Transaction Data for fraud analysis

About DataSet

The "Fraud Analysis" dataset is designed for comprehensive fraud prevention and analysis in a bank setting. It includes a variety of data points collected and generated for the purpose of fraud detection and prescription.

Note:

- This dataset is entirely synthetic, and it does not represent real-world sales data. It is created for educational and practice purposes only.

Credits:

For the Dataset:

- Kaggle: Transaction Data for Fraud Analysis
- Dataset Link: Transaction Data for Fraud Analysis https://www.kaggle.com/datasets/ashishraut64/indian-startups-top-300/data (https://www.kaggle.com/datasets/ashishraut64/indian-startups-top-300/data)

Original Creator:

- Ishita Biswas

Workflow

- Understanding Data
- EDA (Exploratory Data Analysis)
- Insights

- Conclusion

In [5]:
```
#Importing Libraries

import warnings
warnings.filterwarnings("ignore")

import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [6]:
```
df = pd.read_csv("synthetic_financial_data.csv")
df
```

Out[6]:

	transaction_id	customer_id	merchant_id	amount	transaction_time	is_fraudulent	car
0	1	1082	2027	5758.59	2023-01-01 00:00:00	0	Mast
1	2	1015	2053	1901.56	2023-01-01 00:00:01	1	
2	3	1004	2035	1248.86	2023-01-01 00:00:02	1	Mast
3	4	1095	2037	7619.05	2023-01-01 00:00:03	1	D
4	5	1036	2083	1890.10	2023-01-01 00:00:04	1	Mast
...	...	...	...	...	...	...	
9995	9996	1056	2023	8935.28	2023-01-01 02:46:35	1	Mast
9996	9997	1053	2026	30.15	2023-01-01 02:46:36	0	Mast
9997	9998	1041	2034	6333.64	2023-01-01 02:46:37	0	An E
9998	9999	1009	2019	2837.13	2023-01-01 02:46:38	1	
9999	10000	1082	2070	7209.43	2023-01-01 02:46:39	1	D

10000 rows × 11 columns

Introduction

Financial fraud remains a critical challenge in today's digital world, impacting both businesses and individuals. To combat this ever-evolving threat, a robust understanding of fraudulent activities is essential. The "Fraud Analysis" dataset serves as a valuable resource designed to address this need.

Understanding Data

In [7]:
```
1 df.head()   #Head - returns a specified number of rows, string from the
```

Out[7]:

	transaction_id	customer_id	merchant_id	amount	transaction_time	is_fraudulent	card_ty
0	1	1082	2027	5758.59	2023-01-01 00:00:00	0	MasterC
1	2	1015	2053	1901.56	2023-01-01 00:00:01	1	V
2	3	1004	2035	1248.86	2023-01-01 00:00:02	1	MasterC
3	4	1095	2037	7619.05	2023-01-01 00:00:03	1	Disco
4	5	1036	2083	1890.10	2023-01-01 00:00:04	1	MasterC

In [8]:
```
1 df.tail()  #Tail - returns a specified number of last rows
```

Out[8]:

	transaction_id	customer_id	merchant_id	amount	transaction_time	is_fraudulent	car
9995	9996	1056	2023	8935.28	2023-01-01 02:46:35	1	Mast
9996	9997	1053	2026	30.15	2023-01-01 02:46:36	0	Mast
9997	9998	1041	2034	6333.64	2023-01-01 02:46:37	0	An E
9998	9999	1009	2019	2837.13	2023-01-01 02:46:38	1	
9999	10000	1082	2070	7209.43	2023-01-01 02:46:39	1	D

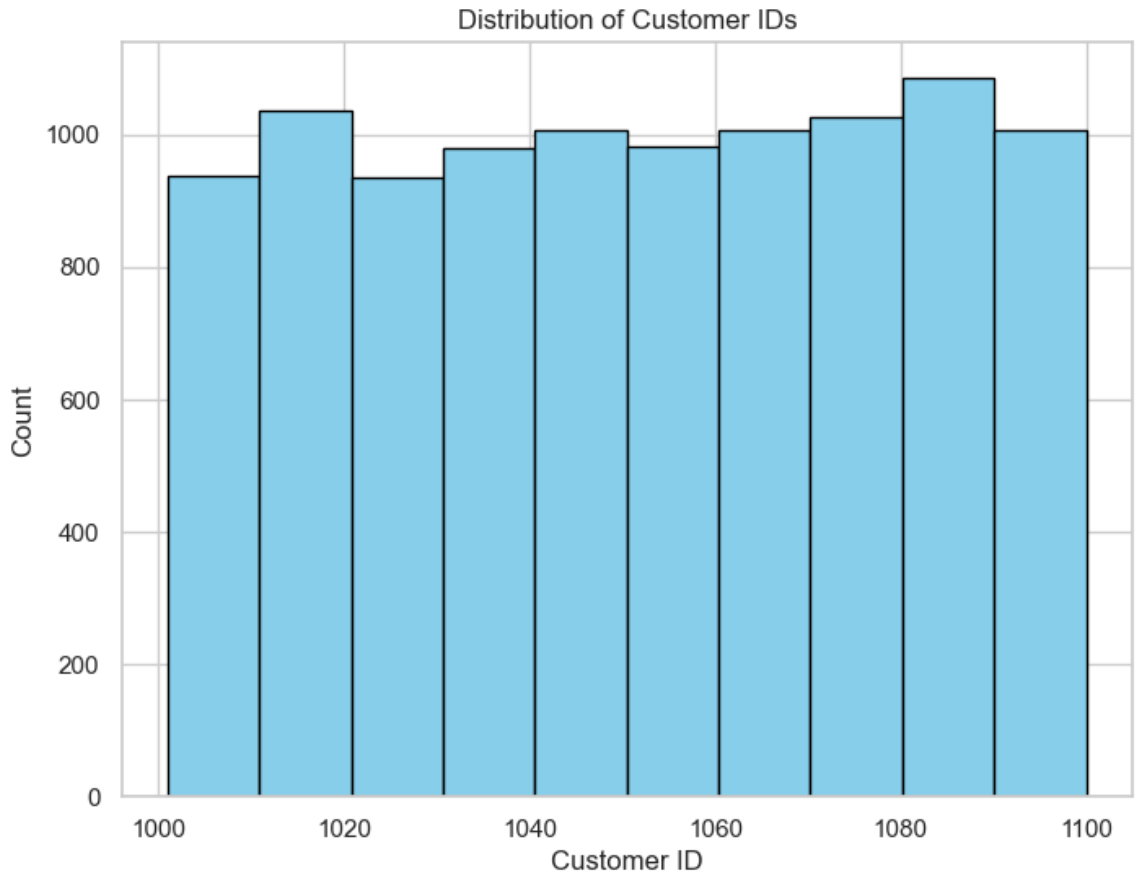
In [9]:
```
1 df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10000 entries, 0 to 9999
Data columns (total 11 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   transaction_id          10000 non-null  int64  
 1   customer_id             10000 non-null  int64  
 2   merchant_id             10000 non-null  int64  
 3   amount                  10000 non-null  float64
 4   transaction_time        10000 non-null  object 
 5   is_fraudulent           10000 non-null  int64  
 6   card_type               10000 non-null  object 
 7   location                10000 non-null  object 
 8   purchase_category       10000 non-null  object 
 9   customer_age            10000 non-null  int64  
 10  transaction_description  10000 non-null  object 
dtypes: float64(1), int64(5), object(5)
memory usage: 859.5+ KB
```

EDA - Exploratory Data Analysis

Let's see customer Ids distribution

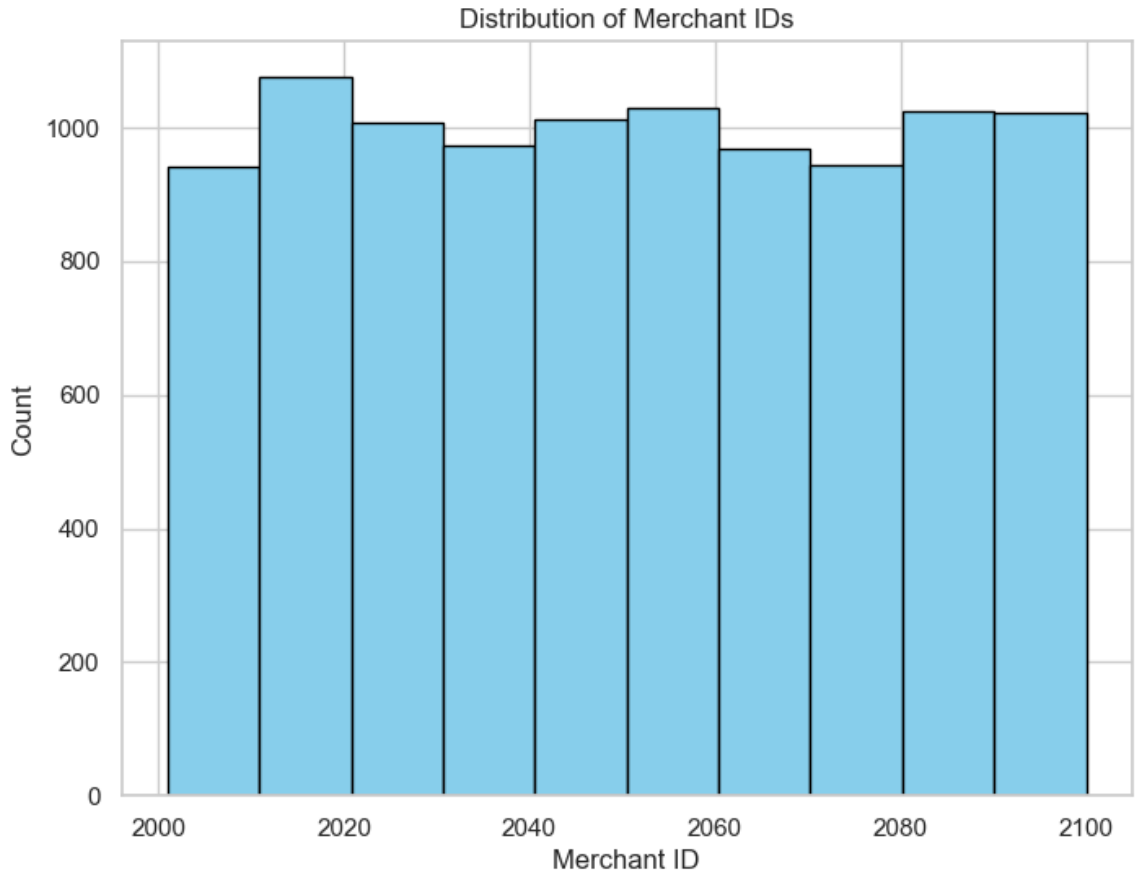
In [32]:
```
plt.figure(figsize=(8, 6))
plt.hist(data=df, x="customer_id", bins=10, color='skyblue', edgecolor=
plt.title('Distribution of Customer IDs')
plt.xlabel('Customer ID')
plt.ylabel('Count')
plt.grid(True)
```



Let's see the Merchant Ids Distribution

In [33]:

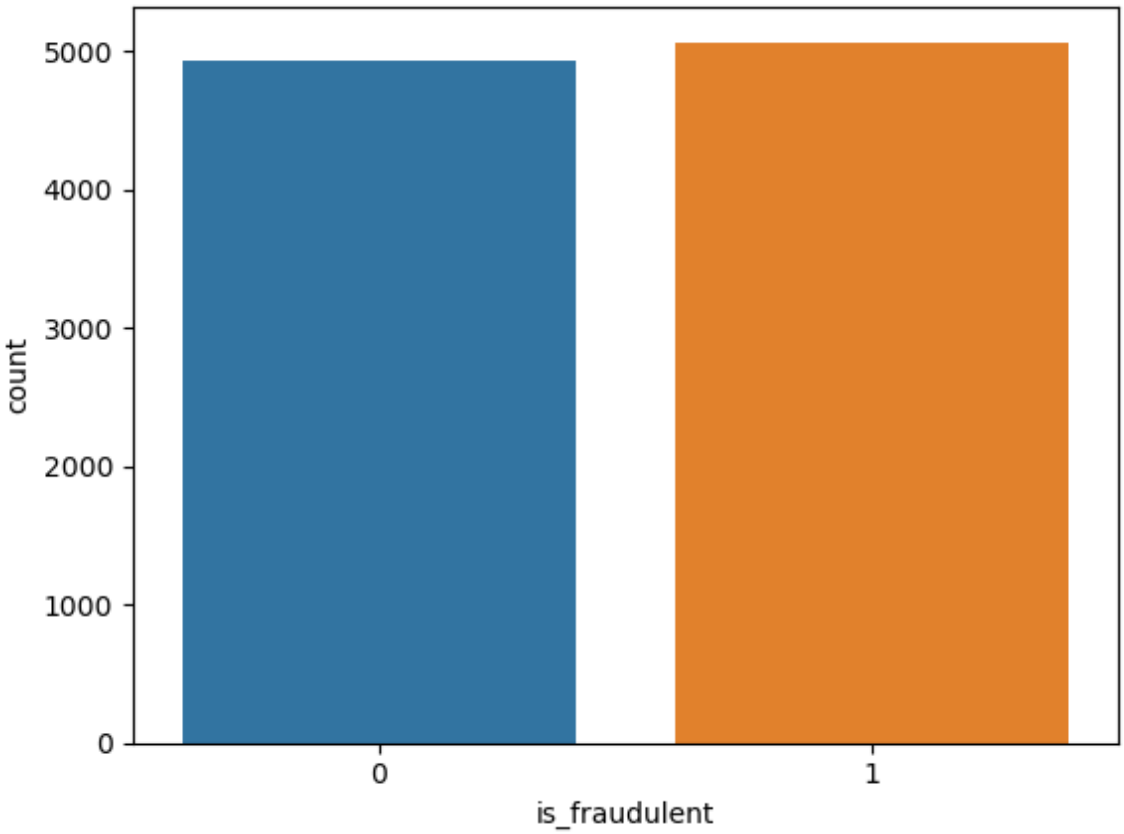
```
plt.figure(figsize=(8, 6))
plt.hist(data=df, x="merchant_id", bins=10, color='skyblue', edgecolor=
plt.title('Distribution of Merchant IDs')
plt.xlabel('Merchant ID')
plt.ylabel('Count')
plt.grid(True)
```



To Identify the fraud Transaction in this dataset. we need to know how many fraud transaction happened.

In [13]:
```
1 sns.countplot(data=df,x='is_fraudulent') # 0 (zero) = No Fraud Transact
```

Out[13]: <AxesSubplot:xlabel='is_fraudulent', ylabel='count'>



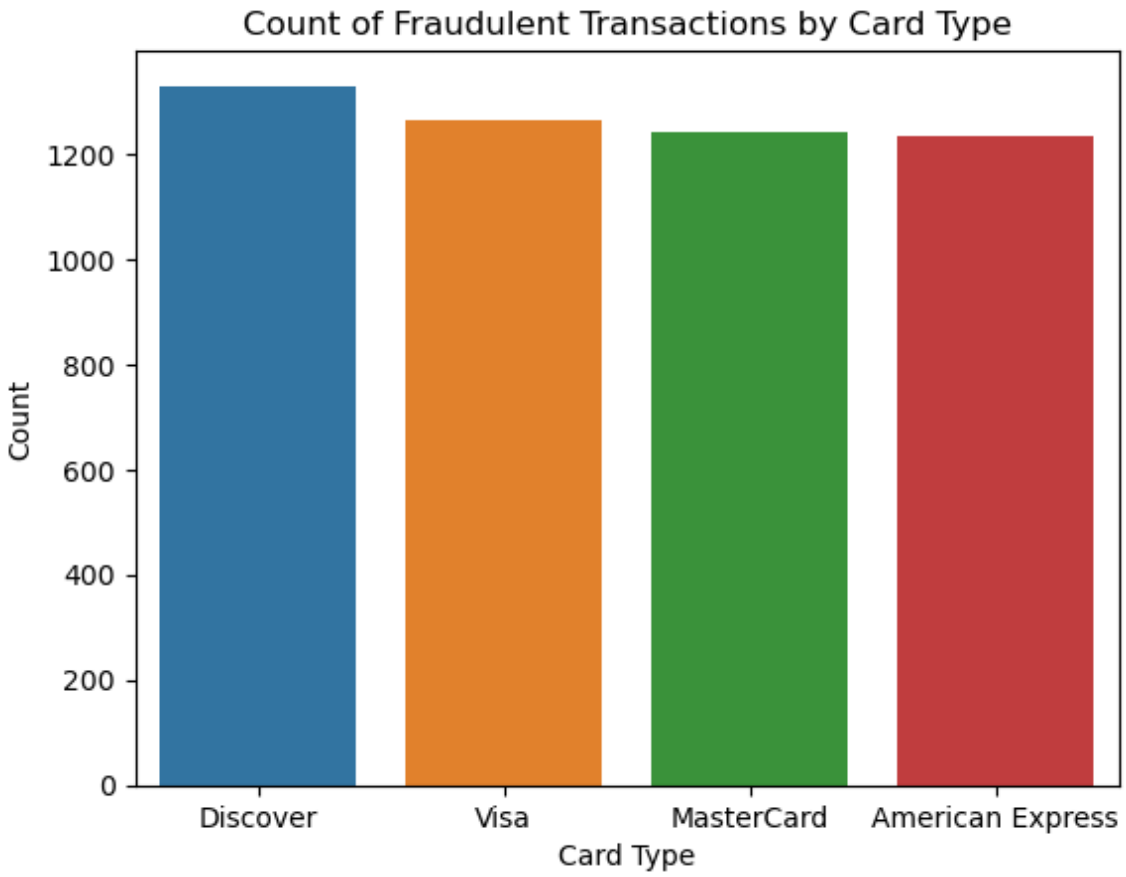
Insight:-

- as you can see the above approx 5,000 fraud transaction happened.

To track down fraud transaction, we need to know from which card type frauderster using to commit the fraud transaction.

In [18]:
```
sns.countplot(data=df[df['is_fraudulent'] == 1], x='card_type', order=d
plt.title('Count of Fraudulent Transactions by Card Type')
plt.xlabel('Card Type')
plt.ylabel('Count')
```

Out[18]: Text(0, 0.5, 'Count')



Insight:-

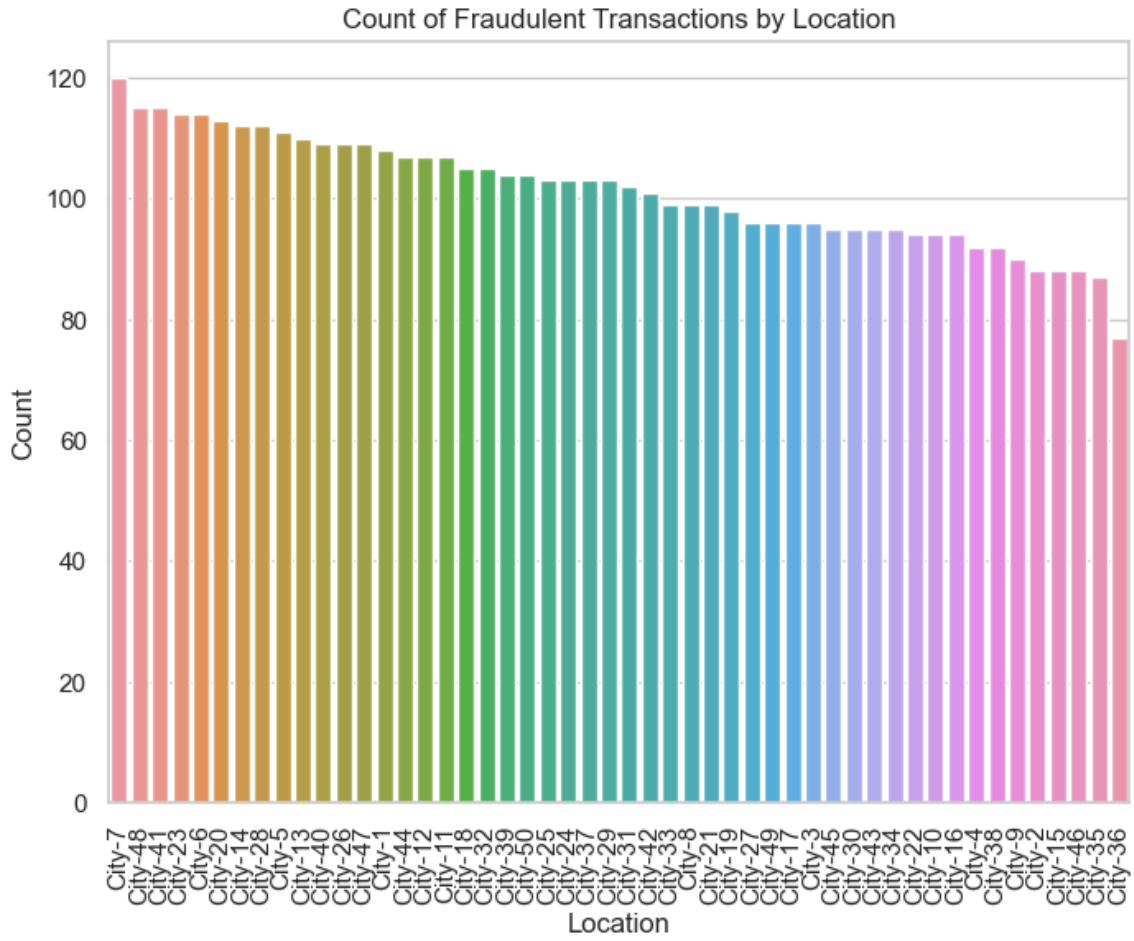
- In the above graph you're seeing the fraudster using which cards the most.
- Fraudster using Discover card type the most the number exceeds more than 1200 transaction
- second is the visa with slightly lower number than discover card
- Mastercard and American Express card are the respective the same positon as 3rd

we need to know from which city frauds are happening.

In [23]:

```
plt.figure(figsize=(8, 6))
sns.set(style="whitegrid")
sns.countplot(data=df[df['is_fraudulent'] == 1], x='location', order=df
plt.title('Count of Fraudulent Transactions by Location')
plt.xlabel('Location')
plt.ylabel('Count')
plt.xticks(rotation=90)
```

Out[23]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
        34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49]),
 [Text(0, 0, 'City-7'),
  Text(1, 0, 'City-48'),
  Text(2, 0, 'City-41'),
  Text(3, 0, 'City-23'),
  Text(4, 0, 'City-6'),
  Text(5, 0, 'City-20'),
  Text(6, 0, 'City-14'),
  Text(7, 0, 'City-28'),
  Text(8, 0, 'City-5'),
  Text(9, 0, 'City-13'),
  Text(10, 0, 'City-40'),
  Text(11, 0, 'City-26'),
  Text(12, 0, 'City-47'),
  Text(13, 0, 'City-1'),
  Text(14, 0, 'City-44'),
  Text(15, 0, 'City-12'),
  Text(16, 0, 'City-11'),
  Text(17, 0, 'City-18'),
  Text(18, 0, 'City-32'),
  Text(19, 0, 'City-39'),
  Text(20, 0, 'City-50'),
  Text(21, 0, 'City-25'),
  Text(22, 0, 'City-24'),
  Text(23, 0, 'City-37'),
  Text(24, 0, 'City-29'),
  Text(25, 0, 'City-31'),
  Text(26, 0, 'City-42'),
  Text(27, 0, 'City-33'),
  Text(28, 0, 'City-8'),
  Text(29, 0, 'City-21'),
  Text(30, 0, 'City-19'),
  Text(31, 0, 'City-27'),
  Text(32, 0, 'City-49'),
  Text(33, 0, 'City-17'),
  Text(34, 0, 'City-3'),
  Text(35, 0, 'City-45'),
  Text(36, 0, 'City-30'),
  Text(37, 0, 'City-43'),
  Text(38, 0, 'City-34'),
  Text(39, 0, 'City-22'),
  Text(40, 0, 'City-10'),
  Text(41, 0, 'City-16'),
  Text(42, 0, 'City-4'),
  Text(43, 0, 'City-38'),
  Text(44, 0, 'City-9'),
  Text(45, 0, 'City-2'),
  Text(46, 0, 'City-15'),
  Text(47, 0, 'City-46'),
  Text(48, 0, 'City-35'),
  Text(49, 0, 'City-36')])
```



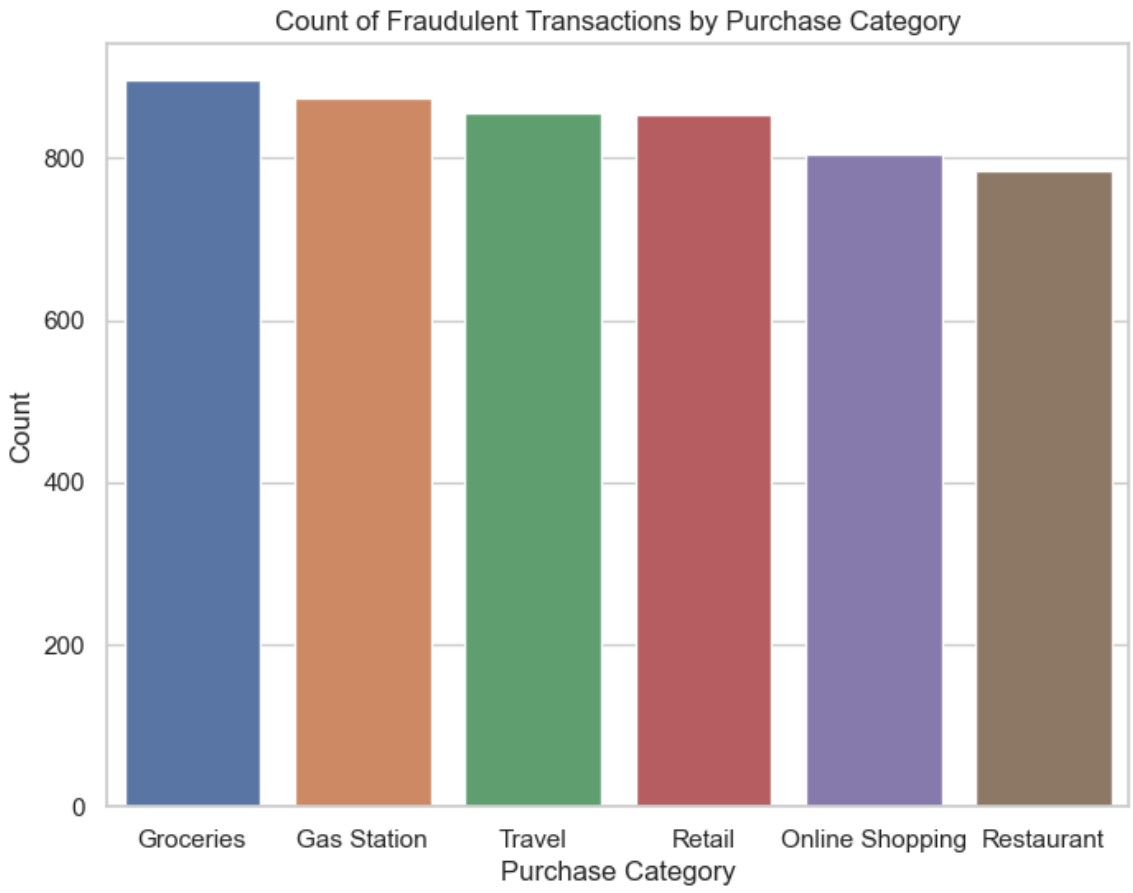
Insight:-

- as you can the see in the above chart.
- City - 7 has the most number of fraud transaction that is 120
- city - 48 and city - 41 has the second most number of fraud transaction more than 100

Let's see in which category fraudster are doing fraud transaction

In [39]:
```
plt.figure(figsize=(8, 6))
sns.set(style="whitegrid")
sns.countplot(data=df[df['is_fraudulent'] == 1], x='purchase_category',
plt.title('Count of Fraudulent Transactions by Purchase Category')
plt.xlabel('Purchase Category')
plt.ylabel('Count')
```

Out[39]:
```
Text(0, 0.5, 'Count')
```



Insight:-

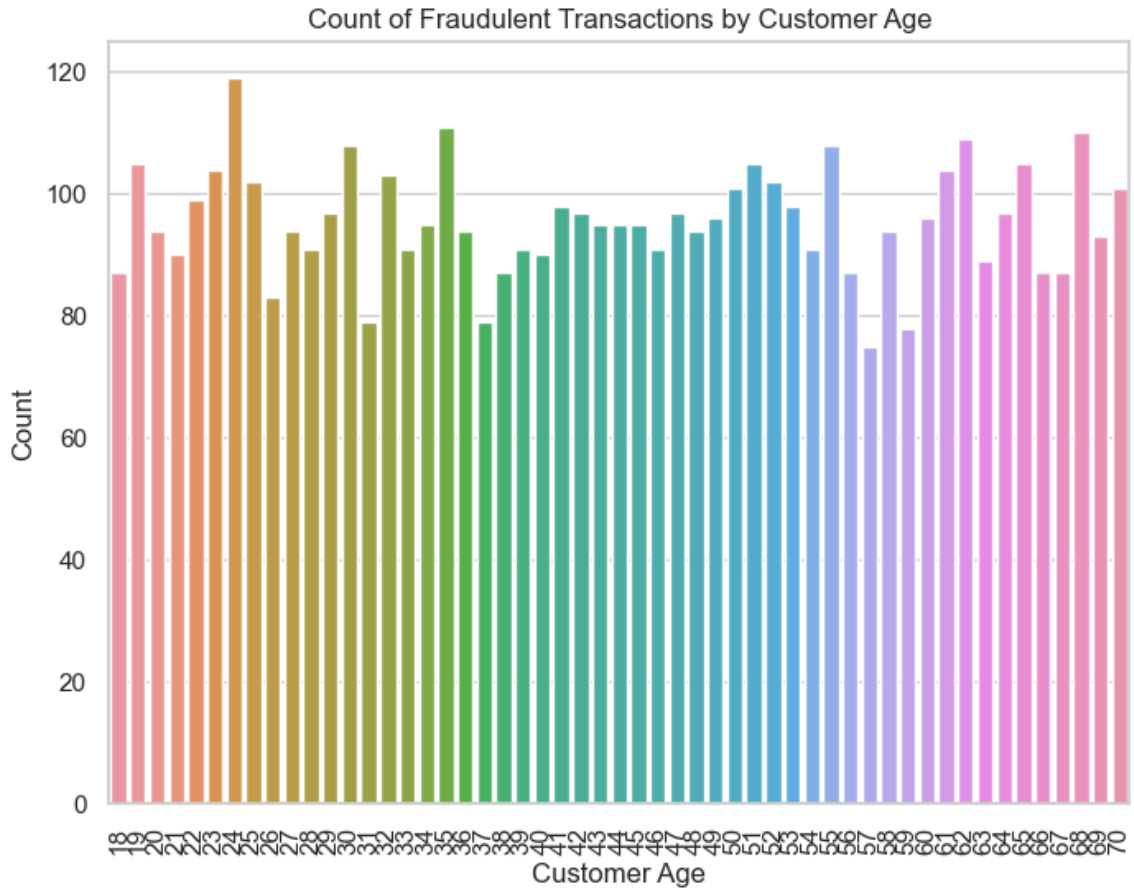
- The most number of fraud transaction is happening in the Grocies Category
- The second is gas station

Let's see the age, to better understand which age group doing the fraud transaction.

In [41]:

```
plt.figure(figsize=(8, 6))
sns.set(style="whitegrid")
sns.countplot(data=df[df['is_fraudulent'] == 1], x='customer_age')
plt.title('Count of Fraudulent Transactions by Customer Age')
plt.xlabel('Customer Age')
plt.ylabel('Count')
plt.xticks(rotation=90)
```

Out[41]:
```
(array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
        17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,
        34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50,
        51, 52]),
 [Text(0, 0, '18'),
  Text(1, 0, '19'),
  Text(2, 0, '20'),
  Text(3, 0, '21'),
  Text(4, 0, '22'),
  Text(5, 0, '23'),
  Text(6, 0, '24'),
  Text(7, 0, '25'),
  Text(8, 0, '26'),
  Text(9, 0, '27'),
  Text(10, 0, '28'),
  Text(11, 0, '29'),
  Text(12, 0, '30'),
  Text(13, 0, '31'),
  Text(14, 0, '32'),
  Text(15, 0, '33'),
  Text(16, 0, '34'),
  Text(17, 0, '35'),
  Text(18, 0, '36'),
  Text(19, 0, '37'),
  Text(20, 0, '38'),
  Text(21, 0, '39'),
  Text(22, 0, '40'),
  Text(23, 0, '41'),
  Text(24, 0, '42'),
  Text(25, 0, '43'),
  Text(26, 0, '44'),
  Text(27, 0, '45'),
  Text(28, 0, '46'),
  Text(29, 0, '47'),
  Text(30, 0, '48'),
  Text(31, 0, '49'),
  Text(32, 0, '50'),
  Text(33, 0, '51'),
  Text(34, 0, '52'),
  Text(35, 0, '53'),
  Text(36, 0, '54'),
  Text(37, 0, '55'),
  Text(38, 0, '56'),
  Text(39, 0, '57'),
  Text(40, 0, '58'),
  Text(41, 0, '59'),
  Text(42, 0, '60'),
  Text(43, 0, '61'),
  Text(44, 0, '62'),
  Text(45, 0, '63'),
  Text(46, 0, '64'),
  Text(47, 0, '65'),
  Text(48, 0, '66'),
  Text(49, 0, '67'),
  Text(50, 0, '68'),
  Text(51, 0, '69'),
  Text(52, 0, '70')])
```



Insight:-

- The msot number of fraudulent is in the age of 24
- Lower number is on the age of 56 and 58

Conclusion

The "Fraud Analysis" dataset serves as a valuable tool for those interested in the critical field of financial fraud prevention and analysis. With its synthetic yet comprehensive data, it offers an opportunity to hone skills and develop models for fraud detection, all within a safe and controlled environment.

As we conclude our exploration, it's important to recognize the significance of staying one step ahead in the battle against financial fraud. The insights gained from this dataset can contribute to improved security measures, safeguarding both individuals and organizations from the devastating impacts of fraudulent activities.

In [ ]:
```
1
```