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

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

```
      customerID  gender  SeniorCitizen Partner Dependents  tenure  \
0     7590-VHVEG  Female              0     Yes         No       1
1     5575-GNVDE    Male              0      No         No      34
2     3668-QPYBK    Male              0      No         No       2
3     7795-CFOCW    Male              0      No         No      45
4     9237-HQITU  Female              0      No         No       2
...          ...     ...            ...     ...        ...     ...
7038  6840-RESVB    Male              0     Yes        Yes      24
7039  2234-XADUH  Female              0     Yes        Yes      72
7040  4801-JZAZL  Female              0     Yes        Yes      11
7041  8361-LTMKD    Male              1     Yes         No       4
7042  3186-AJIEK    Male              0      No         No      66

     PhoneService     MultipleLines InternetService
OnlineSecurity  ...  \
0              No  No phone service             DSL
No  ...
1             Yes                No             DSL
Yes  ...
2             Yes                No             DSL
Yes  ...
3              No  No phone service             DSL
Yes  ...
4             Yes                No     Fiber optic
No  ...
...           ...               ...             ...            ...  ..
.
7038          Yes               Yes             DSL
Yes  ...
7039          Yes               Yes     Fiber optic
No  ...
7040           No  No phone service             DSL
Yes  ...
7041          Yes               Yes     Fiber optic
No  ...
7042          Yes                No     Fiber optic
Yes  ...

     DeviceProtection TechSupport StreamingTV StreamingMovies
Contract  \
0                  No          No          No              No  Month-
to-month
```

```
1                  Yes           No           No               No
One year
2                   No           No           No               No  Month-
to-month
3                  Yes          Yes           No               No
One year
4                   No           No           No               No  Month-
to-month
...                ...          ...          ...              ...
...
7038               Yes          Yes          Yes              Yes
One year
7039               Yes           No          Yes              Yes
One year
7040                No           No           No               No  Month-
to-month
7041                No           No           No               No  Month-
to-month
7042               Yes          Yes          Yes              Yes
Two year

     PaperlessBilling              PaymentMethod MonthlyCharges
TotalCharges  \
0                  Yes           Electronic check          29.85
29.85
1                   No               Mailed check          56.95
1889.5
2                  Yes               Mailed check          53.85
108.15
3                   No  Bank transfer (automatic)          42.30
1840.75
4                  Yes           Electronic check          70.70
151.65
...                ...                        ...            ...
...
7038               Yes               Mailed check          84.80
1990.5
7039               Yes    Credit card (automatic)         103.20
7362.9
7040               Yes           Electronic check          29.60
346.45
7041               Yes               Mailed check          74.40
306.6
7042               Yes  Bank transfer (automatic)         105.65
6844.5

     Churn
0       No
1       No
```

```
2      Yes
3       No
4      Yes
...    ...
7038    No
7039    No
7040    No
7041   Yes
7042    No

[7043 rows x 21 columns]
```

```
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)
```

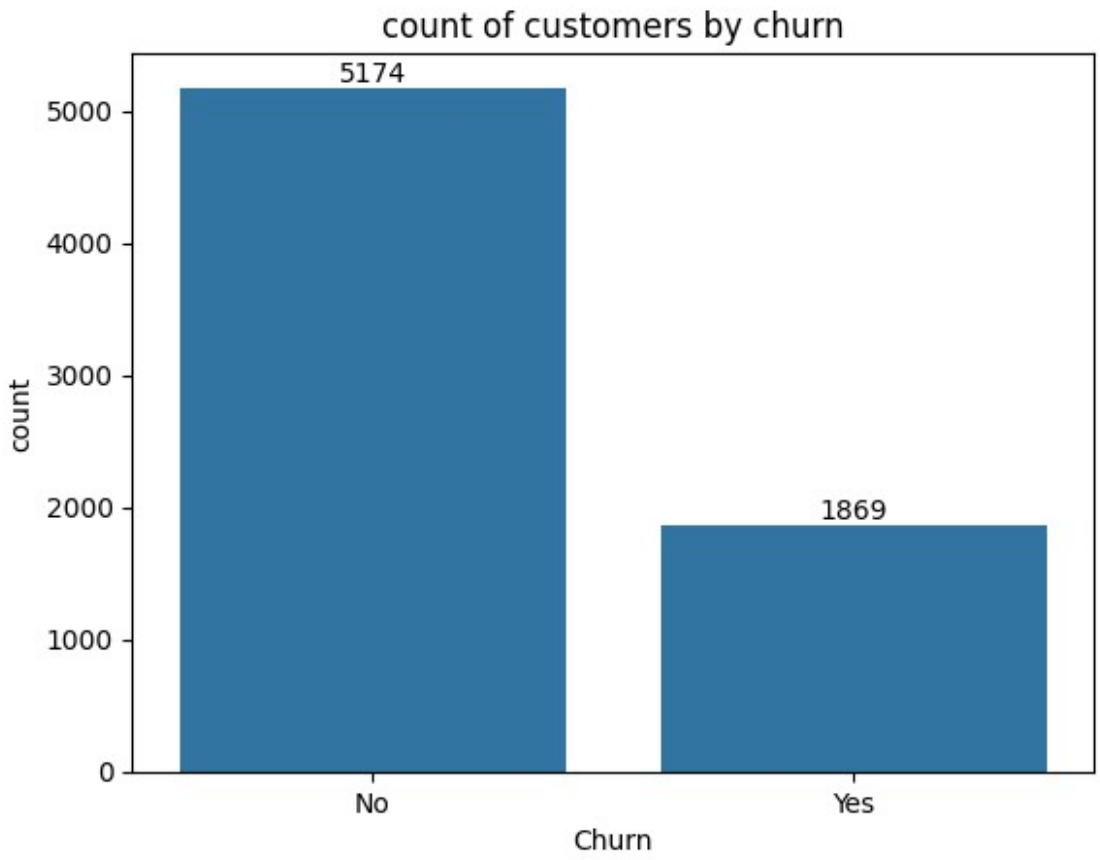
```
df.describe()
```

```
       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
```

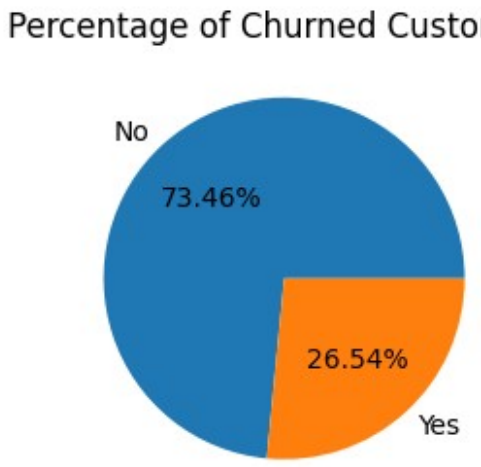
```
df["customerID"].duplicated().sum()
```

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

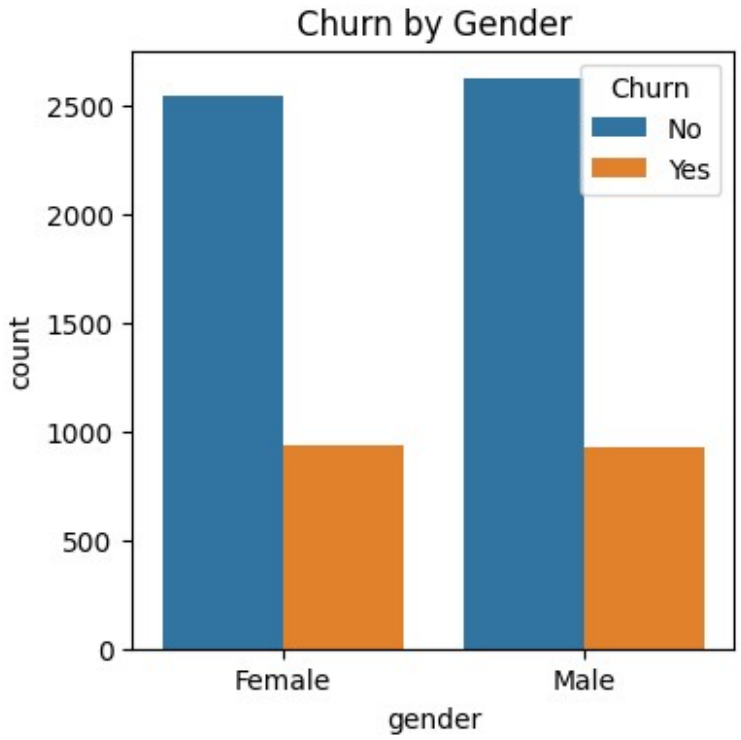
```
def conv(value):
    if value ==1:
        return "yes"
    else:
        return "no"

df['SeniorCitizen'] = df["SeniorCitizen"].apply(conv)

#converted 0 and 1 values of senior citizen to yes/no to make is
easier to understand

ax = sns.countplot(x = 'Churn' , data= df)

ax.bar_label(ax.containers[0])
plt.title("count of customers by churn")
plt.show()
```



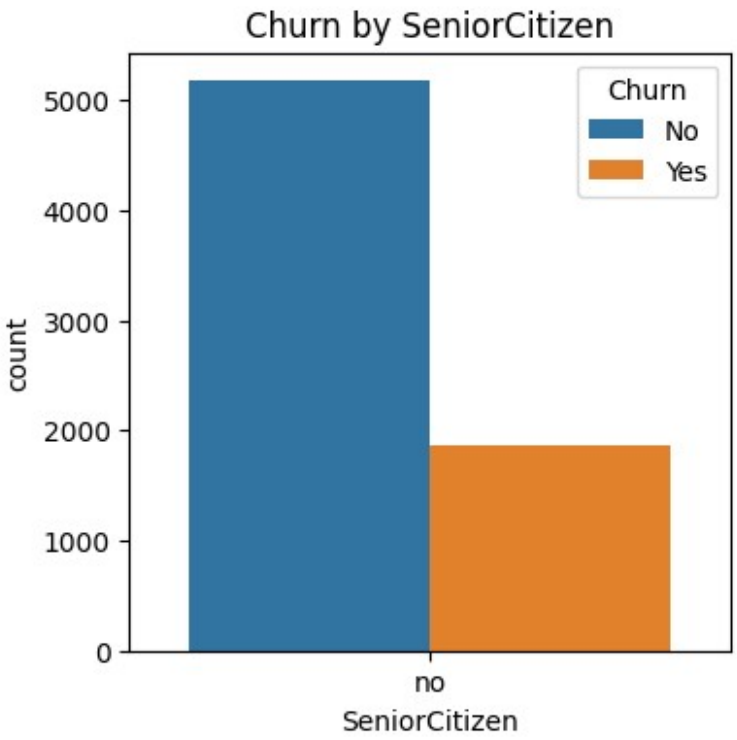

```
plt.figure(figsize = (3,4))
gb = df.groupby("Churn").agg({'Churn':"count"})
plt.pie(gb['Churn'], labels = gb.index, autopct = "%1.2f%%")
plt.title("Percentage of Churned Customers")
plt.show()
```



```
#from the given pie chart we can conclude that 26.54% of our customers
have churned out.
# not let's  explore the reasonbehind it


plt.figure(figsize=(4,4))
sns.countplot(x = "gender", data = df, hue = "Churn")
plt.title("Churn by Gender")
plt.show()
```

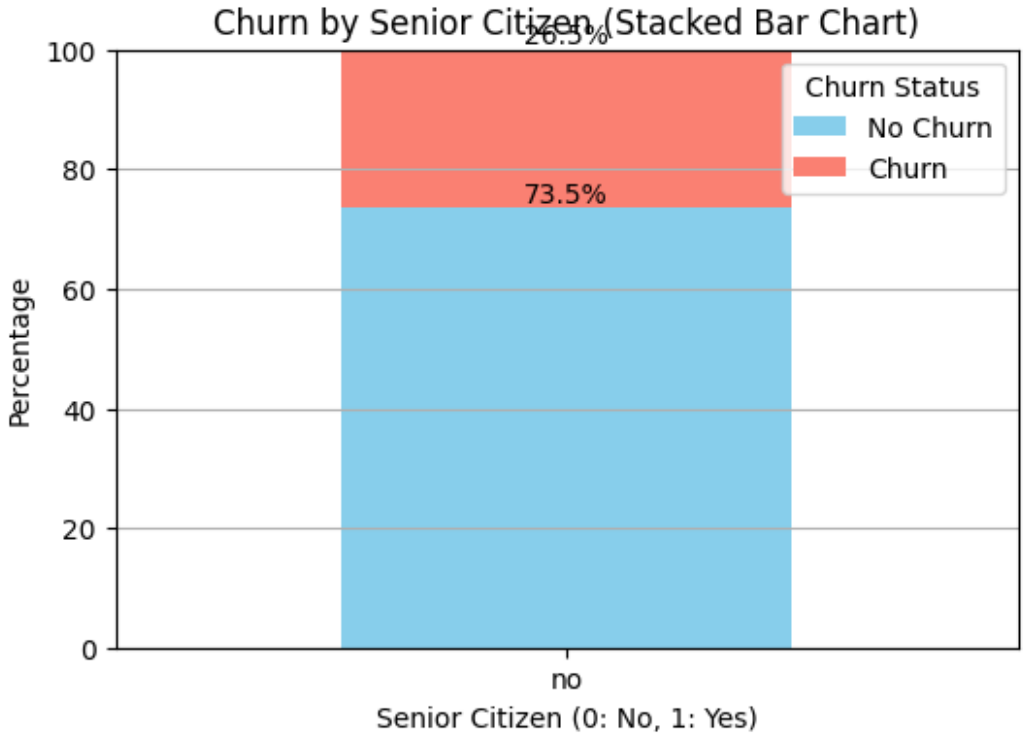



```
plt.figure(figsize=(4,4))
sns.countplot(x = "SeniorCitizen", data = df, hue = "Churn")
plt.title("Churn by SeniorCitizen")
plt.show()
```




```
counts = df.groupby(['SeniorCitizen',
'Churn']).size().unstack(fill_value=0)

# Calculate percentages
percentages = counts.div(counts.sum(axis=1), axis=0) * 100

# Create stacked bar chart
ax = percentages.plot(kind='bar', stacked=True, figsize=(6, 4),
color=['skyblue', 'salmon'])

# Add percentage labels
for p in ax.patches:
    height = p.get_height()
    if height > 0:
        ax.annotate(f'{height:.1f}%',
                    (p.get_x() + p.get_width() / 2, p.get_y() +
height),
                    ha='center', va='bottom')

# Customize the chart
plt.title("Churn by Senior Citizen (Stacked Bar Chart)")
plt.xlabel("Senior Citizen (0: No, 1: Yes)")
plt.ylabel("Percentage")
plt.xticks(rotation=0)
plt.legend(title="Churn Status", labels=['No Churn', 'Churn'])
plt.ylim(0, 100)  # Set y-axis limit to 100%
plt.grid(axis='y')
```

```
plt.show()
```



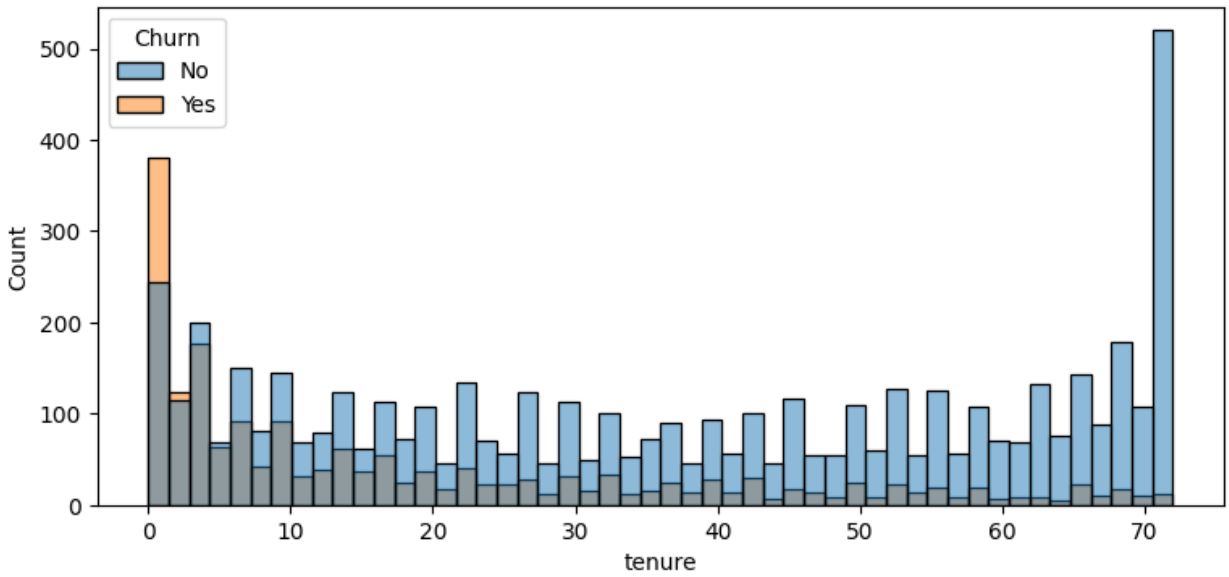

```
#caomparative a grater percentage of people in senior citizen category
have churned.

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

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

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	\
0	7590-VHVEG	Female	0	Yes	No	1	
1	5575-GNVDE	Male	0	No	No	34	
2	3668-QPYBK	Male	0	No	No	2	
3	7795-CFOCW	Male	0	No	No	45	
4	9237-HQITU	Female	0	No	No	2	
...	...	...	...	...	...	...	
7038	6840-RESVB	Male	0	Yes	Yes	24	
7039	2234-XADUH	Female	0	Yes	Yes	72	
7040	4801-JZAZL	Female	0	Yes	Yes	11	
7041	8361-LTMKD	Male	1	Yes	No	4	
7042	3186-AJIEK	Male	0	No	No	66	

```
     PhoneService     MultipleLines InternetService
OnlineSecurity  ...  \
0              No  No phone service             DSL
No  ...
1             Yes                No             DSL
Yes  ...
2             Yes                No             DSL
Yes  ...
3              No  No phone service             DSL
Yes  ...
4             Yes                No     Fiber optic
No  ...
...           ...               ...             ...             ...  ..
.
7038          Yes               Yes             DSL
Yes  ...
7039          Yes               Yes     Fiber optic
No  ...
7040           No  No phone service             DSL
Yes  ...
7041          Yes               Yes     Fiber optic
No  ...
7042          Yes                No     Fiber optic
Yes  ...

     DeviceProtection TechSupport StreamingTV StreamingMovies
Contract  \
0                  No          No          No              No  Month-
to-month
1                 Yes          No          No              No
One year
2                  No          No          No              No  Month-
to-month
3                 Yes         Yes          No              No
One year
4                  No          No          No              No  Month-
to-month
...               ...         ...         ...             ...
...
7038              Yes         Yes         Yes             Yes
One year
7039              Yes          No         Yes             Yes
One year
7040               No          No          No              No  Month-
to-month
7041               No          No          No              No  Month-
to-month
7042              Yes         Yes         Yes             Yes
```

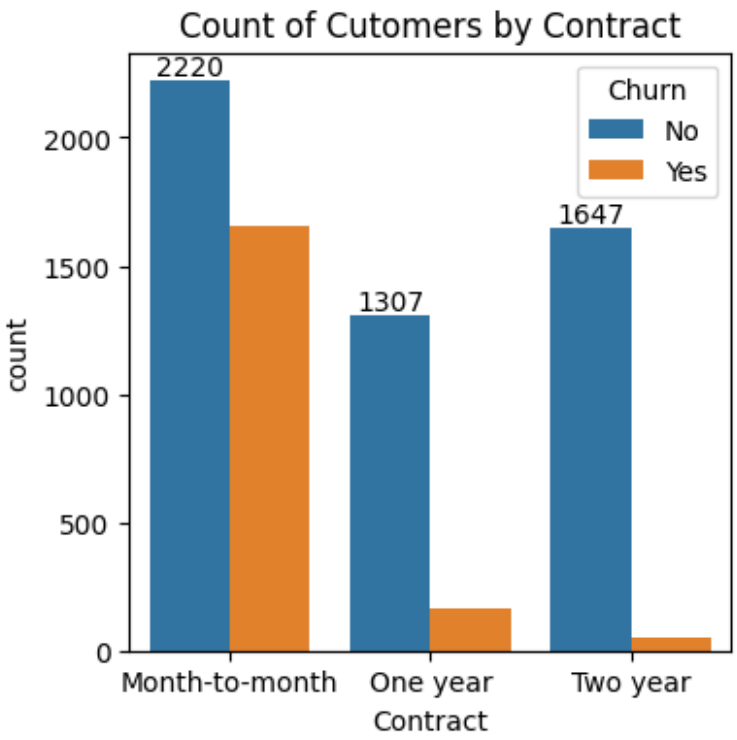
```
Two year

     PaperlessBilling              PaymentMethod MonthlyCharges
TotalCharges  \
0                 Yes           Electronic check          29.85
29.85
1                  No               Mailed check          56.95
1889.5
2                 Yes               Mailed check          53.85
108.15
3                  No  Bank transfer (automatic)          42.30
1840.75
4                 Yes           Electronic check          70.70
151.65
...               ...                        ...            ...
...
7038              Yes               Mailed check          84.80
1990.5
7039              Yes    Credit card (automatic)         103.20
7362.9
7040              Yes           Electronic check          29.60
346.45
7041              Yes               Mailed check          74.40
306.6
7042              Yes  Bank transfer (automatic)         105.65
6844.5

     Churn
0       No
1       No
2      Yes
3       No
4      Yes
...    ...
7038    No
7039    No
7040    No
7041   Yes
7042    No

[7043 rows x 21 columns]
```

```
plt.figure(figsize=(9,4))
sns.histplot(x = "tenure", data=df, hue=("Churn"), bins = 50)
plt.title
plt.show()
```



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

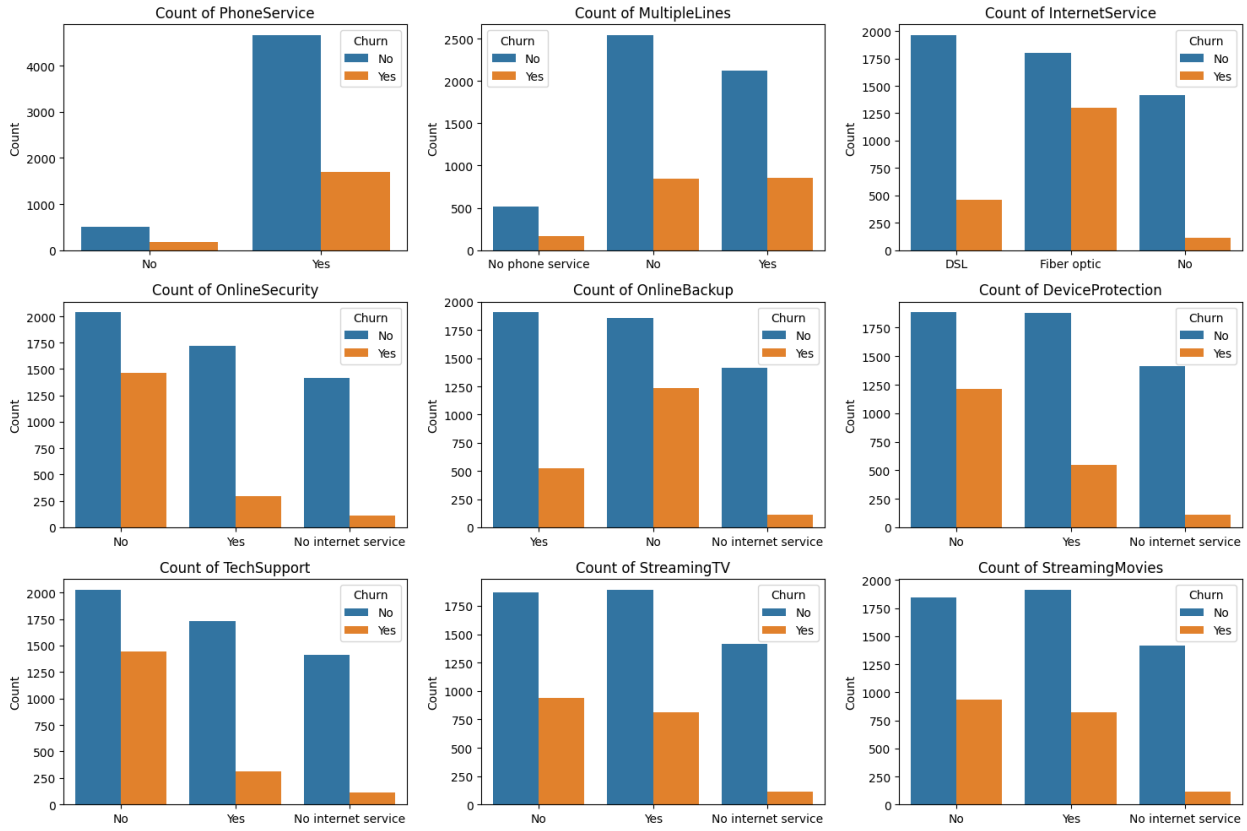
```
plt.figure(figsize=(4,4))
ax = sns.countplot(x = "Contract", data = df, hue="Churn")
ax.bar_label(ax.containers[0])
plt.title("Count of Cutomers by Contract")
plt.show()
```



people who have month to month contract are likely to churn then from those who have 1 or 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)

columns = ['PhoneService', 'MultipleLines', 'InternetService',
           'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
           'TechSupport', 'StreamingTV', 'StreamingMovies']

# Set up the subplots
n_cols = 3  # Number of columns in the subplot grid
n_rows = (len(columns) + n_cols - 1) // n_cols  # Calculate number of
rows

fig, axes = plt.subplots(n_rows, n_cols, figsize=(15, 10))
```

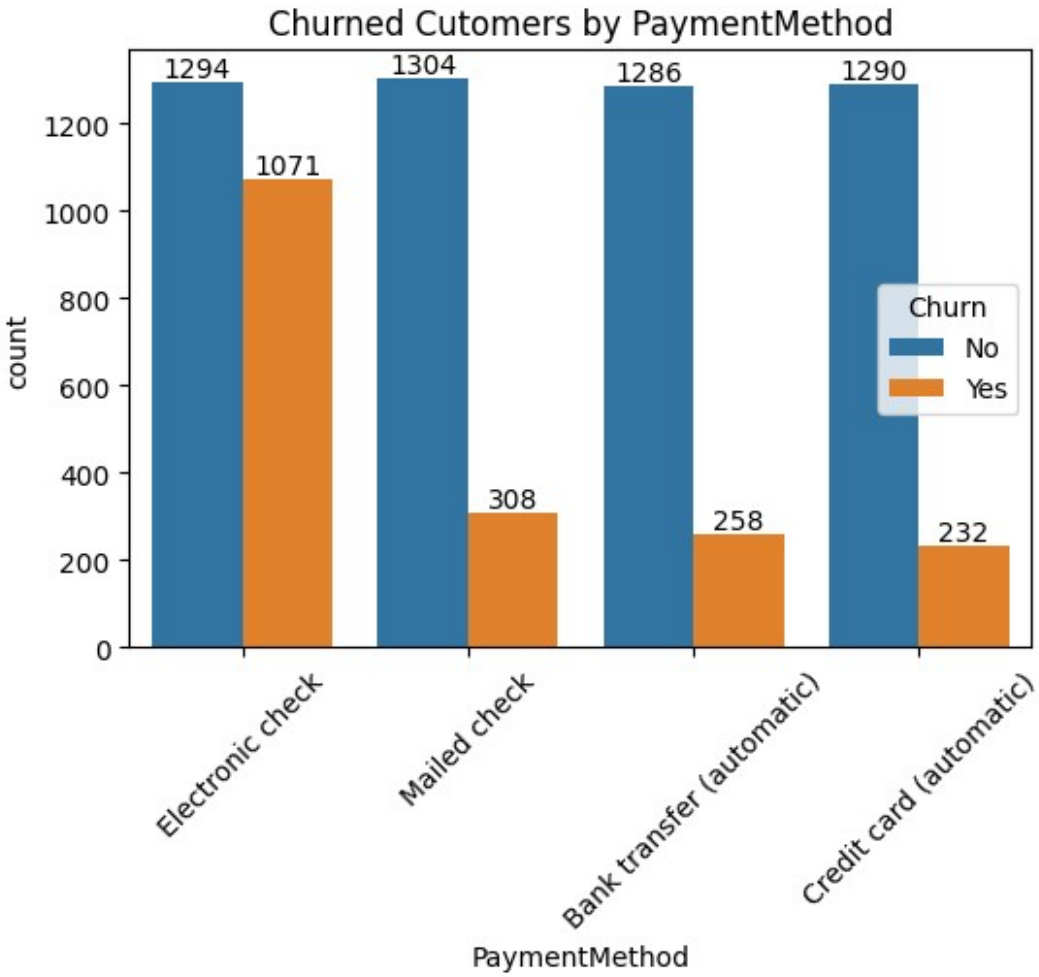
```
axes = axes.flatten()  # Flatten the 2D array of axes for easy
indexing

# Create count plots for each column
for i, col in enumerate(columns):
    sns.countplot(x=col, data=df, ax=axes[i], hue= df["Churn"])
    axes[i].set_title(f'Count of {col}')
    axes[i].set_xlabel('')
    axes[i].set_ylabel('Count')

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

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



The majority of customers who do not churn tend to have services like PhoneService, InternetService (particularly DSL), and OnlineSecurity enabled. For services like OnlineBackup, TechSupport, and StreamingTV, churn rates are noticeably higher when these services are not used or are unavailable.

```
plt.figure(figsize=(6,4))
ax = sns.countplot(x = "PaymentMethod", data = df, hue="Churn")
```

```
ax.bar_label(ax.containers[0])
ax.bar_label(ax.containers[1])
plt.title("Churned Cutomers by PaymentMethod")
plt.xticks(rotation= 45)
plt.show()
```



customer is likely to churn when he is using electronic check as a payment method.