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

df=pd.read_csv('churn.csv')
df.head()
```

```
   customerID  gender  SeniorCitizen Partner Dependents  tenure
PhoneService  \
0  7590-VHVEG  Female              0     Yes         No       1
No
1  5575-GNVDE    Male              0      No         No      34
Yes
2  3668-QPYBK    Male              0      No         No       2
Yes
3  7795-CFOCW    Male              0      No         No      45
No
4  9237-HQITU  Female              0      No         No       2
Yes

      MultipleLines InternetService OnlineSecurity  ...
DeviceProtection  \
0  No phone service             DSL             No  ...
No
1                No             DSL            Yes  ...
Yes
2                No             DSL            Yes  ...
No
3  No phone service             DSL            Yes  ...
Yes
4                No     Fiber optic             No  ...
No

  TechSupport StreamingTV StreamingMovies        Contract
PaperlessBilling  \
0          No          No              No  Month-to-month
Yes
1          No          No              No        One year
No
2          No          No              No  Month-to-month
Yes
3         Yes          No              No        One year
No
4          No          No              No  Month-to-month
Yes

               PaymentMethod MonthlyCharges  TotalCharges Churn
0           Electronic check          29.85         29.85    No
1               Mailed check          56.95        1889.5    No
```

```
2               Mailed check          53.85        108.15   Yes
3  Bank transfer (automatic)          42.30       1840.75    No
4           Electronic check          70.70        151.65   Yes

[5 rows x 21 columns]

df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   customerID        7043 non-null   object 
 1   gender            7043 non-null   object 
 2   SeniorCitizen     7043 non-null   int64  
 3   Partner           7043 non-null   object 
 4   Dependents        7043 non-null   object 
 5   tenure            7043 non-null   int64  
 6   PhoneService      7043 non-null   object 
 7   MultipleLines     7043 non-null   object 
 8   InternetService   7043 non-null   object 
 9   OnlineSecurity    7043 non-null   object 
 10  OnlineBackup      7043 non-null   object 
 11  DeviceProtection  7043 non-null   object 
 12  TechSupport       7043 non-null   object 
 13  StreamingTV       7043 non-null   object 
 14  StreamingMovies   7043 non-null   object 
 15  Contract          7043 non-null   object 
 16  PaperlessBilling  7043 non-null   object 
 17  PaymentMethod     7043 non-null   object 
 18  MonthlyCharges    7043 non-null   float64
 19  TotalCharges      7043 non-null   object 
 20  Churn             7043 non-null   object 
dtypes: float64(1), int64(2), object(18)
memory usage: 1.1+ MB
```

#replacing blanks with 0 as tenure is 0 and no total charges recorded

```
df["TotalCharges"]=df["TotalCharges"].replace(" ","0")
df["TotalCharges"]=df["TotalCharges"].astype("float")

df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7043 entries, 0 to 7042
Data columns (total 21 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   customerID        7043 non-null   object 
 1   gender            7043 non-null   object
 2   SeniorCitizen     7043 non-null   int64
 3   Partner           7043 non-null   object
 4   Dependents        7043 non-null   object
 5   tenure            7043 non-null   int64
 6   PhoneService      7043 non-null   object
 7   MultipleLines     7043 non-null   object
 8   InternetService   7043 non-null   object
 9   OnlineSecurity    7043 non-null   object
 10  OnlineBackup      7043 non-null   object
 11  DeviceProtection  7043 non-null   object
 12  TechSupport       7043 non-null   object
 13  StreamingTV       7043 non-null   object
 14  StreamingMovies   7043 non-null   object
 15  Contract          7043 non-null   object
 16  PaperlessBilling  7043 non-null   object
 17  PaymentMethod     7043 non-null   object
 18  MonthlyCharges    7043 non-null   float64
 19  TotalCharges      7043 non-null   float64
 20  Churn             7043 non-null   object
dtypes: float64(2), int64(2), object(17)
memory usage: 1.1+ MB
```

```
df.isnull().sum().sum()
```

```
np.int64(0)
```

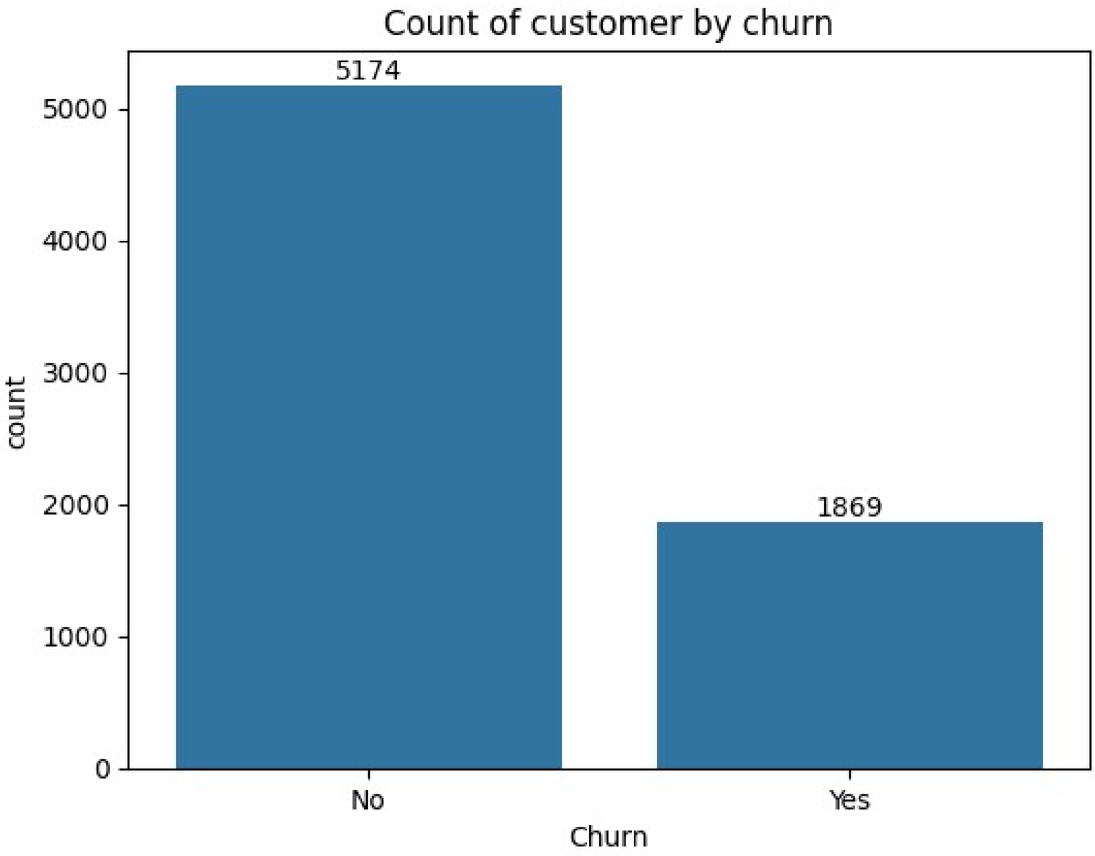
```
df.describe()
```

```
       SeniorCitizen       tenure  MonthlyCharges  TotalCharges
count    7043.000000  7043.000000     7043.000000   7043.000000
mean        0.162147    32.371149       64.761692   2279.734304
std         0.368612    24.559481       30.090047   2266.794470
min         0.000000     0.000000       18.250000      0.000000
25%         0.000000     9.000000       35.500000    398.550000
50%         0.000000    29.000000       70.350000   1394.550000
75%         0.000000    55.000000       89.850000   3786.600000
max         1.000000    72.000000      118.750000   8684.800000
```

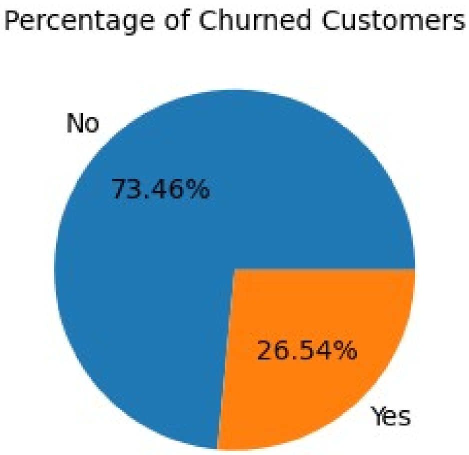
```
df["customerID"].duplicated().sum()
```

```
np.int64(0)
```

```
ax=sns.countplot(x='Churn',data=df)
ax.bar_label(ax.containers[0])
plt.title("Count of customer by churn")
plt.show()
```



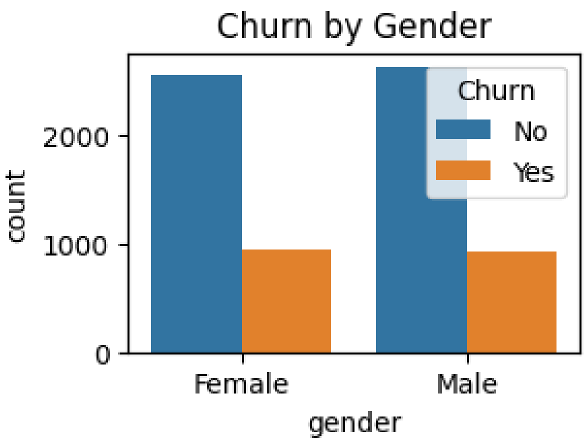
```python
plt.figure(figsize=(3,4))
gb=df.groupby("Churn").agg({'Churn':"count"})
plt.pie(gb['Churn'],  labels= gb.index,autopct = "%1.2F%%")
plt.title("Percentage of Churned Customers",fontsize=10)
plt.show()
gb
```



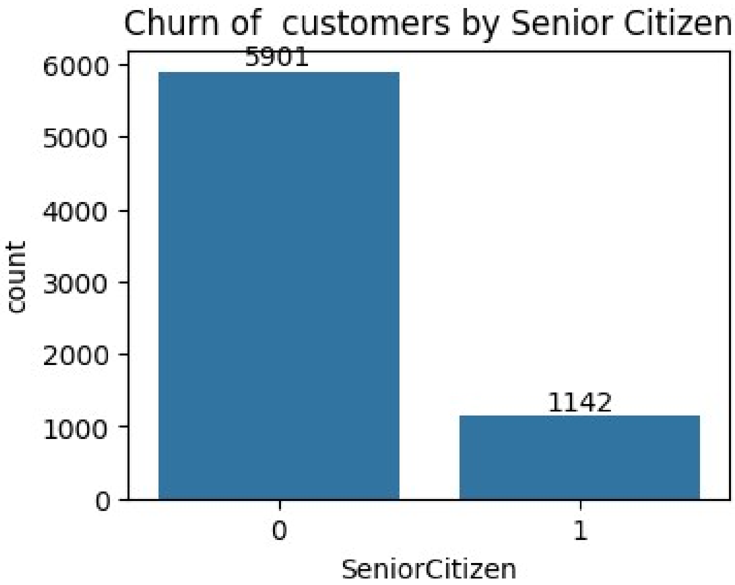
```
       Churn
Churn
No      5174
Yes     1869
```

#from the givenpie chart we can conclude that @^.54% of our customers have churnder out,#not lets's explor the reason behind it

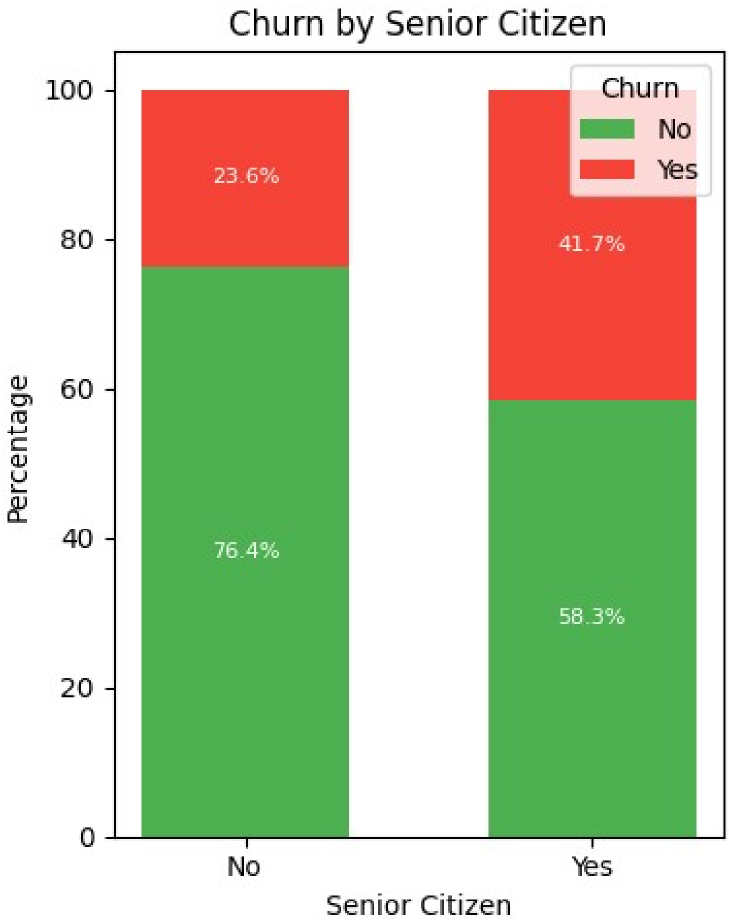
```
plt.figure(figsize=(3,2))
sns.countplot(x= "gender",data= df ,hue="Churn",)
plt.title("Churn by Gender")
plt.show()
```



```
plt.figure(figsize=(4,3))
ax=sns.countplot(x= "SeniorCitizen",data= df)
ax.bar_label(ax.containers[0])
plt.title("Churn of  customers by Senior Citizen")
plt.show()
```

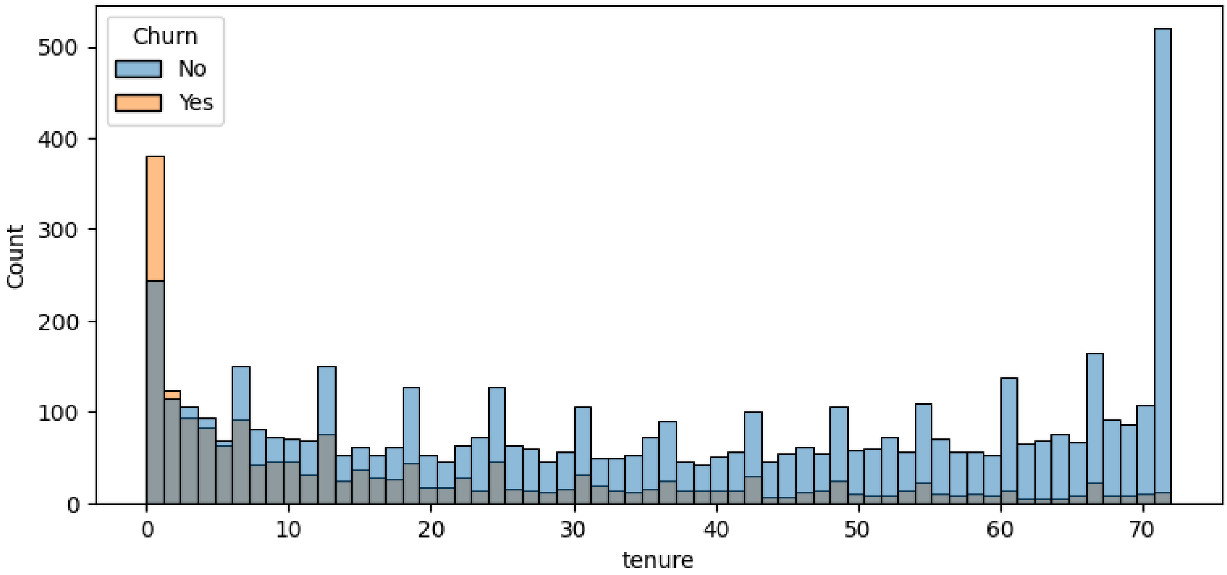


```
import pandas as pd
import matplotlib.pyplot as plt

# 1. Count data by group
grouped = df.groupby(['SeniorCitizen',
'Churn']).size().unstack(fill_value=0)

# 2. Convert counts to percentage
percentages = grouped.div(grouped.sum(axis=1), axis=0) * 100

# 3. Plot
fig, ax = plt.subplots(figsize=(4,5))

colors = ['#4CAF50', '#F44336']  # Green for No, Red for Yes
bottoms = [0, 0]  # for stacking

for idx, churn_val in enumerate(percentages.columns):  # "No" and
"Yes"
    heights = percentages[churn_val].values
    bars = ax.bar(
        percentages.index,
        heights,
        bottom=bottoms,
        color=colors[idx],
        label=churn_val,
        width=0.6
    )

    # Add % labels
    for i, bar in enumerate(bars):
        ax.text(
            bar.get_x() + bar.get_width()/2,
            bottoms[i] + heights[i]/2,
            f'{heights[i]:.1f}%',
            ha='center', va='center', fontsize=8, color='white'
        )

    bottoms = [bottoms[i] + heights[i] for i in range(len(heights))]

# 4. Formatting
ax.set_xticks([0, 1])
ax.set_xticklabels(['No', 'Yes'])  # SeniorCitizen: 0 = No, 1 = Yes
ax.set_xlabel("Senior Citizen")
ax.set_ylabel("Percentage")
ax.set_title("Churn by Senior Citizen")
ax.legend(title='Churn')
plt.tight_layout()
plt.show()
```



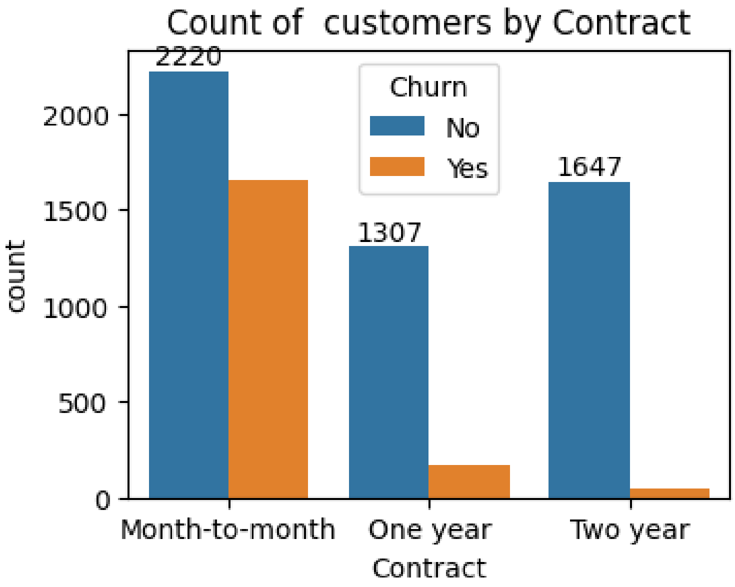
#comparative a greater percentage of people in senior citizen category have churned

```
plt.figure(figsize=(9,4))
sns.histplot(x="tenure",data=df,bins=60,hue="Churn")
plt.show()
```



#people who have used our services for a long time have stays and people who used our services #1 or #2 months have churned

```
plt.figure(figsize=(4,3))
ax=sns.countplot(x= "Contract",data= df,hue="Churn",)
ax.bar_label(ax.containers[0])
plt.title("Count of  customers by Contract")
plt.show()
```

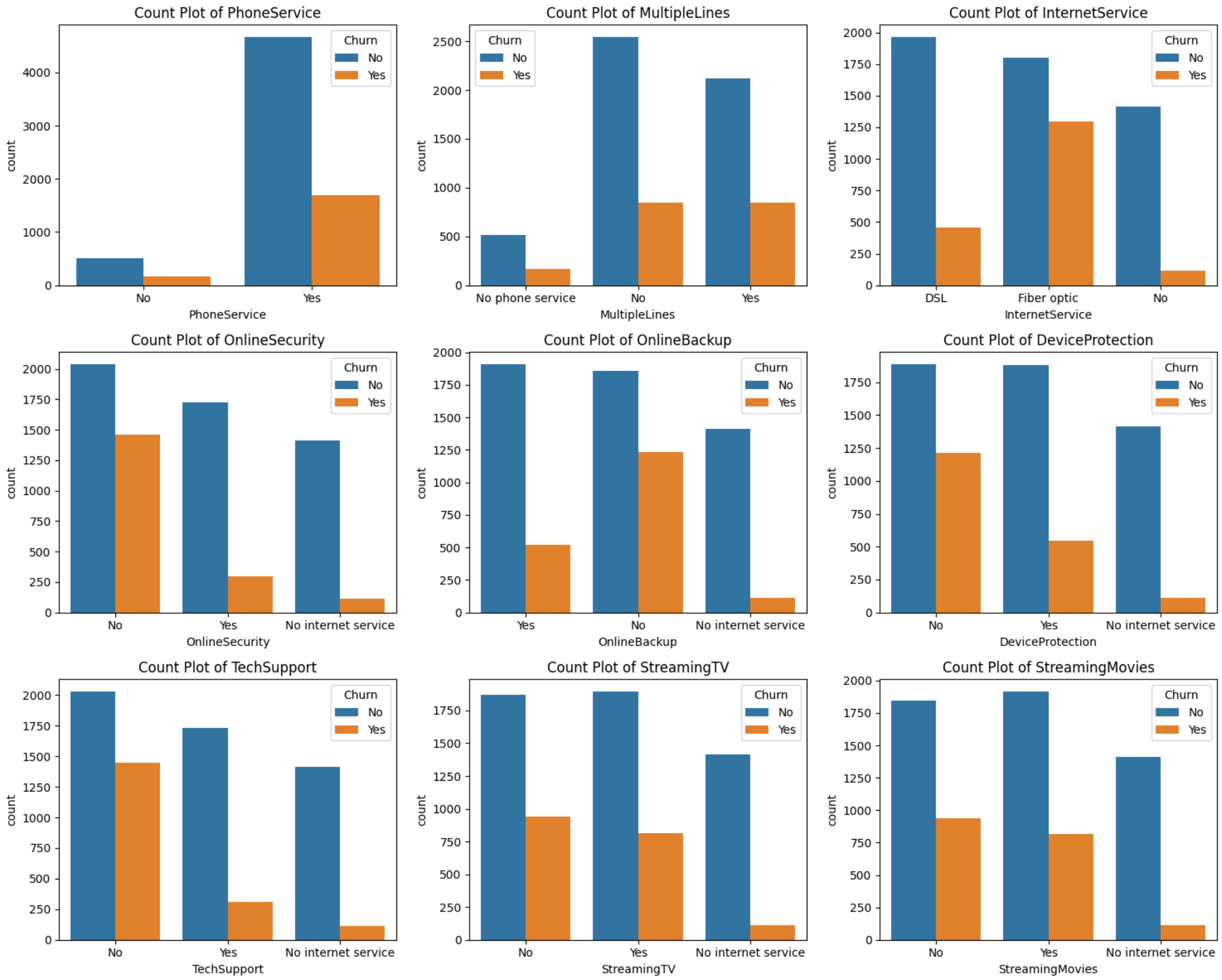


#people who have month to month contract are likely to churn then from those whohave 1 oe 2 years of contract

```
df.columns.values
```

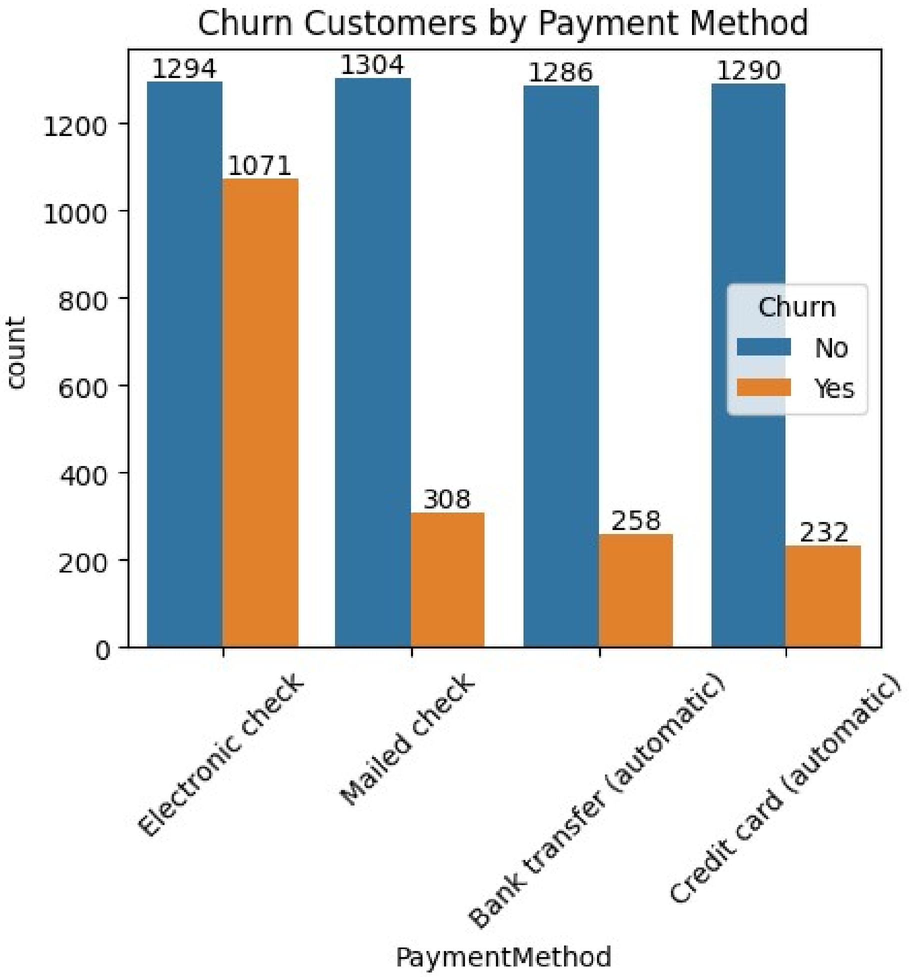
```
array(['customerID', 'gender', 'SeniorCitizen', 'Partner',
'Dependents',
       'tenure', 'PhoneService', 'MultipleLines', 'InternetService',
       'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
       'TechSupport', 'StreamingTV', 'StreamingMovies', 'Contract',
       'PaperlessBilling', 'PaymentMethod', 'MonthlyCharges',
       'TotalCharges', 'Churn'], dtype=object)
```

```
# Example DataFrame (replace this with your actual data)
# df = pd.read_csv("your_file.csv")

# Columns to plot
columns = [
    'PhoneService', 'MultipleLines', 'InternetService',
    'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
    'TechSupport', 'StreamingTV', 'StreamingMovies'
]

# Set up the matplotlib figure
n_cols = 3
n_rows = (len(columns) + n_cols - 1) // n_cols  # ceil division for
rows

fig, axes = plt.subplots(n_rows, n_cols, figsize=(15, n_rows * 4))
axes = axes.flatten()  # Flatten in case of 2D axes array

# Create a countplot for each column
for i, col in enumerate(columns):
    sns.countplot(data=df, x=col, ax=axes[i],hue =df["Churn"])
    axes[i].set_title(f"Count Plot of {col}")
    axes[i].tick_params(axis='x', rotation=0)  # Rotate x labels if
needed

# Turn off any unused subplots
for j in range(i + 1, len(axes)):
    fig.delaxes(axes[j])

plt.tight_layout()
plt.show()
```



#Most customers have PhoneService and MultipleLines, but churn is higher among those with InternetService (DSL/Fiber). Customers without OnlineSecurity, OnlineBackup, DeviceProtection, or TechSupport tend to churn more. Streaming services (TV/Movies) don’t show a strong difference in churn, but lack of security/backup services correlates with higher churn.

```
plt.figure(figsize=(5,4))
ax=sns.countplot(x= "PaymentMethod",data= df,hue="Churn",)
ax.bar_label(ax.containers[0])
ax.bar_label(ax.containers[1])
plt.xticks(rotation=45)
plt.title("Churn Customers by Payment Method")
plt.show()
```



#customers is likely to churn when he is using electronic payment method