```
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.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
```

```
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.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)
```

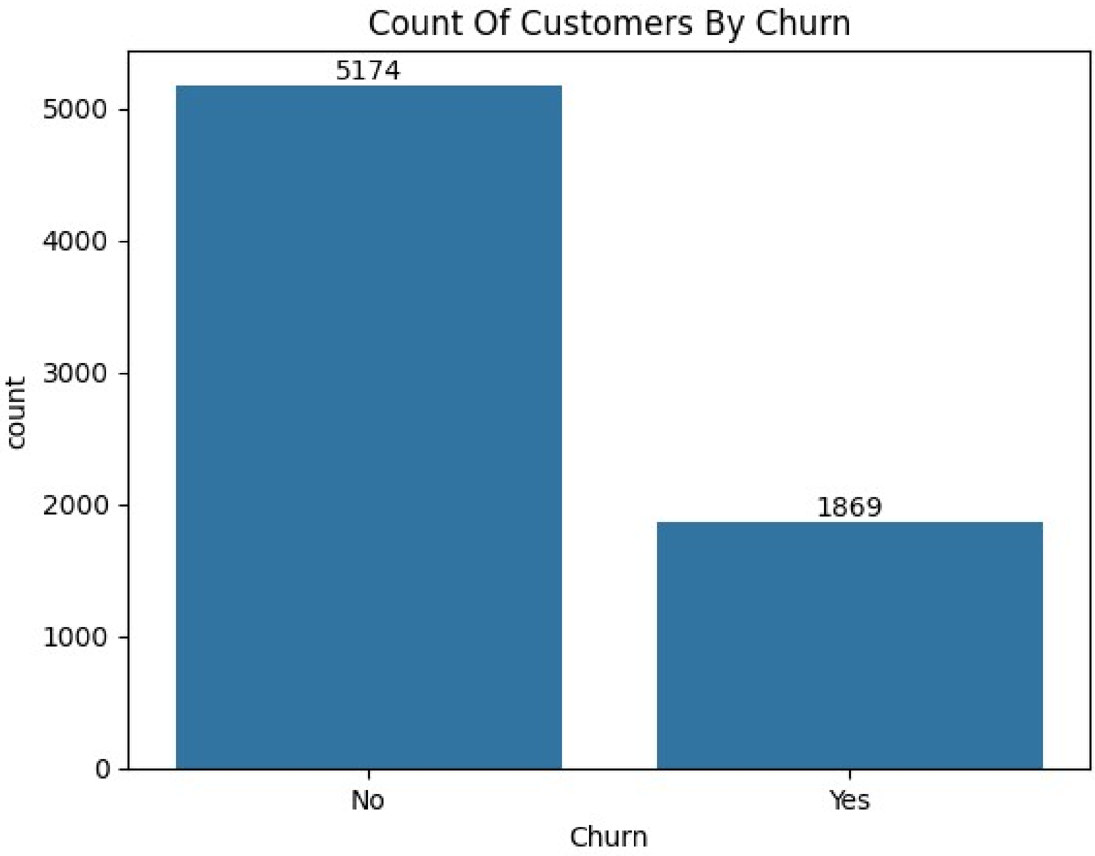
```
def cov(value):
    if value == 1:
        return "yes"
    else:
        return "no"

df['SeniorCitizen'] = df['SeniorCitizen'].apply(cov)

df.head(10)
```

```
   customerID  gender SeniorCitizen Partner Dependents  tenure 
PhoneService  \
0  7590-VHVEG  Female            no     Yes         No       1 
No   
1  5575-GNVDE    Male            no      No         No      34 
Yes   
2  3668-QPYBK    Male            no      No         No       2 
Yes   
3  7795-CFOCW    Male            no      No         No      45 
No   
4  9237-HQITU  Female            no      No         No       2 
Yes   
5  9305-CDSKC  Female            no      No         No       8 
```

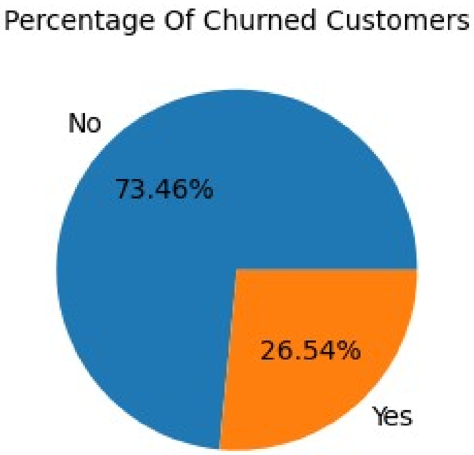
```
Yes
6  1452-KIOVK    Male            no      No        Yes       22
Yes
7  6713-OKOMC  Female            no      No         No       10
No
8  7892-POOKP  Female            no     Yes         No       28
Yes
9  6388-TABGU    Male            no      No        Yes       62
Yes

      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]
```

```python
ax=sns.countplot(x = "Churn", data = df)
ax.bar_label(ax.containers[0])
plt.title("Count Of Customers By Churn")
plt.show()
```

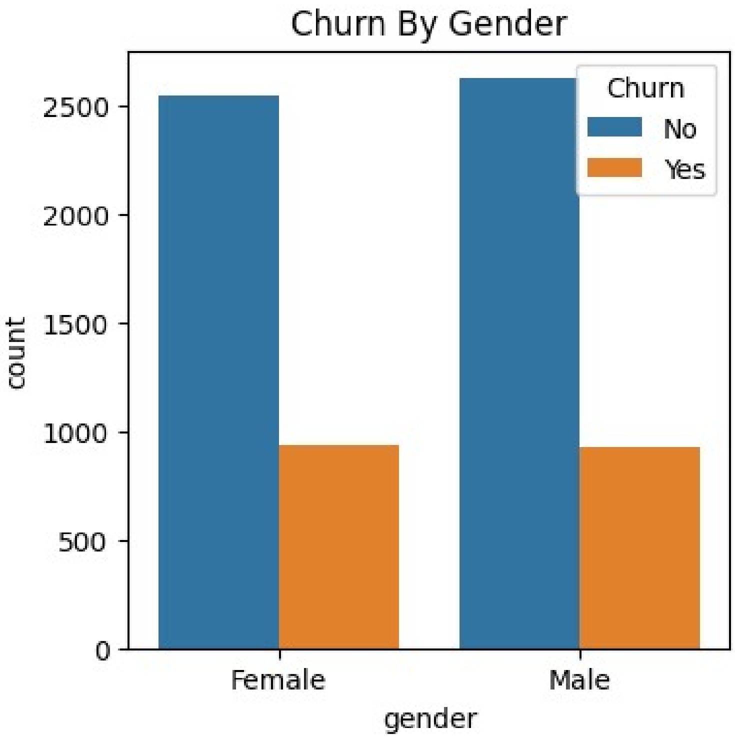


```
plt.figure(figsize=(3,4))
gb=df.groupby('Churn').agg({'Churn':'count'})
plt.pie(gb['Churn'], labels=gb.index, autopct="%1.2f%%")
plt.title("Percentage Of Churned Customers",fontsize = 10)
plt.show()
```



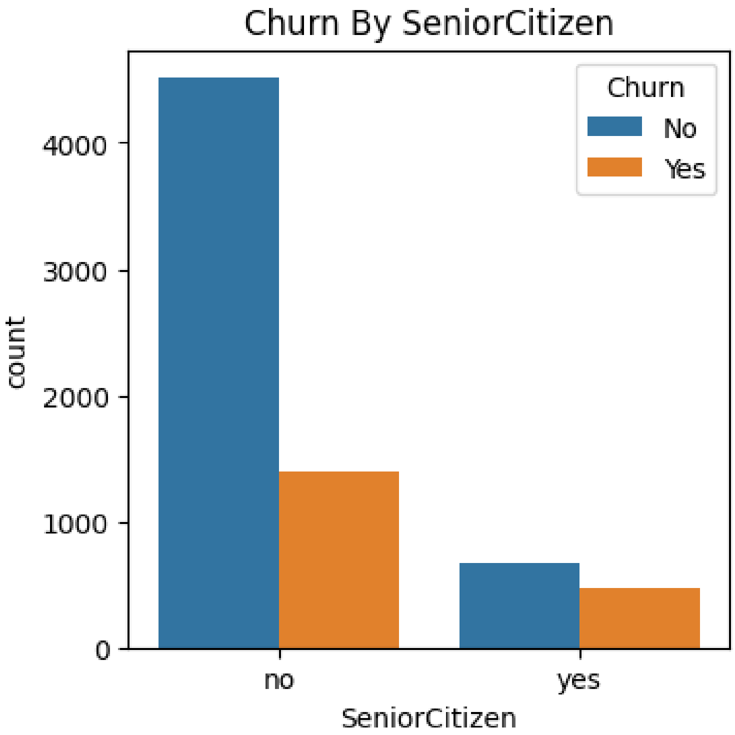
```
plt.figure(figsize=(4,4))
sns.countplot(x='gender', data = df, hue= "Churn")
plt.title("Churn By Gender")
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```

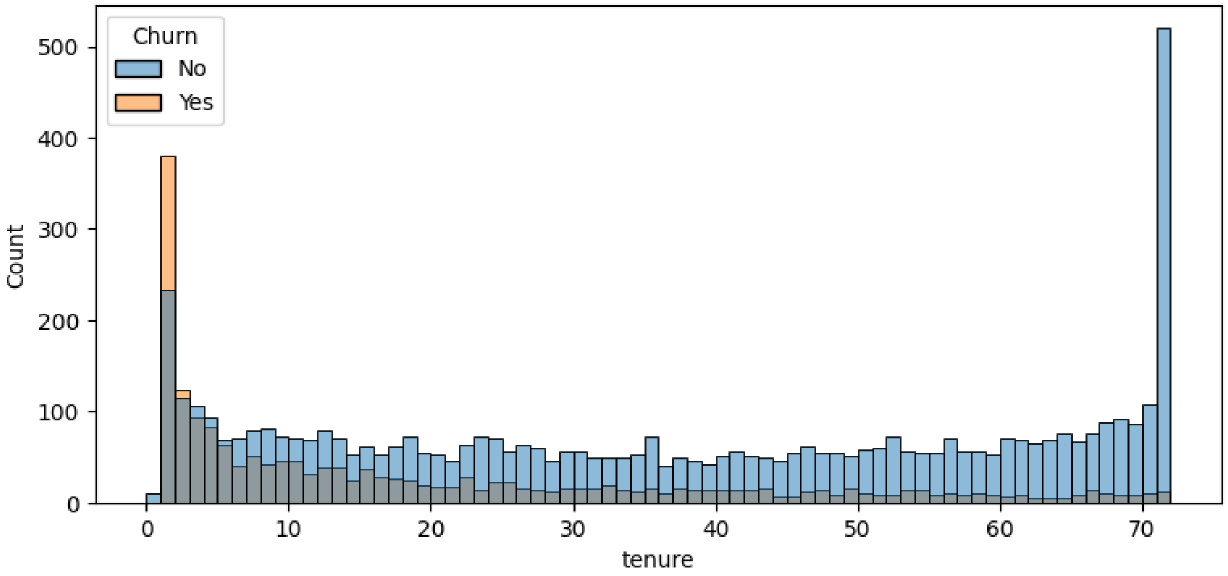


```
plt.figure(figsize=(4,4))
sns.countplot(x='SeniorCitizen', data = df, hue= "Churn")
plt.title("Churn By SeniorCitizen")
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```

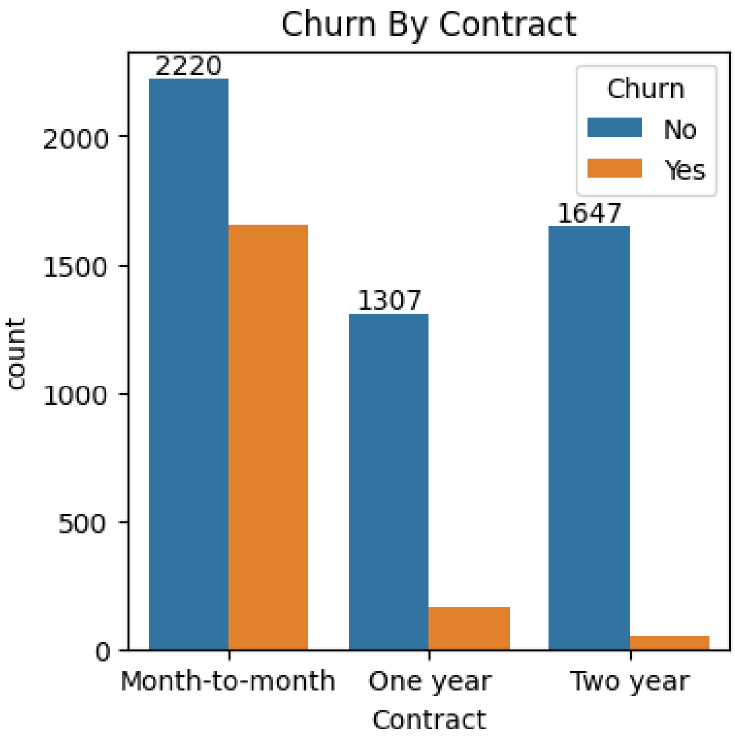


```
plt.figure(figsize=(9,4))
sns.histplot(x ="tenure",data = df,bins = 72 ,hue="Churn")
plt.show()
```



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

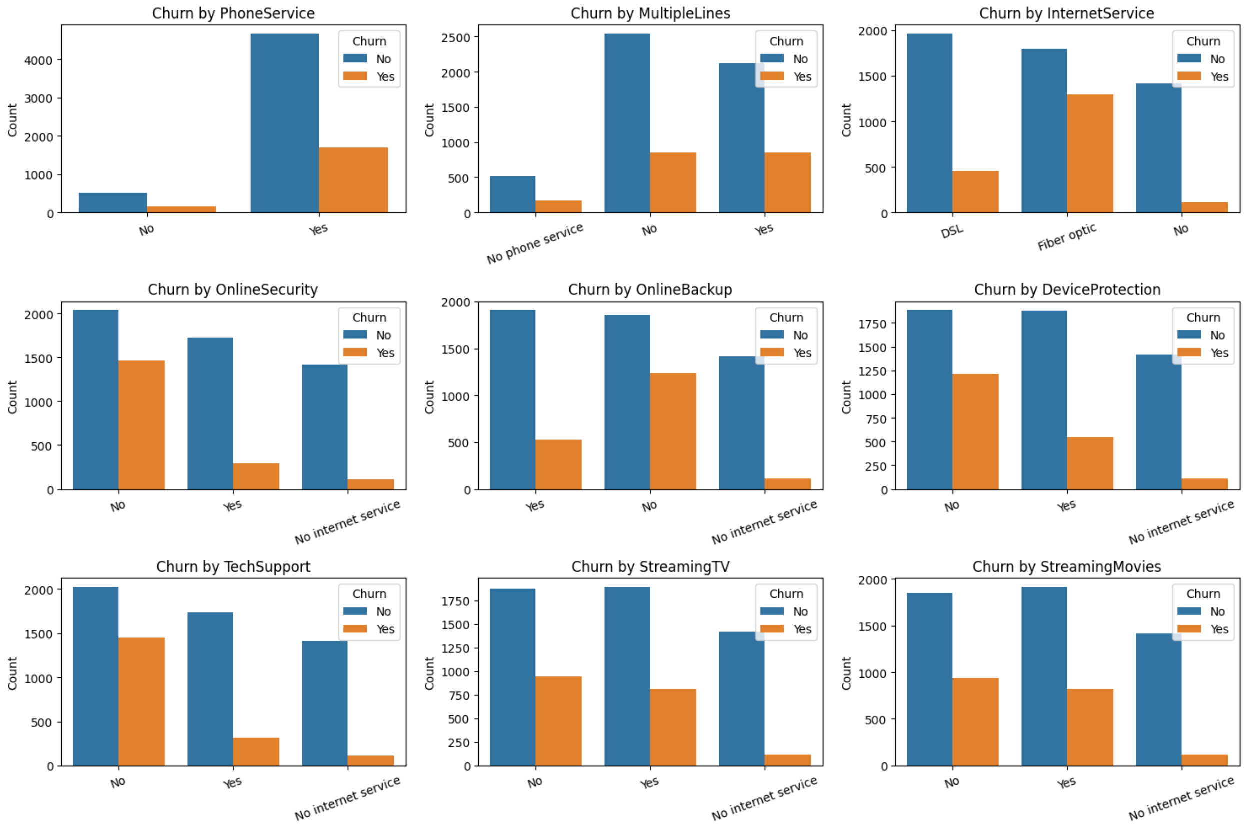
```
plt.figure(figsize=(4,4))
ax=sns.countplot(x='Contract', data = df ,hue="Churn")
ax.bar_label(ax.containers[0])
plt.title("Churn By Contract")
plt.show

<function matplotlib.pyplot.show(close=None, block=None)>
```



#people who have month to month contract are likely to churn then from those who have 1 or 2 year 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)
```

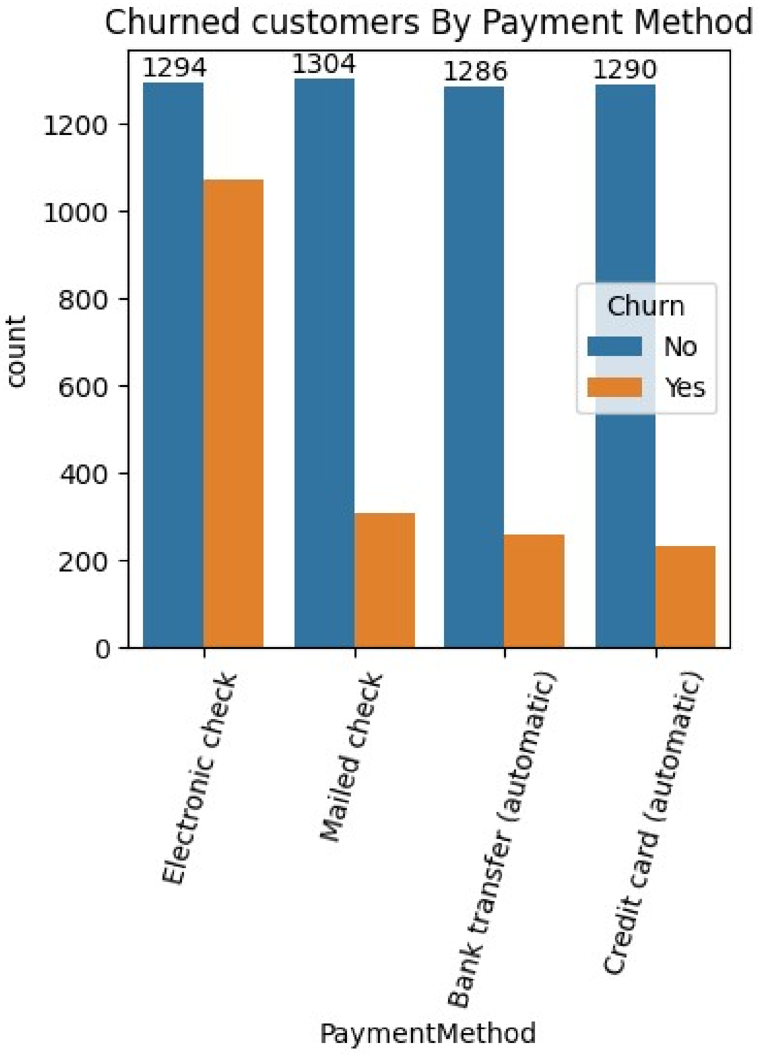
```
# List of categorical columns to plot
cols = [
    'PhoneService', 'MultipleLines', 'InternetService',
    'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
    'TechSupport', 'StreamingTV', 'StreamingMovies'
]

# Set up grid for subplots
n_cols = 3  # number of plots per row
n_rows = (len(cols) + n_cols - 1) // n_cols  # calculate required rows

plt.figure(figsize=(15, 10))

# Loop through columns and create subplots
for i, col in enumerate(cols, 1):
    plt.subplot(n_rows, n_cols, i)
    sns.countplot(data=df, x=col, hue='Churn')
    plt.title(f'Churn by {col}')
    plt.xlabel('')
    plt.ylabel('Count')
    plt.xticks(rotation=20)
    plt.legend(title='Churn', loc='upper right')

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



#Customers without internet-related services (like OnlineSecurity, TechSupport, or DeviceProtection) show noticeably higher churn rates. In contrast, those with services such as PhoneService or Streaming options tend to have lower churn. Overall, having additional online features or bundled services appears to reduce customer churn, while limited or no service options increase the likelihood of churn.

```
plt.figure(figsize=(4,4))
ax=sns.countplot(x='PaymentMethod', data = df ,hue="Churn")
ax.bar_label(ax.containers[0])
plt.title("Churned customers By Payment Method")
plt.xticks(rotation=75)
plt.show
```

```
<function matplotlib.pyplot.show(close=None, block=None)>
```



#customers islikely to churn when he is using electronic check as a payment method