Untitled

September 3, 2025

[2]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

[3]:
```
df = pd.read_csv('Customer Churn.csv')
```

[20]:
```
df.head(10)
```

[20]:
```
   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
5  9305-CDSKC  Female              0      No         No       8          Yes
6  1452-KIOVK    Male              0      No        Yes      22          Yes
7  6713-OKOMC  Female              0      No         No      10           No
8  7892-POOKP  Female              0     Yes         No      28          Yes
9  6388-TABGU    Male              0      No        Yes      62          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
5               Yes     Fiber optic             No  …              Yes
6               Yes     Fiber optic             No  …               No
7  No phone service             DSL            Yes  …               No
8               Yes     Fiber optic             No  …              Yes
9                No             DSL            Yes  …               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
5          No         Yes             Yes  Month-to-month             Yes
6          No         Yes              No  Month-to-month             Yes
7          No          No              No  Month-to-month              No
8         Yes         Yes             Yes  Month-to-month             Yes
9          No          No              No        One year              No

               PaymentMethod MonthlyCharges  TotalCharges  Churn
0           Electronic check          29.85         29.85     No
1               Mailed check          56.95       1889.50     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           Electronic check          99.65        820.50    Yes
6    Credit card (automatic)          89.10       1949.40     No
7               Mailed check          29.75        301.90     No
8           Electronic check         104.80       3046.05    Yes
9  Bank transfer (automatic)          56.15       3487.95     No

[10 rows x 21 columns]
```

```
[5]: 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
```

```
[8]: df['TotalCharges'] = df['TotalCharges'].replace(" ",0)
```

```
[9]: df['TotalCharges'] = df['TotalCharges'].astype("float")
```

```
[19]: df.head(5)
```

```
[19]:    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       1889.50     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]
```

```
[15]: df.isnull().sum().sum()
```

```
[15]: 0
```

```
[16]: df.describe()
```

[16]:

```
       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
```

[18]:

```
df.duplicated().sum()
```

[18]:

```
0
```

[22]:

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

[22]:

```
0
```

[23]:

```
def conv(value):
    if value == 1:
        return "YES"
    else:
        return "NO"

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

[27]:

```
df['SeniorCitizen'].head(30)
```

[27]:

```
0       NO
1       NO
2       NO
3       NO
4       NO
5       NO
6       NO
7       NO
8       NO
9       NO
10      NO
11      NO
12      NO
13      NO
14      NO
15      NO
16      NO
17      NO
18      NO
```

```
19      NO
20     YES
21      NO
22      NO
23      NO
24      NO
25      NO
26      NO
27      NO
28      NO
29      NO
Name: SeniorCitizen, dtype: object
```

[29]:

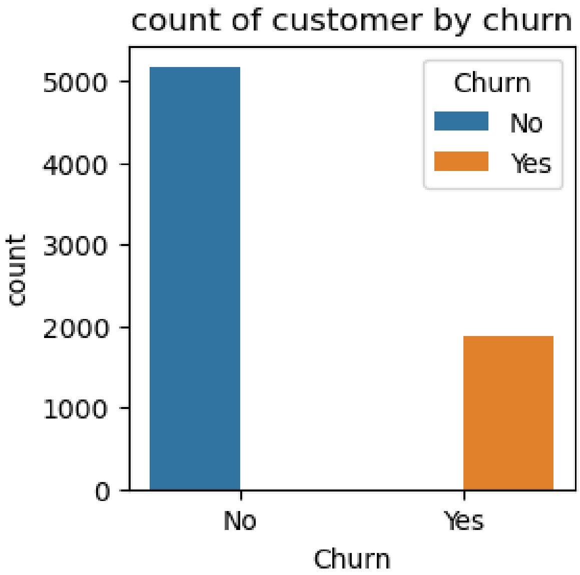
```python
df.columns
```

[29]:

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

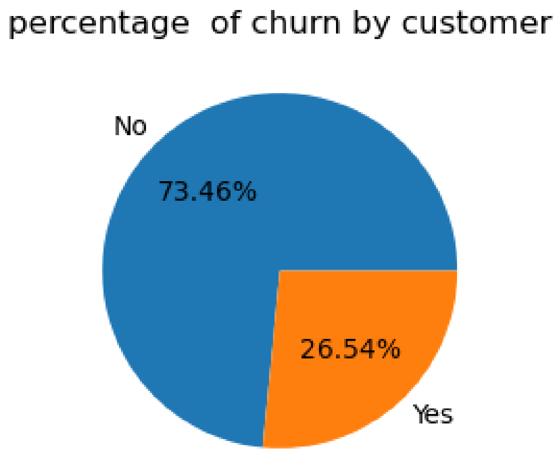
[52]:

```python
plt.figure(figsize=(3,3))
sns.countplot(x='Churn',data = df,hue='Churn')
plt.title('count of customer by churn')
plt.show()
```



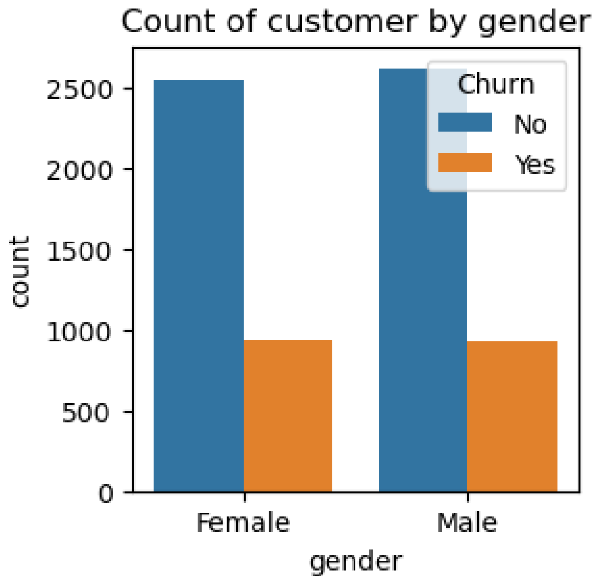
[51]:

```
plt.figure(figsize=(3,3))
gb = df.groupby('Churn').agg({"Churn":"count"})
plt.pie(gb['Churn'],labels = gb.index,autopct = '%1.2f%%')
plt.title("percentage  of churn by customer")

plt.show()
```



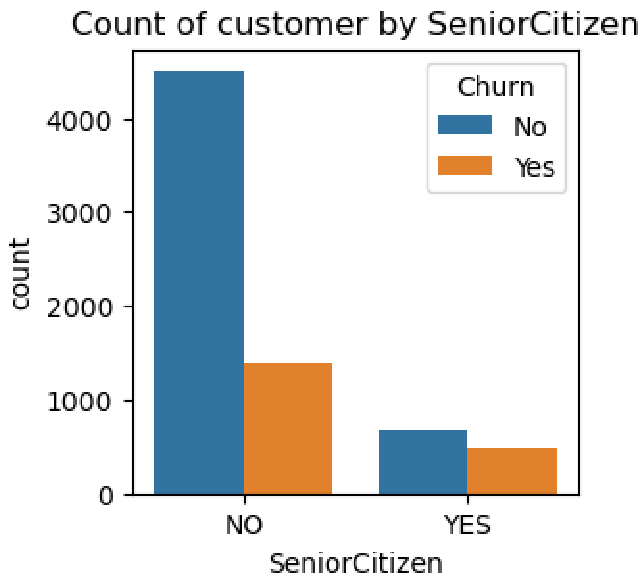
[54]:

```
plt.figure(figsize=(3,3))
sns.countplot(x='gender',data = df,hue='Churn')
plt.title('Count of customer by gender')
plt.show()
```



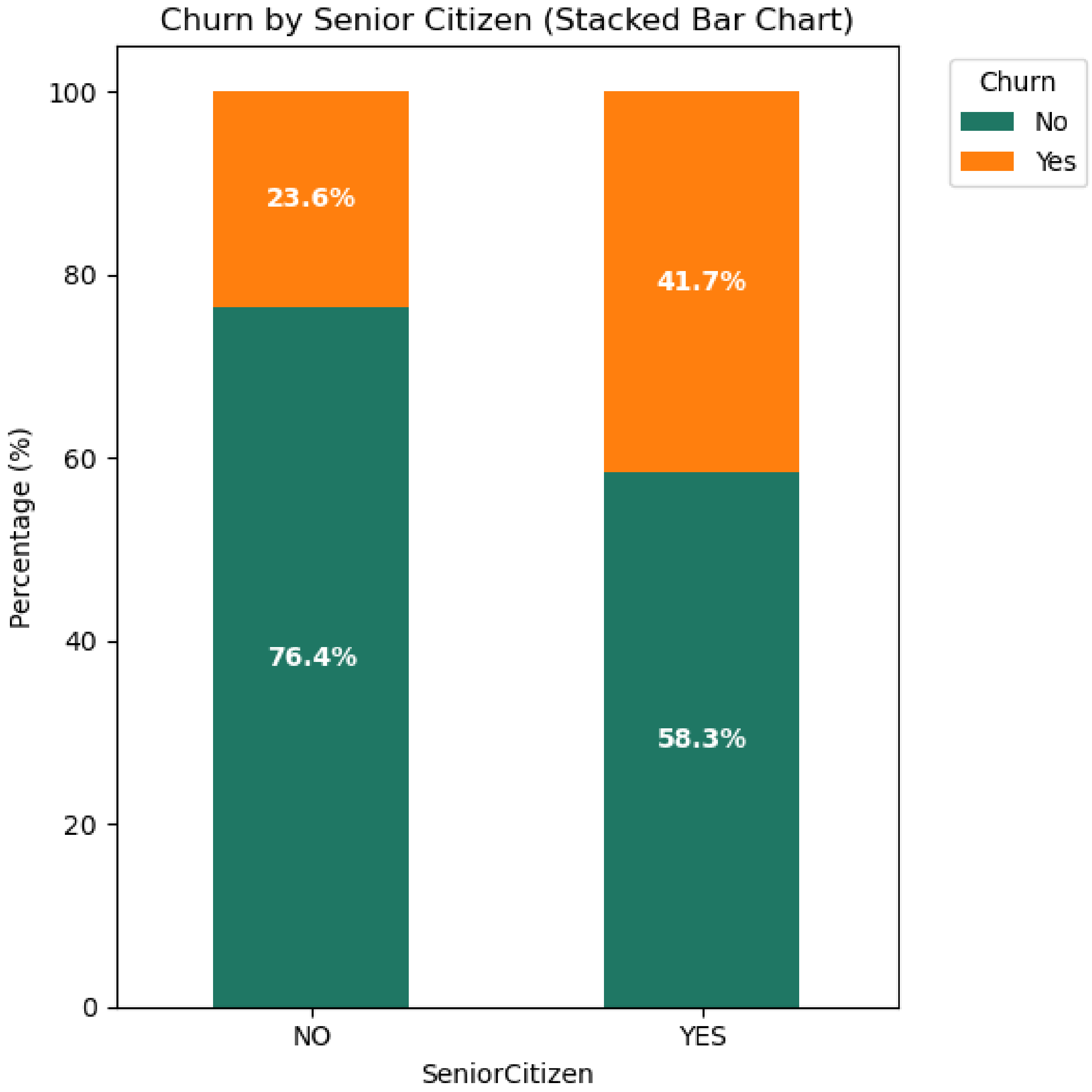
[55]:

```
plt.figure(figsize=(3,3))
sns.countplot(x='SeniorCitizen',data = df,hue='Churn')
plt.title('Count of customer by SeniorCitizen')
plt.show()
```

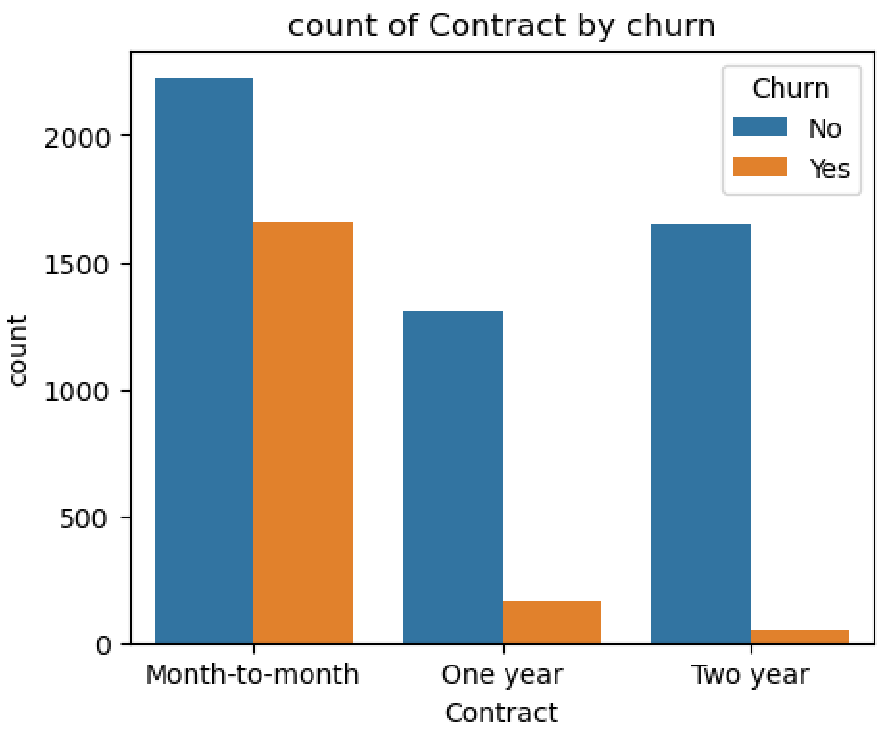


[59]:

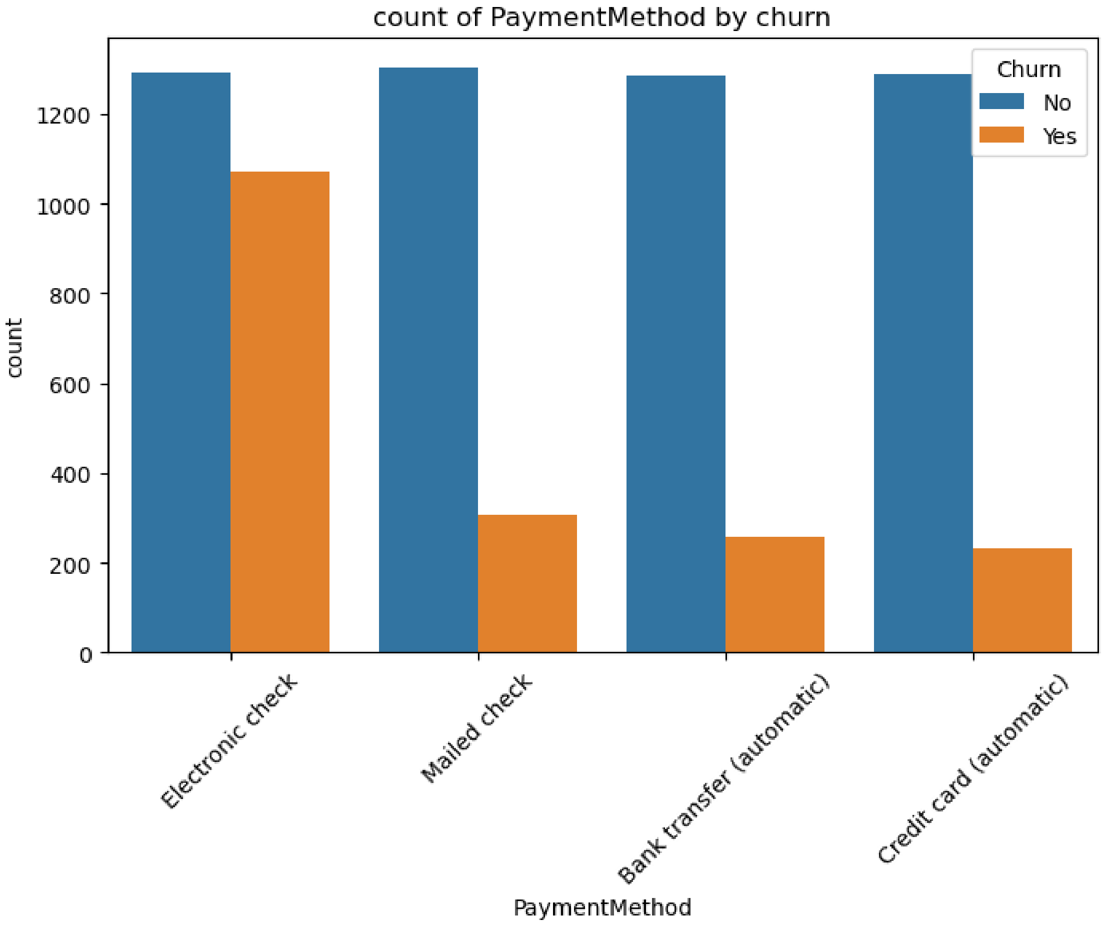
```
total_counts = df.groupby('SeniorCitizen')['Churn'].
 ↪value_counts(normalize=True).unstack() * 100

fig, ax = plt.subplots(figsize=(6,6))

total_counts.plot(kind='bar', stacked=True, ax=ax, color=['#1f7764', '#ff7f0e'])

# Add percentage labels inside the bars
for p in ax.patches:
    width, height = p.get_width(), p.get_height()
    x, y = p.get_xy()
    if height > 0:  # avoid showing 0.0% labels
        ax.text(x + width / 2, y + height / 2, f'{height:.1f}%',
                ha='center', va='center', color='white', fontsize=10,␣
 ↪fontweight='bold')

plt.title("Churn by Senior Citizen (Stacked Bar Chart)")
plt.xlabel("SeniorCitizen")
plt.ylabel("Percentage (%)")
plt.xticks(rotation=0)
plt.legend(title="Churn", bbox_to_anchor=(1.05, 1), loc='upper left')
plt.tight_layout()
plt.show()
```



[66]:
```
plt.figure(figsize=(5,4))
sns.countplot(x='Contract',data = df,hue='Churn')
plt.title('count of Contract by churn')
plt.show()
```



[76]:
```
plt.figure(figsize=(8,5))
sns.countplot(x='PaymentMethod',data = df,hue='Churn')
plt.title('count of PaymentMethod by churn')
plt.xticks(rotation = 45)
plt.show()
```

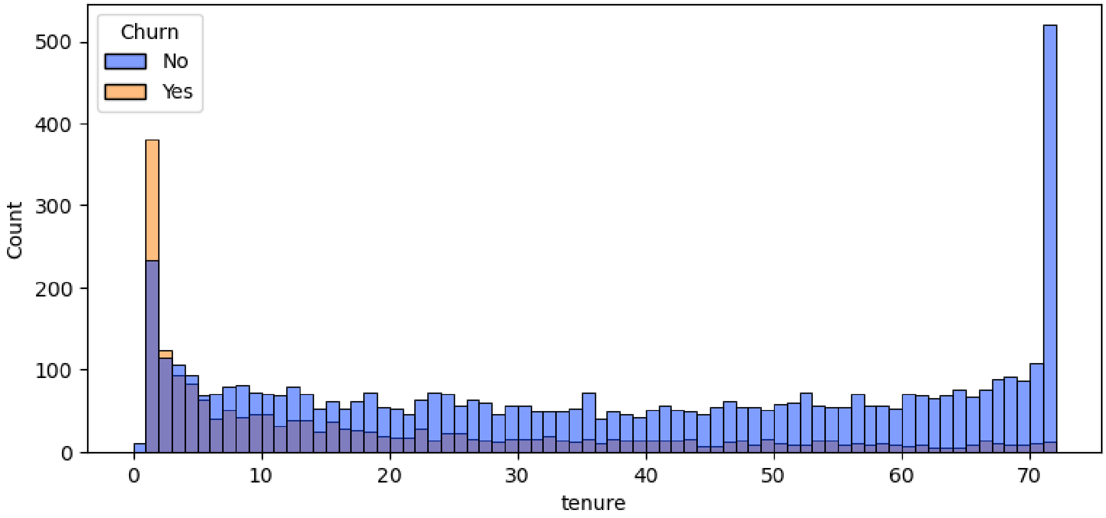


[74]:

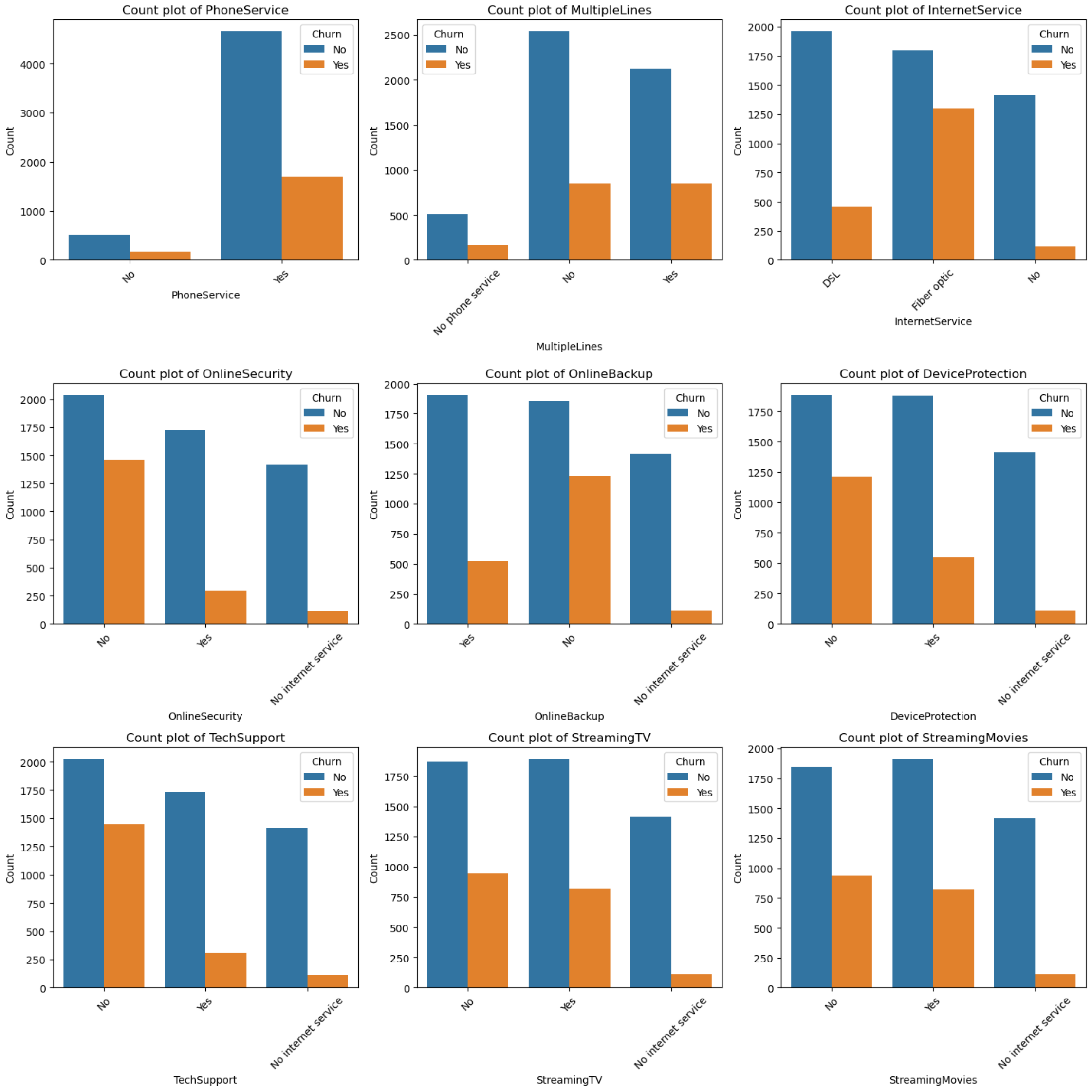
```
df = df.replace([np.inf, -np.inf], np.nan).dropna(subset=['tenure'])
```

[79]:

```
plt.figure(figsize = (9,4))
sns.histplot(x = 'tenure',data = df,bins = 72,hue ='Churn',palette= "bright")
plt.show()
```



[84]:
```
columns = ['PhoneService', 'MultipleLines', 'InternetService',
           'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
           'TechSupport', 'StreamingTV', 'StreamingMovies']

n_cols = 3
n_rows = (len(columns) + n_cols - 1) // n_cols  # ceiling division

fig, axes = plt.subplots(n_rows, n_cols, figsize=(15, n_rows * 5))
axes = axes.flatten()

# Create count plots
for i, col in enumerate(columns):
    sns.countplot(x=col, data=df, hue="Churn", ax=axes[i])
    axes[i].set_title(f'Count plot of {col}')
    axes[i].set_xlabel(col)
    axes[i].set_ylabel('Count')
    axes[i].tick_params(axis='x', rotation=45)

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

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


Count plot of PhoneService
Churn
No
Yes
Count
0
1000
2000
3000
4000
No
Yes
PhoneService
Count plot of MultipleLines
Churn
No
Yes
Count
0
500
1000
1500
2000
2500
No phone service
No
Yes
MultipleLines
Count plot of InternetService
Churn
No
Yes
Count
0
250
500
750
1000
1250
1500
1750
2000
DSL
Fiber optic
No
InternetService
Count plot of OnlineSecurity
Churn
No
Yes
Count
0
250
500
750
1000
1250
1500
1750
2000
No
Yes
No internet service
OnlineSecurity
Count plot of OnlineBackup
Churn
No
Yes
Count
0
250
500
750
1000
1250
1500
1750
2000
Yes
No
No internet service
OnlineBackup
Count plot of DeviceProtection
Churn
No
Yes
Count
0
250
500
750
1000
1250
1500
1750
No
Yes
No internet service
DeviceProtection
Count plot of TechSupport
Churn
No
Yes
Count
0
250
500
750
1000
1250
1500
1750
2000
No
Yes
No internet service
TechSupport
Count plot of StreamingTV
Churn
No
Yes
Count
0
250
500
750
1000
1250
1500
1750
No
Yes
No internet service
StreamingTV
Count plot of StreamingMovies
Churn
No
Yes
Count
0
250
500
750
1000
1250
1500
1750
2000
No
Yes
No internet service
StreamingMovies

[ ]: