In [5]:
```
!pip install matplotlib
!pip install seaborn
!pip install numpy
```

```
Requirement already satisfied: matplotlib in c:\users\tanay\anaconda3\lib\site-pac
kages (3.5.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\tanay\anaconda3\lib\site-
packages (from matplotlib) (9.0.1)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\tanay\anaconda3\lib\si
te-packages (from matplotlib) (3.0.4)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\tanay\anaconda3\lib\s
ite-packages (from matplotlib) (4.25.0)
Requirement already satisfied: numpy>=1.17 in c:\users\tanay\anaconda3\lib\site-pa
ckages (from matplotlib) (1.21.5)
Requirement already satisfied: packaging>=20.0 in c:\users\tanay\anaconda3\lib\sit
e-packages (from matplotlib) (21.3)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\tanay\anaconda3\li
b\site-packages (from matplotlib) (2.8.2)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\tanay\anaconda3\lib\s
ite-packages (from matplotlib) (1.3.2)
Requirement already satisfied: cycler>=0.10 in c:\users\tanay\anaconda3\lib\site-p
ackages (from matplotlib) (0.11.0)
Requirement already satisfied: six>=1.5 in c:\users\tanay\anaconda3\lib\site-packa
ges (from python-dateutil>=2.7->matplotlib) (1.16.0)
Requirement already satisfied: seaborn in c:\users\tanay\anaconda3\lib\site-packag
es (0.11.2)
Requirement already satisfied: matplotlib>=2.2 in c:\users\tanay\anaconda3\lib\sit
e-packages (from seaborn) (3.5.1)
Requirement already satisfied: pandas>=0.23 in c:\users\tanay\anaconda3\lib\site-p
ackages (from seaborn) (1.4.2)
Requirement already satisfied: scipy>=1.0 in c:\users\tanay\anaconda3\lib\site-pac
kages (from seaborn) (1.7.3)
Requirement already satisfied: numpy>=1.15 in c:\users\tanay\anaconda3\lib\site-pa
ckages (from seaborn) (1.21.5)
Requirement already satisfied: cycler>=0.10 in c:\users\tanay\anaconda3\lib\site-p
ackages (from matplotlib>=2.2->seaborn) (0.11.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\tanay\anaconda3\lib\s
ite-packages (from matplotlib>=2.2->seaborn) (1.3.2)
Requirement already satisfied: pillow>=6.2.0 in c:\users\tanay\anaconda3\lib\site-
packages (from matplotlib>=2.2->seaborn) (9.0.1)
Requirement already satisfied: packaging>=20.0 in c:\users\tanay\anaconda3\lib\sit
e-packages (from matplotlib>=2.2->seaborn) (21.3)
Requirement already satisfied: pyparsing>=2.2.1 in c:\users\tanay\anaconda3\lib\si
te-packages (from matplotlib>=2.2->seaborn) (3.0.4)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\tanay\anaconda3\li
b\site-packages (from matplotlib>=2.2->seaborn) (2.8.2)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\tanay\anaconda3\lib\s
ite-packages (from matplotlib>=2.2->seaborn) (4.25.0)
Requirement already satisfied: pytz>=2020.1 in c:\users\tanay\anaconda3\lib\site-p
ackages (from pandas>=0.23->seaborn) (2021.3)
Requirement already satisfied: six>=1.5 in c:\users\tanay\anaconda3\lib\site-packa
ges (from python-dateutil>=2.7->matplotlib>=2.2->seaborn) (1.16.0)
Requirement already satisfied: numpy in c:\users\tanay\anaconda3\lib\site-packages
(1.21.5)
```

In [3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [6]:
```
df = pd.read_csv("F:\Customer Churn.csv")
df.head()
```

Out[6]:

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

5 rows × 21 columns

In [7]:
```
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
```

In [8]:
```
df["TotalCharges"] = df["TotalCharges"].replace(" ",0)
df["TotalCharges"] = df["TotalCharges"].astype("float")
```

In [9]:
```
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
```

In [10]:
```
df.isnull().sum()
```

Out[10]:
```
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
```

In [11]:
```
df.describe()
```

Out[11]:

	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

In [12]:
```
df["customerID"].duplicated().sum()
```

Out[12]:
```
0
```

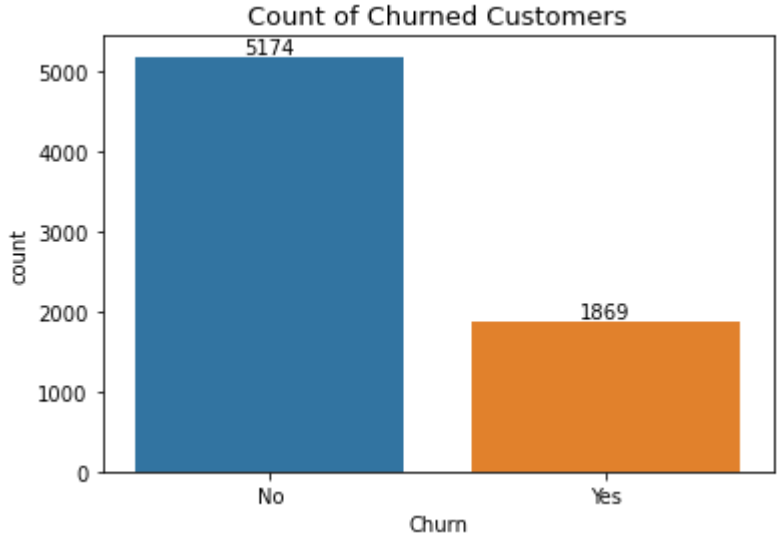
In [13]:
```
df["SeniorCitizen"] = df["SeniorCitizen"].replace(1,"yes")
df["SeniorCitizen"] = df["SeniorCitizen"].replace(0,"no")
df
```

Out[13]:

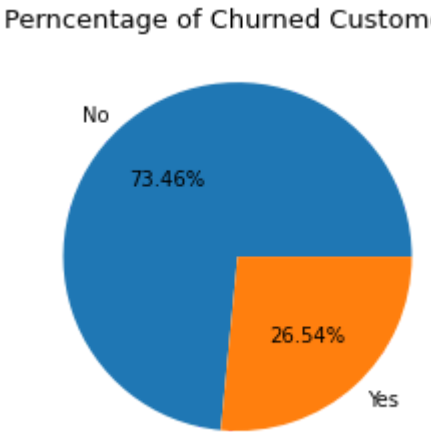
	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLin
0	7590-VHVEG	Female	no	Yes	No	1	No	No phor servi
1	5575-GNVDE	Male	no	No	No	34	Yes	N
2	3668-QPYBK	Male	no	No	No	2	Yes	N
3	7795-CFOCW	Male	no	No	No	45	No	No phor servi
4	9237-HQITU	Female	no	No	No	2	Yes	N
...	...	...	...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

7043 rows × 21 columns

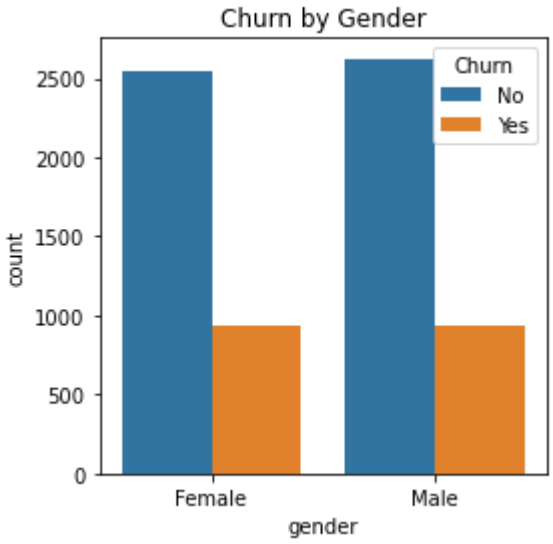
In [14]:
```
ax = sns.countplot(x = "Churn",data = df)
ax.bar_label(ax.containers[0])
plt.title('Count of Churned Customers',fontsize=13)
plt.show()
```



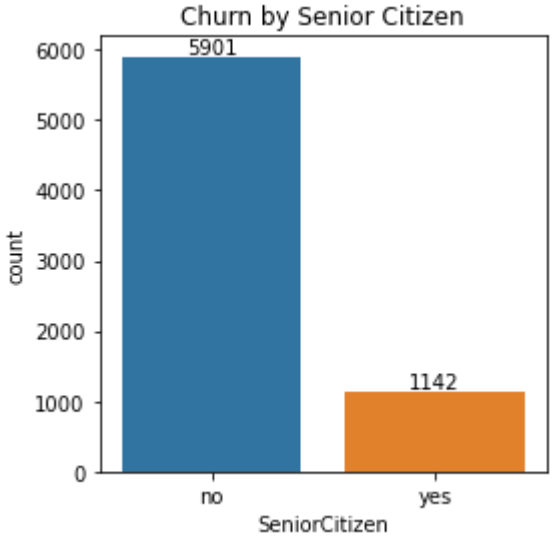
In [15]:
```
gb = df.groupby("Churn").agg({"Churn":"count"})
plt.pie(gb["Churn"],labels = gb.index,autopct = "%1.2f%%")
plt.title("Perncentage of Churned Customers",fontsize=13)
plt.show()
```



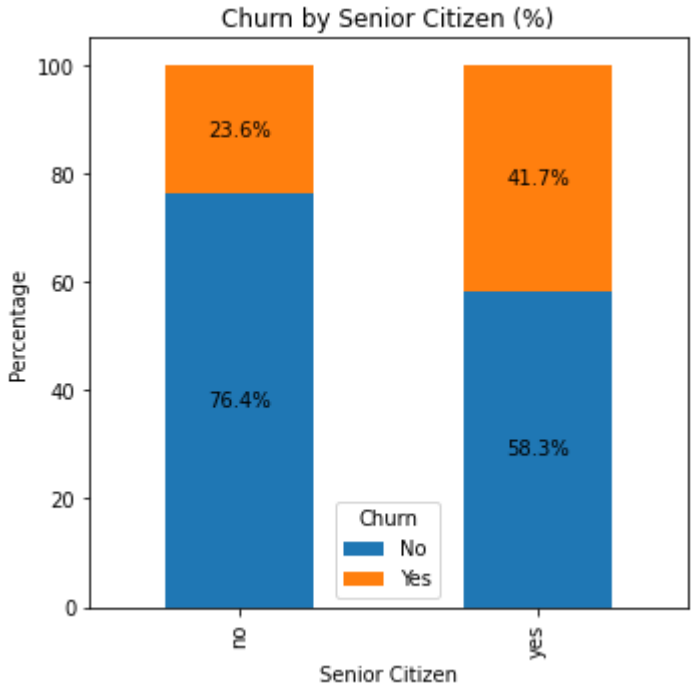
In [16]:
```
plt.figure(figsize=(4,4))
sns.countplot(x = 'gender',data = df,hue="Churn")
plt.title("Churn by Gender")
plt.show()
```



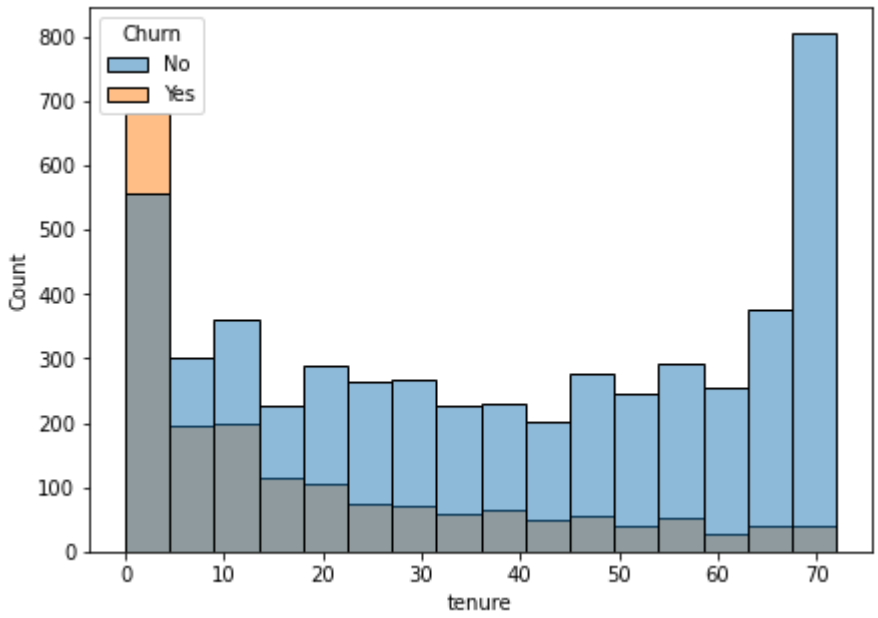
In [20]:
```
plt.figure(figsize=(4,4))
bx = sns.countplot(x = 'SeniorCitizen',data = df)
bx.bar_label(bx.containers[0])
plt.title("Churn by Senior Citizen")
plt.show()
```



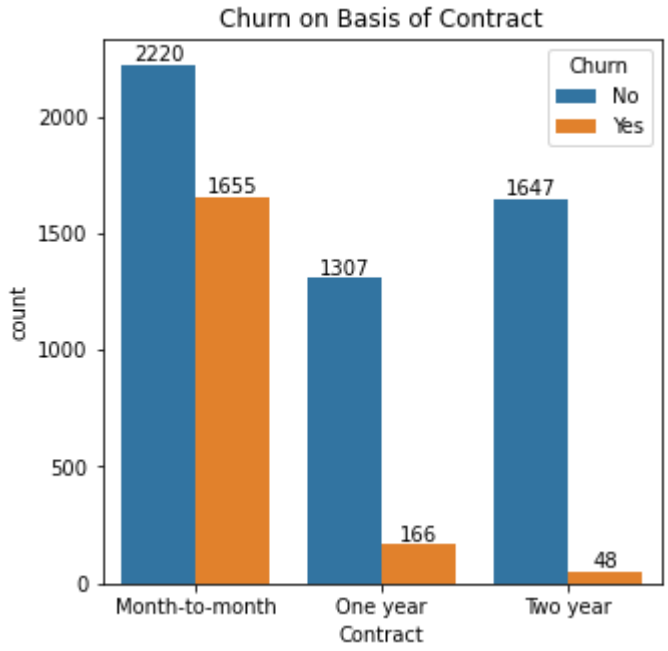
In [18]:
```
counts = pd.crosstab(df['SeniorCitizen'], df['Churn'])

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

# Plot 100% stacked bar chart
ax = percentages.plot(
    kind='bar',
    stacked=True,
    figsize=(5, 5)
)

# Titles and labels
plt.title("Churn by Senior Citizen (%)")
plt.xlabel("Senior Citizen")
plt.ylabel("Percentage")
plt.legend(title="Churn")

# Add percentage labels inside bars
for container in ax.containers:
    ax.bar_label(
        container,
        fmt='%.1f%%',
        label_type='center'
    )

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



In [19]:
```
plt.figure(figsize=(7,5))
sns.histplot(x = "tenure",data = df,hue="Churn")
plt.show()
```



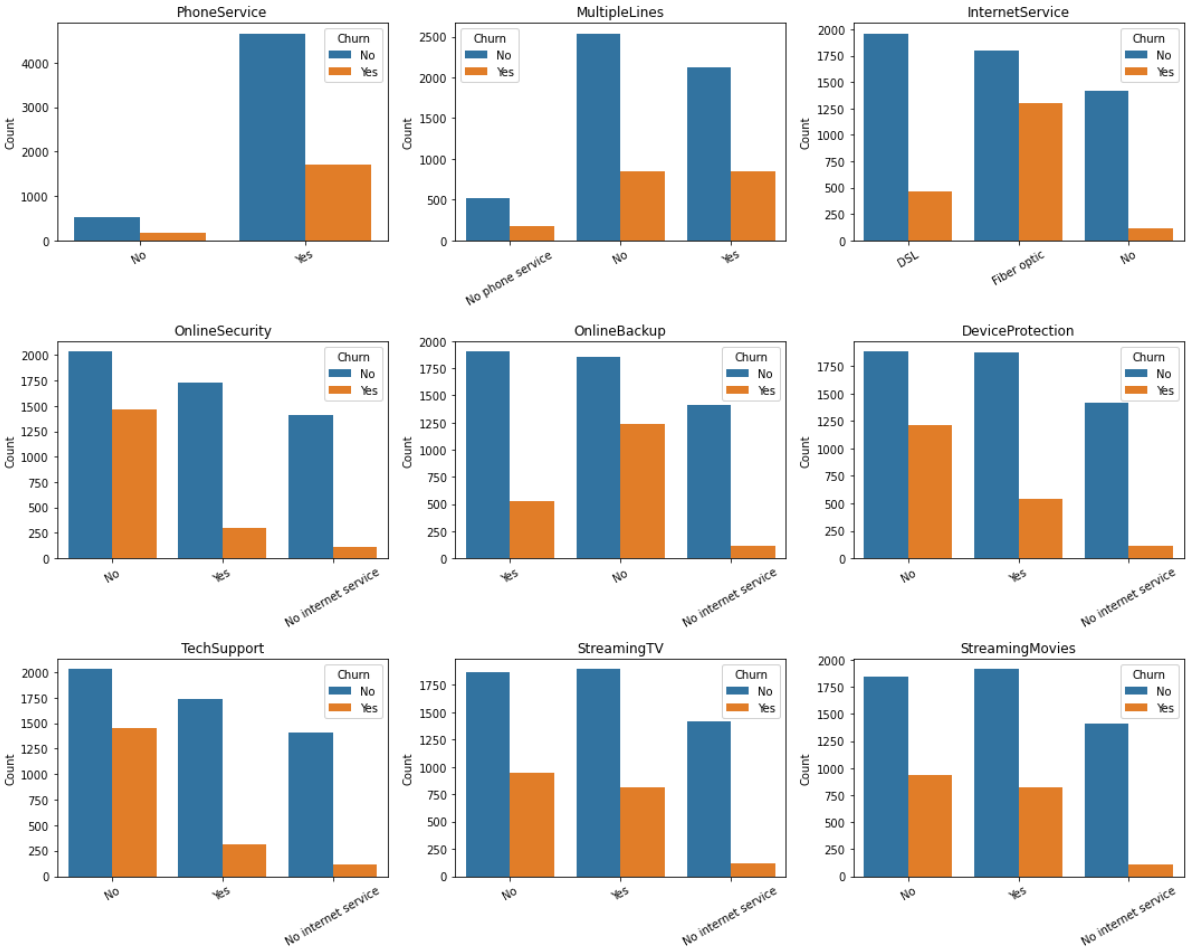
In [39]:
```
plt.figure(figsize=(5,5))
bx = sns.countplot(x = 'Contract',data = df,hue='Churn')
bx.bar_label(bx.containers[0])
bx.bar_label(bx.containers[1])
plt.title("Churn on Basis of Contract")
plt.show()
```



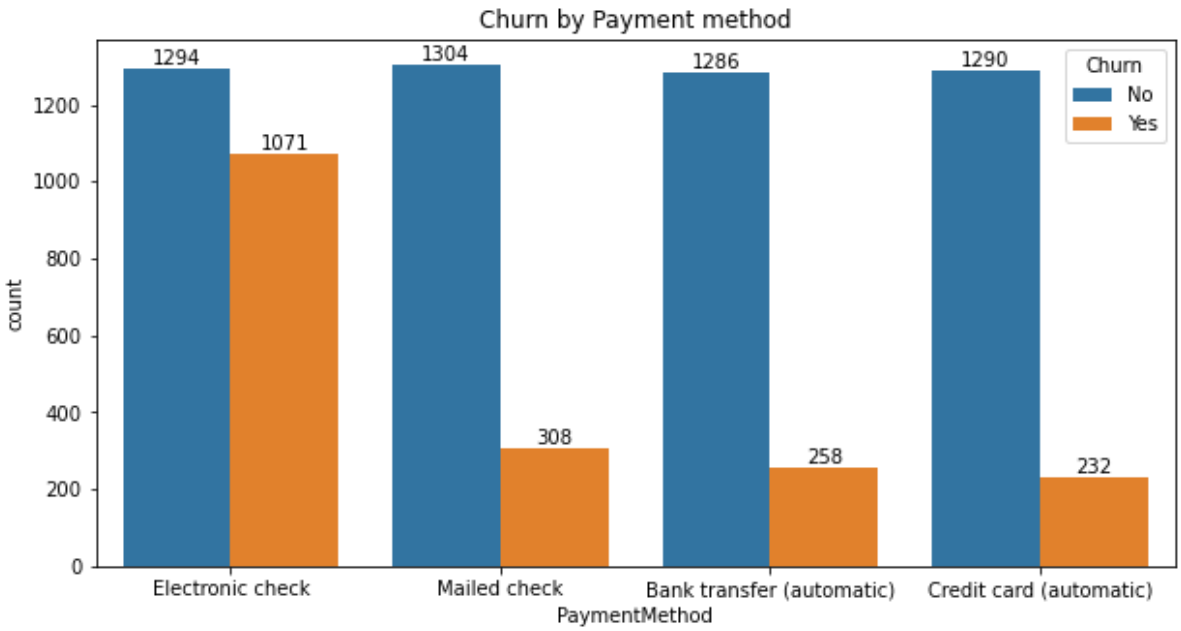
In [23]:
```
df.columns.values
```

Out[23]:
```
array(['customerID', 'gender', 'SeniorCitizen', 'Partner', 'Dependents',
       'tenure', 'PhoneService', 'MultipleLines', 'InternetService',
       'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
       'TechSupport', 'StreamingTV', 'StreamingMovies', 'Contract',
       'PaperlessBilling', 'PaymentMethod', 'MonthlyCharges',
       'TotalCharges', 'Churn'], dtype=object)
```

In [25]:
```
cols = [
    'PhoneService', 'MultipleLines', 'InternetService',
    'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
    'TechSupport', 'StreamingTV', 'StreamingMovies'
]

# Create subplots (3 rows x 3 columns)
fig, axes = plt.subplots(3, 3, figsize=(15, 12))
axes = axes.flatten()

# Plot countplots
for ax, col in zip(axes, cols):
    sns.countplot(
        x=col,
        data=df,
        ax=ax,
        hue="Churn"
    )
    ax.set_title(col)
    ax.set_xlabel("")
    ax.set_ylabel("Count")
    ax.tick_params(axis='x', rotation=30)

# Remove empty subplots (safety)
for i in range(len(cols), len(axes)):
    fig.delaxes(axes[i])

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



In [37]:
```
plt.figure(figsize=(10,5))
ax = sns.countplot(x="PaymentMethod",data=df,hue='Churn')
ax.bar_label(ax.containers[0])
ax.bar_label(ax.containers[1])
plt.title("Churn by Payment method")
plt.show()
```



In [36]:

Out[36]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x000001BF970D3490>
```

In [ ]: