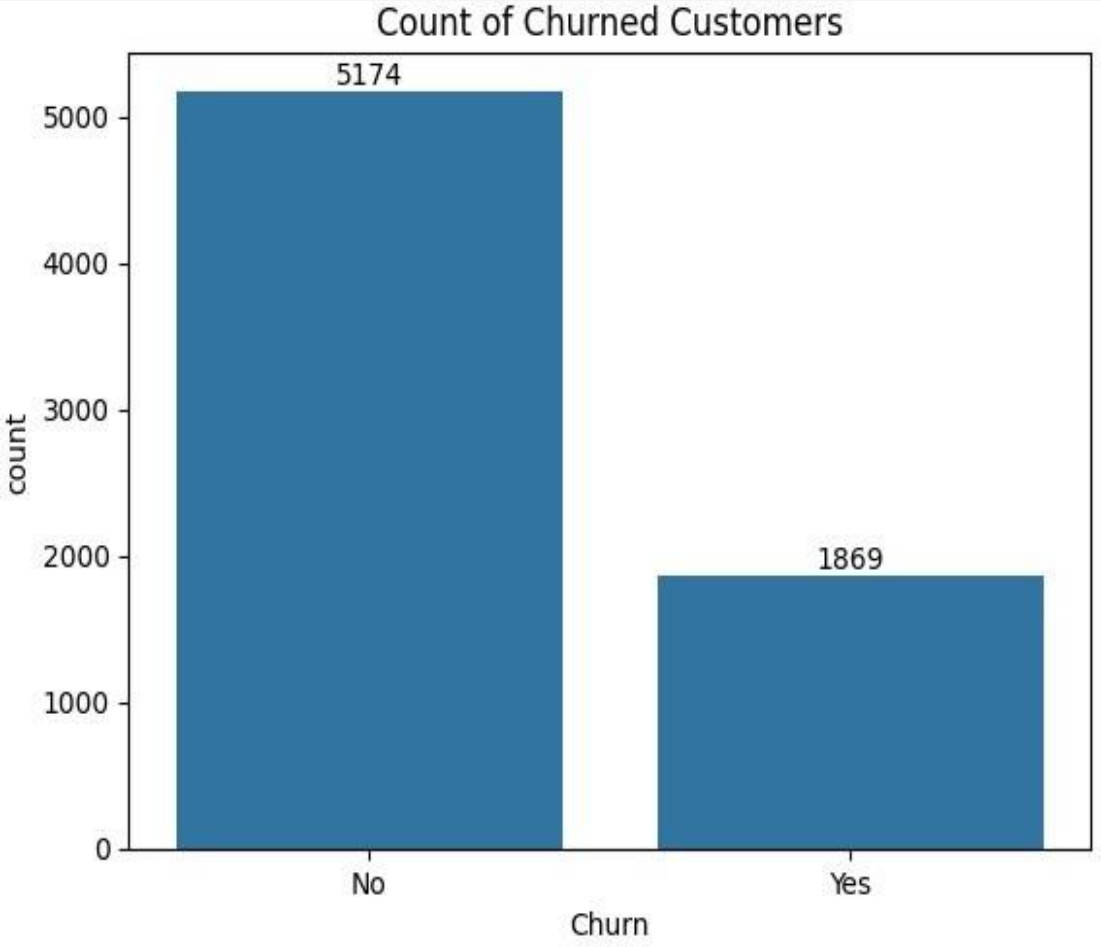
# Data Analysis, Cleaning, Transformation and Visualization

```
import pandas
as pdimport
numpy as np
import matplotlib.pyplot
as mpimport seaborn as sns

df=pd.read_csv('Customer 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
                                                 Cont
ractPaperlessBilling  \
0          No          No              No Month-to-
monthYes
1          No          No              No        One
yearNo
2          No          No              No Month-to-
monthYes
3         Yes          No              No        One
```

```
ax=sns.countplot( x = "Churn" , data = df )
ax.bar_label(ax.containers[0])
mp.title("Count of Churned Customers" , fontsize=12)
mp.show()
```



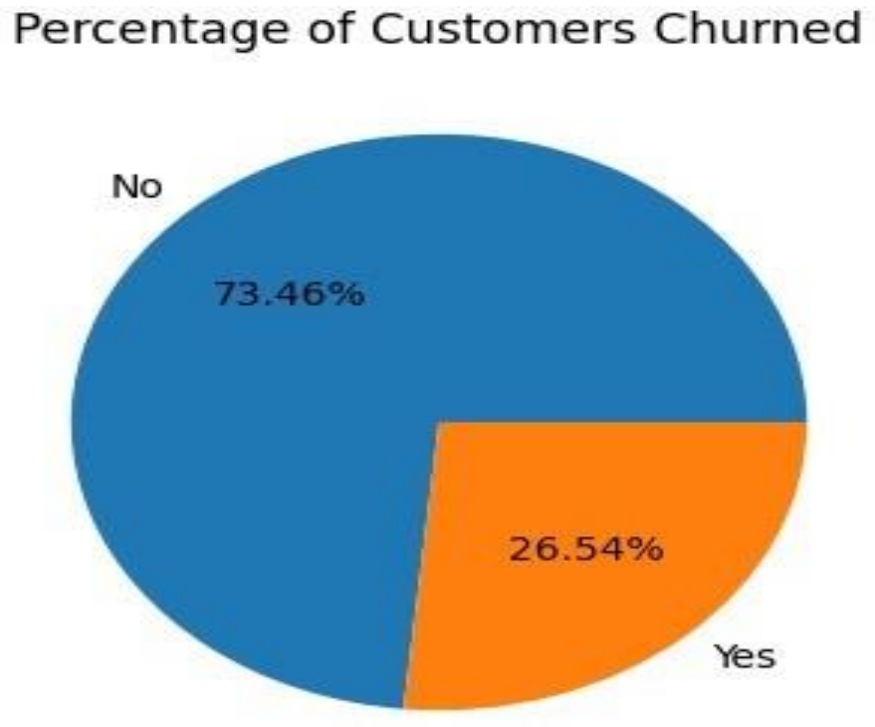

```
Churn                0
dtype: int64
```

```
df.describe()
```

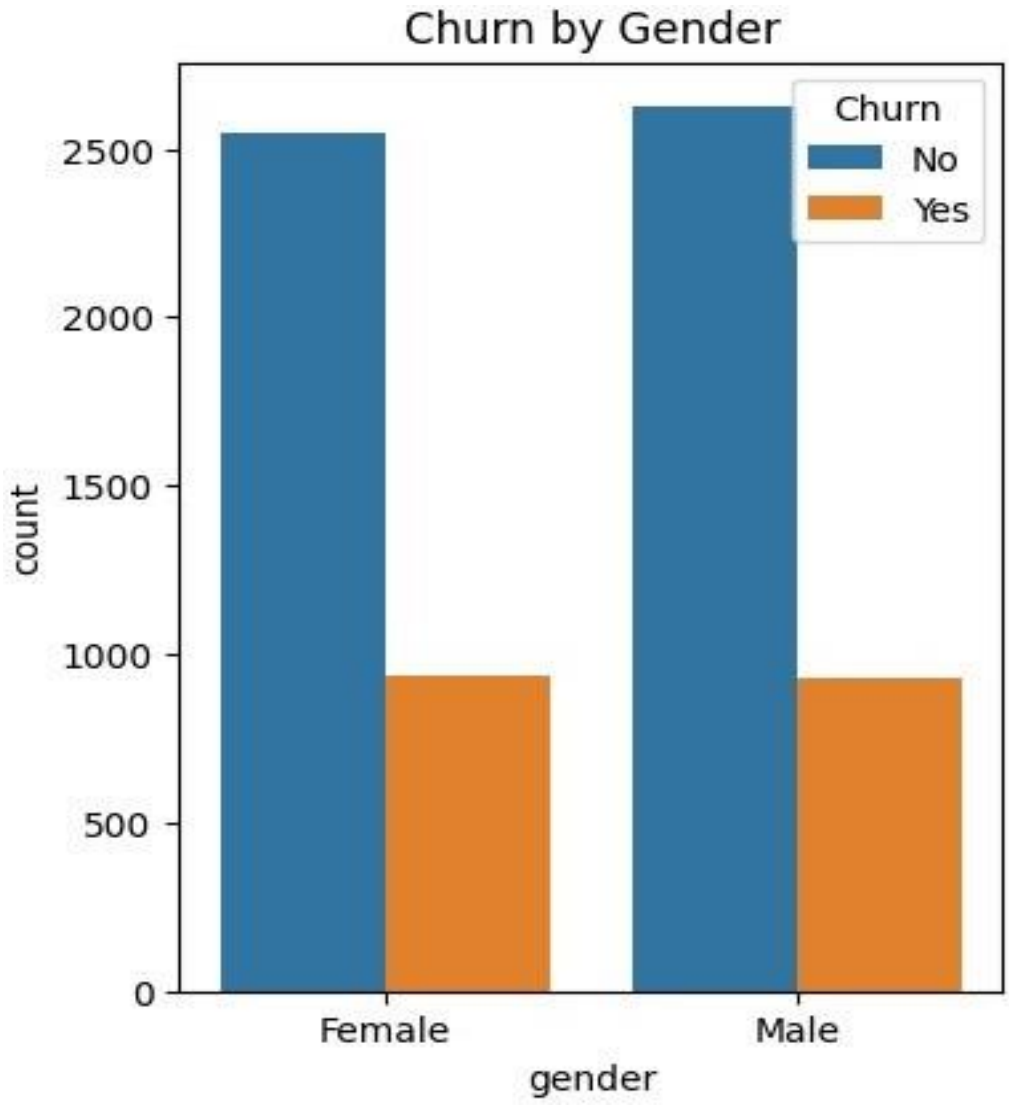
| | SeniorCitizen | tenure | MonthlyCharges | TotalCharges |
|---|---|---|---|---|
| count | 7043.000000 | 7043.000000 | 7043.000000 | 7043.000000 |
| mean | 0.162147 | 32.371149 | 64.761692 | 2279.734304 |
| std | 0.368612 | 24.559481 | 30.090047 | 2266.794470 |
| min | 0.000000 | 0.000000 | 18.250000 | 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.duplicated().s

um()0

def
    convert(val
    ue):if
    value==1:
        return
    "yes"else:
        return "no"

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

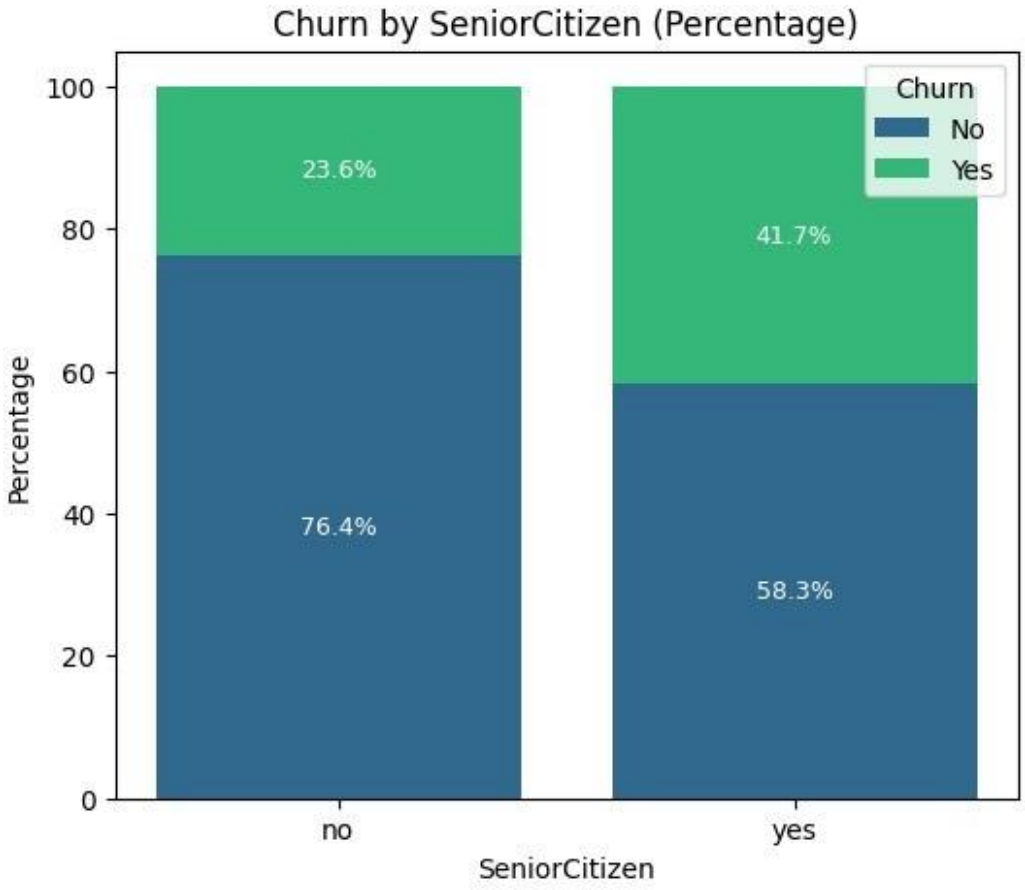
```
mp.figure(figsize=(4,4))
gb=df.groupby("Churn").agg({'Churn' : "count"})
mp.pie(gb['Churn'] , labels=gb.index , autopct =
"%1.2f%%")mp.title("Percentage of Customers Churned",
fontsize=12) mp.show()
```



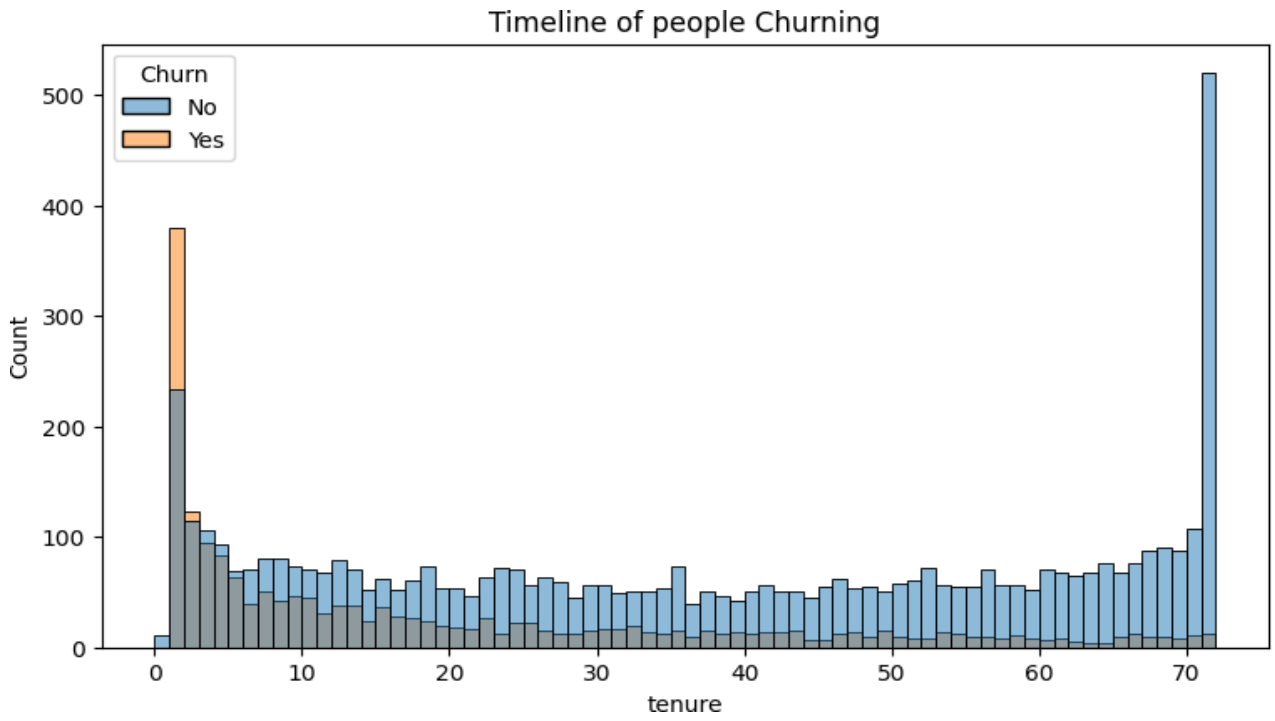
```
mp.figure(figsize=(4,5))
sns.countplot(x="gender", data= df ,
hue="Churn")mp.title("Churn by Gender")
mp.show()
```



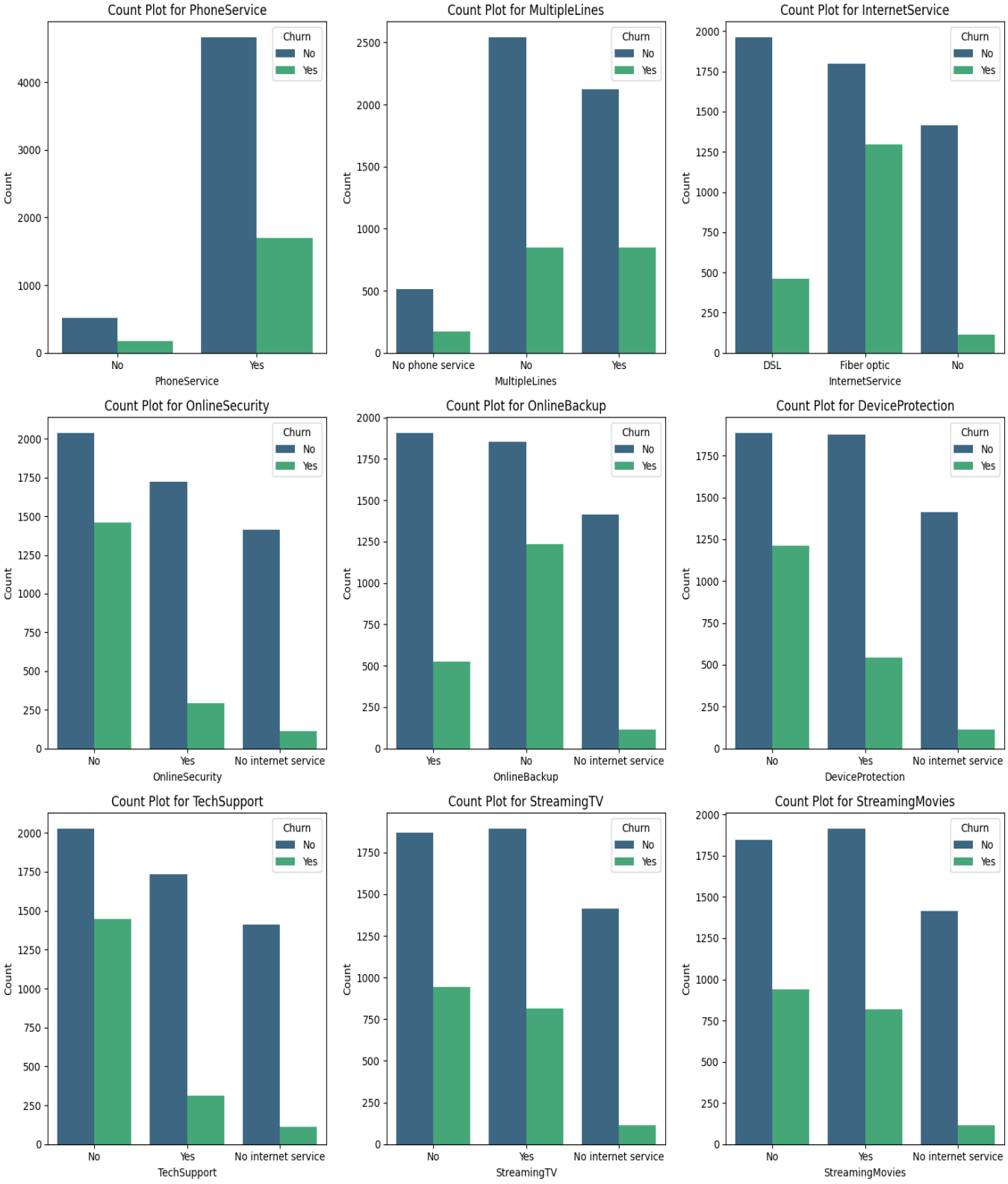
```
ax.set_title('Churn by SeniorCitizen (Percentage)')
ax.set_ylabel('Percentage') ax.set_xlabel('SeniorCitizen')
ax.set_xticks(range(len(pivot.index))) ax.set_xticklabels(pivot.index,
rotation=0) horizontal
ax.legend(title='Churn', loc='upper right')mp.show()
```




```
mp.figure(figsize=(9,5))
sns.histplot(x="tenure" , data=df , hue ="Churn" , bins=72)
mp.title("Timeline of people Churning")
mp.show()
```




```
# Plot each column
for idx, col in enumerate(columns):
    sns.countplot(data=df, x=col, ax=axes[idx],
palette='viridis' ,hue=df["Churn"])  # Explicitly set hue=None
    axes[idx].set_title(f'Count Plot for {col}')
    axes[idx].set_xlabel(col)
    axes[idx].set_ylabel('Count')

# Remove any unused subplots
for idx in range(len(columns), len(axes)):
    fig.delaxes(axes[idx])  # Remove unused axes

# Adjust layout to avoid overlap
mp.tight_layo
ut()mp.show()
```


Count Plot for PhoneService
Churn
No
Yes
Count
PhoneService
Count Plot for MultipleLines
No phone service
MultipleLines
Count Plot for InternetService
DSL
Fiber optic
InternetService
Count Plot for OnlineSecurity
No internet service
OnlineSecurity
Count Plot for OnlineBackup
OnlineBackup
Count Plot for DeviceProtection
DeviceProtection
Count Plot for TechSupport
TechSupport
Count Plot for StreamingTV
StreamingTV
Count Plot for StreamingMovies
StreamingMovies