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

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

[3]:

	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection	TechSupport
0	7590-VHVEG	Female	0	Yes	No	1	No	No phone service	DSL	No	...	No	No
1	5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	...	Yes	No
2	3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	...	No	No
3	7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	Yes	...	Yes	Yes
4	9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	...	No	No

5 rows × 21 columns

[19]:
```
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
```

[22]:
```
df["TotalCharges"]=df["TotalCharges"].replace(" ","0")
df["TotalCharges"]=df["TotalCharges"].astype("float")
```

[23]:
```
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
```

[24]:
```
df.isnull().sum()
```

[24]:
```
customerID          0
gender              0
SeniorCitizen       0
Partner             0
Dependents          0
tenure              0
PhoneService        0
MultipleLines       0
InternetService     0
OnlineSecurity      0
OnlineBackup        0
DeviceProtection    0
TechSupport         0
StreamingTV         0
StreamingMovies     0
Contract            0
PaperlessBilling    0
PaymentMethod       0
MonthlyCharges      0
TotalCharges        0
Churn               0
dtype: int64
```

[26]:
```
df.describe()
```

[26]:

	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

[28]:
```
df.duplicated().sum()
```

[28]:
```
0
```

[29]:
```
df["customerID"].duplicated().sum()
```

[29]:
```
0
```

```
# convert the 0 and 1 int yes and no in SeniorCitizen column
```

[32]:
```
def convert(value):
    if value==1:
        return "yes"
    else:
        return "no"

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

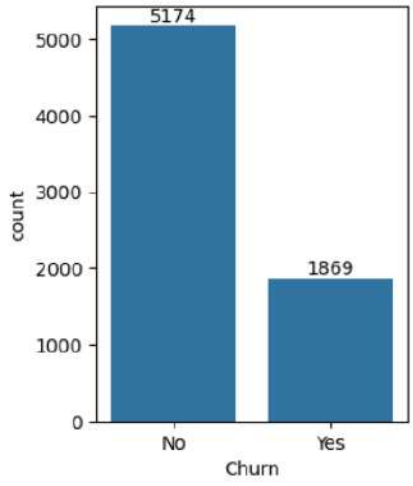
[38]:
```
df.head()
```

[38]:

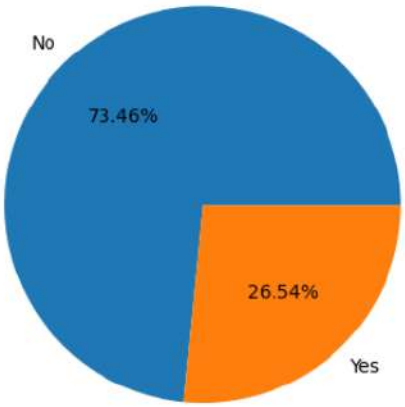
	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	...	DeviceProtection	TechSupport
0	7590-VHVEG	Female	no	Yes	No	1	No	No phone service	DSL	No	...	No	No
1	5575-GNVDE	Male	no	No	No	34	Yes	No	DSL	Yes	...	Yes	No
2	3668-QPYBK	Male	no	No	No	2	Yes	No	DSL	Yes	...	No	No
3	7795-CFOCW	Male	no	No	No	45	No	No phone service	DSL	Yes	...	Yes	Yes
4	9237-HQITU	Female	no	No	No	2	Yes	No	Fiber optic	No	...	No	No

5 rows × 21 columns

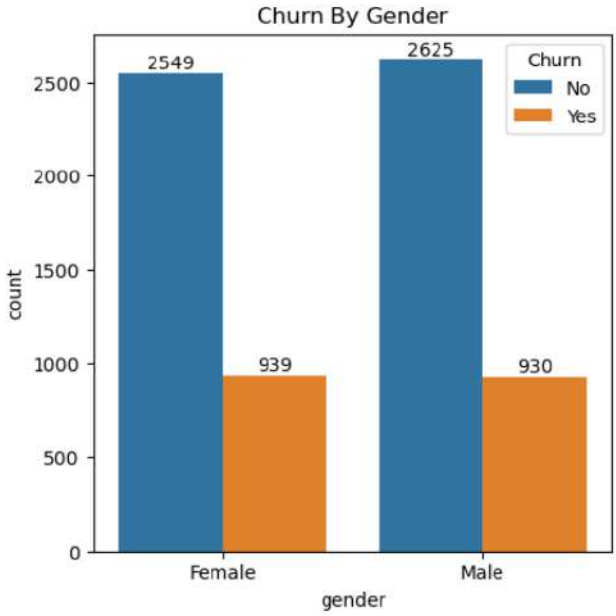
[77]:
```
plt.figure(figsize=(3,4))
ax=sns.countplot(x="Churn",data=df)
ax.bar_label(ax.containers[0])
plt.show()
```



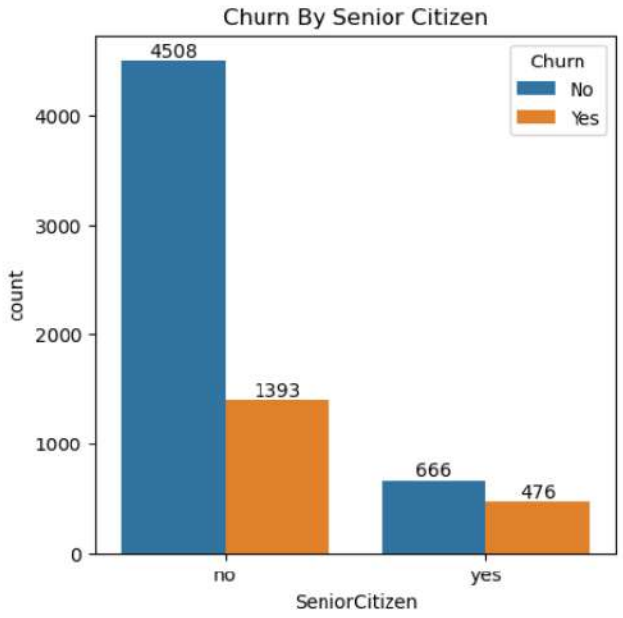
[43]:
```
gb=df.groupby("Churn").agg({"Churn":"count"})
plt.pie(gb["Churn"],labels= gb.index,autopct="%1.2f%%")
plt.show()
```



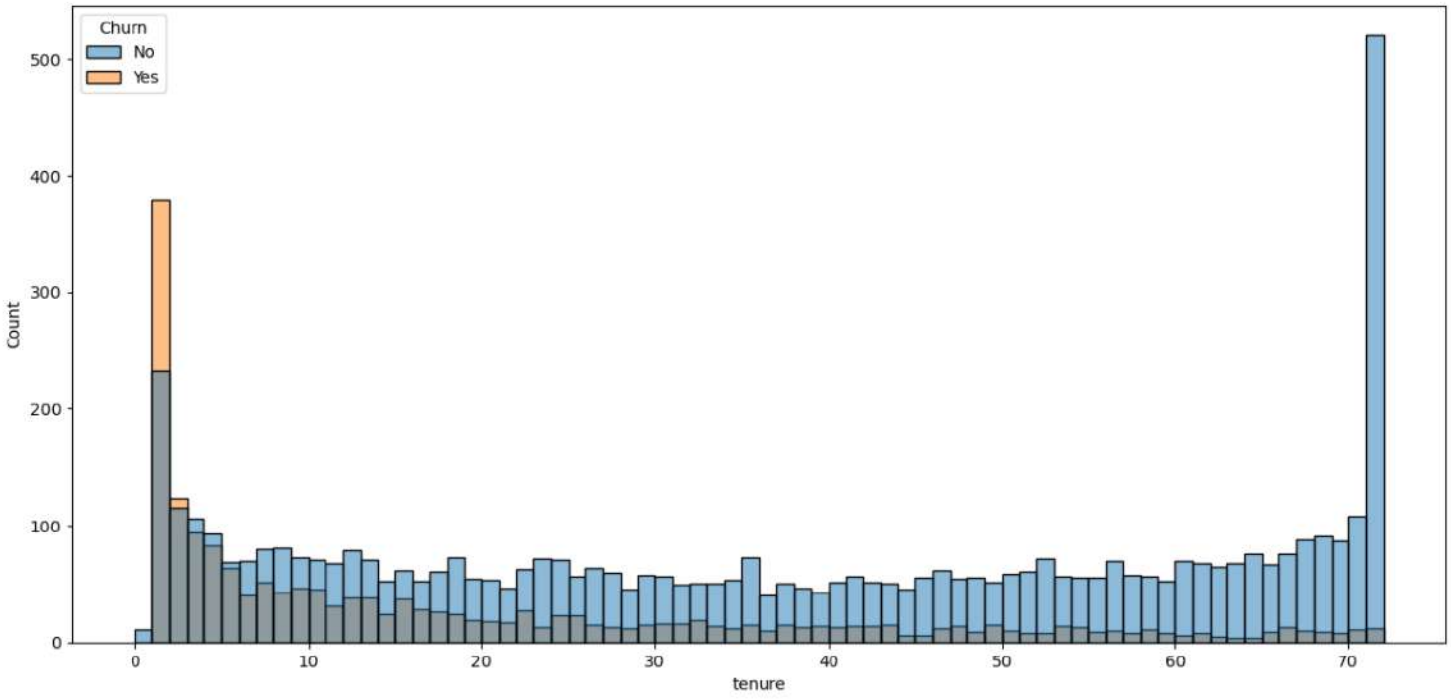
[80]:
```
plt.figure(figsize=(5,5))
ax=sns.countplot(x="gender",data=df,hue='Churn')
ax.bar_label(ax.containers[0])
ax.bar_label(ax.containers[1])
plt.title("Churn By Gender")
plt.show()
```



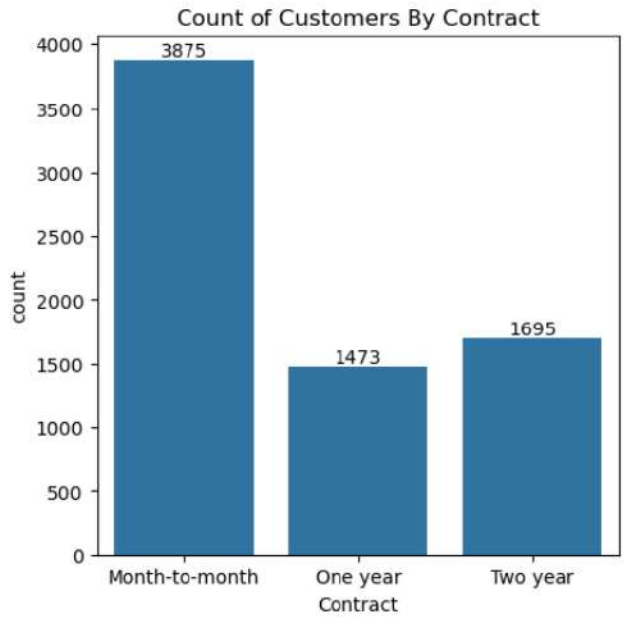
[82]:
```
plt.figure(figsize=(5,5))
ax=sns.countplot(x="SeniorCitizen",data=df,hue='Churn')
ax.bar_label(ax.containers[0])
ax.bar_label(ax.containers[1])
plt.title("Churn By Senior Citizen")
plt.show()
```



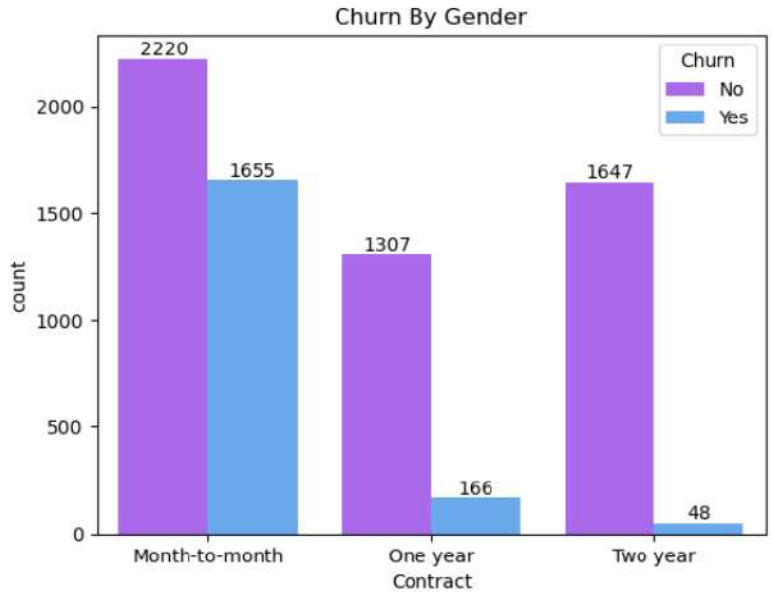
[61]:
```
plt.figure(figsize=(15,7))
sns.histplot(x="tenure",data=df,bins=72,hue="Churn")
plt.show()
```



[83]:
```
plt.figure(figsize=(5,5))
ax=sns.countplot(x="Contract",data=df)
ax.bar_label(ax.containers[0])

plt.title("Count of Customers By Contract")
plt.show()
```



[66]:
```
ax=sns.countplot(x="Contract",data=df,hue='Churn',palette="cool_r")
ax.bar_label(ax.containers[0])
ax.bar_label(ax.containers[1])
plt.title("Churn By Gender")
plt.show()
```

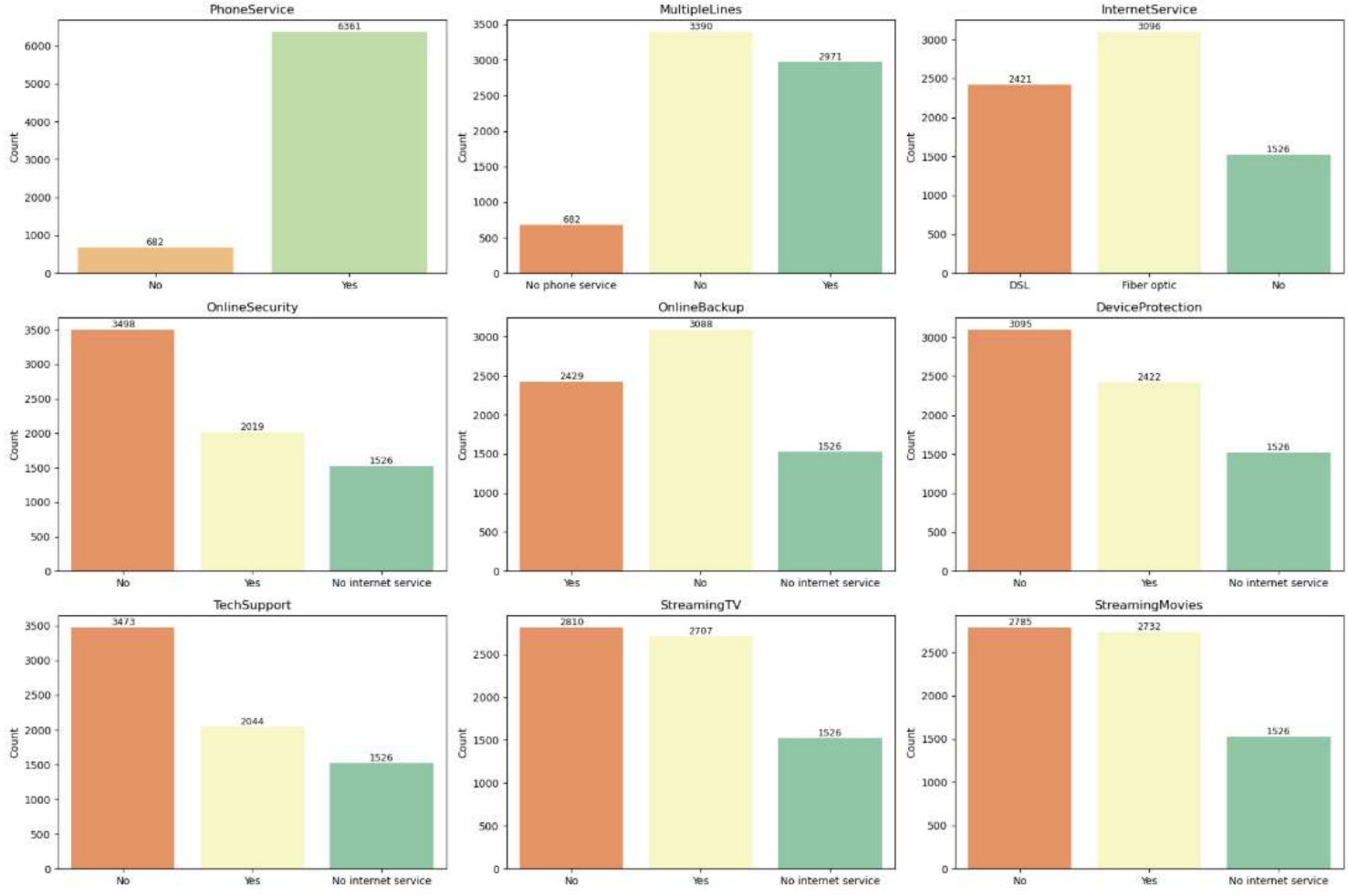


[67]:
```
df.columns.values
```

[67]:
```
array(['customerID', 'gender', 'SeniorCitizen', 'Partner', 'Dependents',
       'tenure', 'PhoneService', 'MultipleLines', 'InternetService',
       'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
       'TechSupport', 'StreamingTV', 'StreamingMovies', 'Contract',
       'PaperlessBilling', 'PaymentMethod', 'MonthlyCharges',
       'TotalCharges', 'Churn'], dtype=object)
```

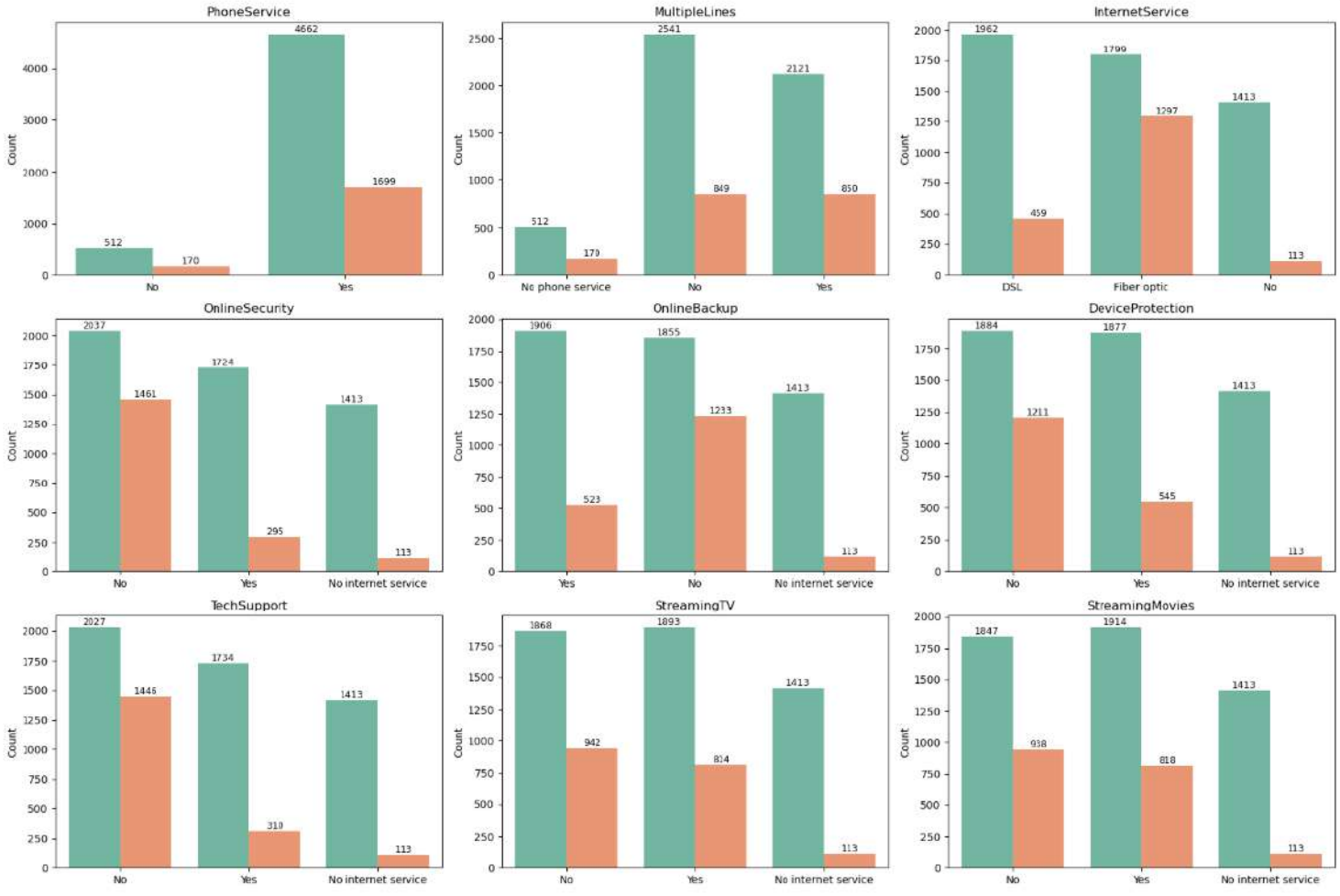
Services Taken by Customers

[73]:
```
cols = [
    'PhoneService', 'MultipleLines', 'InternetService',
    'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
    'TechSupport', 'StreamingTV', 'StreamingMovies'
]

n = len(cols)
rows = 3
cols_per_row = 3

fig, axes = plt.subplots(rows, cols_per_row, figsize=(18, 12))
axes = axes.flatten()

for i, col in enumerate(cols):
    ax = axes[i]
    bars = sns.countplot(data=df, x=col, ax=ax, hue=col, legend=False, palette="Spectral")
    ax.set_title(col, fontsize=12)
    ax.set_xlabel("")
    ax.set_ylabel("Count")

    # Add labels (counts) on top of bars
    for container in ax.containers:
        ax.bar_label(container, fmt="%d", label_type="edge", fontsize=9)

# Remove extra empty subplots
for j in range(i+1, rows*cols_per_row):
    fig.delaxes(axes[j])

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

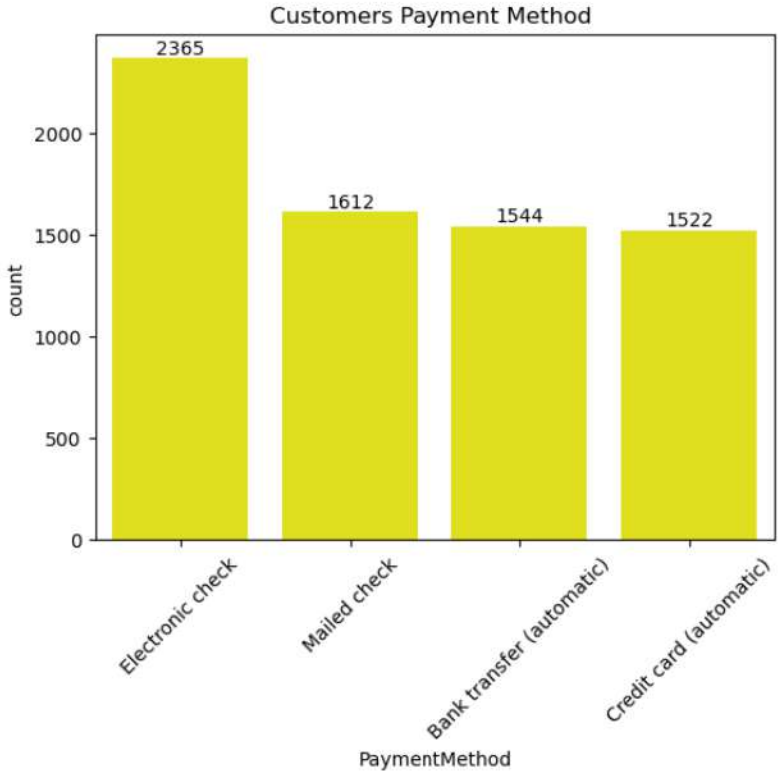


Services Taken by Customers Or how many of them Churn

[72]:
```
cols = [
    'PhoneService', 'MultipleLines', 'InternetService',
    'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
    'TechSupport', 'StreamingTV', 'StreamingMovies'
]

n = len(cols)
rows = 3
cols_per_row = 3

fig, axes = plt.subplots(rows, cols_per_row, figsize=(18, 12))
axes = axes.flatten()

for i, col in enumerate(cols):
    ax = axes[i]
    bars = sns.countplot(data=df, x=col, ax=ax, hue=df["Churn"], legend=False, palette="Set2")
    ax.set_title(col, fontsize=12)
    ax.set_xlabel("")
    ax.set_ylabel("Count")

    # Add labels (counts) on top of bars
    for container in ax.containers:
        ax.bar_label(container, fmt="%d", label_type="edge", fontsize=9)

# Remove extra empty subplots
for j in range(i+1, rows*cols_per_row):
    fig.delaxes(axes[j])

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



[10]:
```
ax=sns.countplot(x="PaymentMethod",data=df,color="yellow")
ax.bar_label(ax.containers[0])
plt.title("Customers Payment Method ")
plt.xticks(rotation=45)
plt.show()
```



[90]:
```
ax=sns.countplot(x="PaymentMethod",data=df,hue='Churn',palette="cool_r")
ax.bar_label(ax.containers[0])
ax.bar_label(ax.containers[1])
plt.title("Customer Churn By Payment Method ")
plt.xticks(rotation=45)
plt.show()
```