Creating Fact and Dimension Tables for PowerBI

Dataset is from the XGBoost Churn Prediction Model

In [1]:
```
import pandas as pd
```

In [2]:
```
# STEP 1: Load the CSV dataset
bi_churn = pd.read_csv("XGBoost_dataset_updated.csv")
```

In [4]:
```
bi_churn.head()
```

Out[4]:

	Customer_Age	Dependent_count	Months_on_book	Total_Relationship_Count	Months
0	45	3	39	5	
1	49	5	44	6	
2	51	3	36	4	
3	40	4	34	3	
4	40	3	21	5	

5 rows × 34 columns

In [7]:
```
bi_churn.columns
```

Out[7]:
```
Index(['Customer_Age', 'Dependent_count', 'Months_on_book',
       'Total_Relationship_Count', 'Months_Inactive_12_mon',
       'Contacts_Count_12_mon', 'Credit_Limit', 'Total_Revolving_Bal',
       'Total_Amt_Chng_Q4_Q1', 'Total_Trans_Amt', 'Total_Trans_Ct',
       'Total_Ct_Chng_Q4_Q1', 'Avg_Utilization_Ratio', 'Gender_M',
       'Education_Level_Doctorate', 'Education_Level_Graduate',
       'Education_Level_High School', 'Education_Level_Post-Graduate',
       'Education_Level_Uneducated', 'Education_Level_Unknown',
       'Marital_Status_Married', 'Marital_Status_Single',
       'Marital_Status_Unknown', 'Income_Category_$40K - $60K',
       'Income_Category_$60K - $80K', 'Income_Category_$80K - $120K',
       'Income_Category_Less than $40K', 'Income_Category_Unknown',
       'Card_Category_Gold', 'Card_Category_Platinum', 'Card_Category_Silver',
       'Revolving_Bal_Per_Limit', 'Avg_Transaction_Value', 'Churn'],
      dtype='object')
```

In [9]:
```
# Remove duplicates
bi_churn_cleaned = bi_churn.drop_duplicates()

#Check for nulls
null_counts = bi_churn_cleaned.isnull().sum()
print("Null counts per column:\n", null_counts)

# Confirm data types
print("\nData types:\n", bi_churn_cleaned.dtypes)
```

```
# STEP 4: Preview cleaned data
bi_churn_cleaned.head()
```

```
Null counts per column:
 Customer_Age                      0
Dependent_count                    0
Months_on_book                     0
Total_Relationship_Count           0
Months_Inactive_12_mon             0
Contacts_Count_12_mon              0
Credit_Limit                       0
Total_Revolving_Bal                0
Total_Amt_Chng_Q4_Q1               0
Total_Trans_Amt                    0
Total_Trans_Ct                     0
Total_Ct_Chng_Q4_Q1                0
Avg_Utilization_Ratio              0
Gender_M                           0
Education_Level_Doctorate          0
Education_Level_Graduate           0
Education_Level_High School        0
Education_Level_Post-Graduate      0
Education_Level_Uneducated         0
Education_Level_Unknown            0
Marital_Status_Married             0
Marital_Status_Single              0
Marital_Status_Unknown             0
Income_Category_$40K - $60K        0
Income_Category_$60K - $80K        0
Income_Category_$80K - $120K       0
Income_Category_Less than $40K     0
Income_Category_Unknown            0
Card_Category_Gold                 0
Card_Category_Platinum             0
Card_Category_Silver               0
Revolving_Bal_Per_Limit            0
Avg_Transaction_Value              0
Churn                              0
dtype: int64

Data types:
 Customer_Age                       int64
Dependent_count                     int64
Months_on_book                      int64
Total_Relationship_Count            int64
Months_Inactive_12_mon              int64
Contacts_Count_12_mon               int64
Credit_Limit                      float64
Total_Revolving_Bal                 int64
Total_Amt_Chng_Q4_Q1              float64
Total_Trans_Amt                     int64
Total_Trans_Ct                      int64
Total_Ct_Chng_Q4_Q1               float64
Avg_Utilization_Ratio             float64
Gender_M                             bool
Education_Level_Doctorate            bool
Education_Level_Graduate             bool
Education_Level_High School          bool
Education_Level_Post-Graduate        bool
Education_Level_Uneducated           bool
Education_Level_Unknown              bool
Marital_Status_Married               bool
Marital_Status_Single                bool
```

```
Marital_Status_Unknown              bool
Income_Category_$40K - $60K         bool
Income_Category_$60K - $80K         bool
Income_Category_$80K - $120K        bool
Income_Category_Less than $40K      bool
Income_Category_Unknown             bool
Card_Category_Gold                  bool
Card_Category_Platinum              bool
Card_Category_Silver                bool
Revolving_Bal_Per_Limit          float64
Avg_Transaction_Value            float64
Churn                              int64
dtype: object
```

Out[9]:

	Customer_Age	Dependent_count	Months_on_book	Total_Relationship_Count	Months
0	45	3	39	5	
1	49	5	44	6	
2	51	3	36	4	
3	40	4	34	3	
4	40	3	21	5	

5 rows × 34 columns

In [ ]:

Creating Dimension Tables

In [11]:
```
# Step 1: Reconstruct label columns from one-hot
def reverse_one_hot(df, prefix):
    cols = [col for col in df.columns if col.startswith(prefix)]
    return df[cols].idxmax(axis=1).str.replace(prefix, '').str.strip('_')

bi_churn_cleaned['Income_Category'] = reverse_one_hot(bi_churn_cleaned, "Income_
bi_churn_cleaned['Card_Category'] = reverse_one_hot(bi_churn_cleaned, "Card_Cate
bi_churn_cleaned['Marital_Status'] = reverse_one_hot(bi_churn_cleaned, "Marital_
bi_churn_cleaned['Education_Level'] = reverse_one_hot(bi_churn_cleaned, "Educati
```

In [13]:
```
# Step 2: Create dimension tables
def create_dimension(df, column_name, id_name):
    dim = df[[column_name]].drop_duplicates().reset_index(drop=True)
    dim[id_name] = range(1, len(dim) + 1)
    return dim

dim_income = create_dimension(bi_churn_cleaned, "Income_Category", "Income_Categ
dim_card = create_dimension(bi_churn_cleaned, "Card_Category", "Card_Category_ID
dim_marital = create_dimension(bi_churn_cleaned, "Marital_Status", "Marital_Stat
dim_education = create_dimension(bi_churn_cleaned, "Education_Level", "Education
```

In [15]:
```
# Step 3: Merge dimension IDs into the fact table
fact_df = bi_churn_cleaned.merge(dim_income, on="Income_Category", how="left") \
                          .merge(dim_card, on="Card_Category", how="left") \
```

```
                            .merge(dim_marital, on="Marital_Status", how="left") \
                            .merge(dim_education, on="Education_Level", how="left"
```

In [17]:
```
# Step 4: Drop one-hot and label columns
columns_to_drop = [col for col in bi_churn_cleaned.columns if
                   col.startswith("Income_Category_") or
                   col.startswith("Card_Category_") or
                   col.startswith("Marital_Status_") or
                   col.startswith("Education_Level_")] + \
                   ['Income_Category', 'Card_Category', 'Marital_Status', 'Educa
```

In [19]:
```
fact_customer_transactions = fact_df.drop(columns=columns_to_drop)
```

In [21]:
```
fact_df.head()
```

Out[21]:

	Customer_Age	Dependent_count	Months_on_book	Total_Relationship_Count	Months
0	45	3	39	5	
1	49	5	44	6	
2	51	3	36	4	
3	40	4	34	3	
4	40	3	21	5	

5 rows × 42 columns

In [26]:
```
# Step 5: Save to CSV for Power BI
fact_customer_transactions.to_csv("fact_customer_transactions.csv", index=False)
dim_income.to_csv("dim_income.csv", index=False)
dim_card.to_csv("dim_card.csv", index=False)
dim_marital.to_csv("dim_marital.csv", index=False)
dim_education.to_csv("dim_education.csv", index=False)
```

In [ ]:
```

```