mean :-

```
set @OrderAmountHikeFromlastYear = (select round(avg(OrderAmountHikeFromlastYear))
                                     from customer_churn);
```

Average Value is : 16

Imputing variable to missing values in the OrderAmountHikeFromlastYear column--

```
update customer_churn
set OrderAmountHikeFromlastYear= @OrderAmountHikeFromlastYear
where OrderAmountHikeFromlastYear IS NULL;
```

The variable is successfully imputed as average value where the value is "Null"

### D) DaySinceLastOrder (Mean/Average) :-

Creating variable to DaySinceLastOrder :-

```
set @DaySinceLastOrder= (select round(avg(DaySinceLastOrder)) from
                          customer_churn);
```

Average Value is : 5

Imputing variable to missing values in the DaySinceLastOrder column :-

```
update customer_churn
set DaySinceLastOrder=@DaySinceLastOrder
where DaySinceLastOrder IS NULL;
```

The variable is successfully imputed as average value where the value is "Null"

### Question 2) Imputing mode for following columns

- a) Tenure
- b) CouponUsed
- c) OrderCount

#### a) Imputing mode for Tenure :-

Creating variable to Tenure mode :-

```
set @Tenure = (select Tenure from customer_churn
               group by Tenure order by count(*) desc
               limit 1);
```

-- Display the mode

```
select @Tenure;
```

Tenure =1

-- Impute missing values in Tenure column with the mode

```
update customer_churn
set Tenure=@Tenure
where tenure is null;
```

## **b) Imputing mode for CouponUsed**

-- Creating variable to CouponUsed mode

```
set @CouponUsed = (select CouponUsed
                    from customer_churn
                    group by CouponUsed
                    order by count(*) desc
                    limit 1);
```

-- Displaying the mode

```
select @CouponUsed;
```

-- -- Impute missing values in Coupon column with the mode

```
update Customer_churn
set CouponUsed=@CouponUsed
where CouponUsed Is Null;
```

-- c) Imputing mode for OrderCount

-- -- Creating variable to OrderCount mode

```
Set @OrderCount= (select OrderCount
                   From Customer_churn
                   group by OrderCount
                   order by count(*)
                   limit 1);
```