

```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
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-JJAZL	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
to-month				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
```

```
#replace blanks with 0 as tenure is 0 and no total charges are recorded
```

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

```
np.int64(0)
```

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

```
np.int64(0)
```

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

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

```
df.head()
```

	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-CF0CW	Male	no	No	No	45
No						
4	9237-HQITU	Female	no	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				

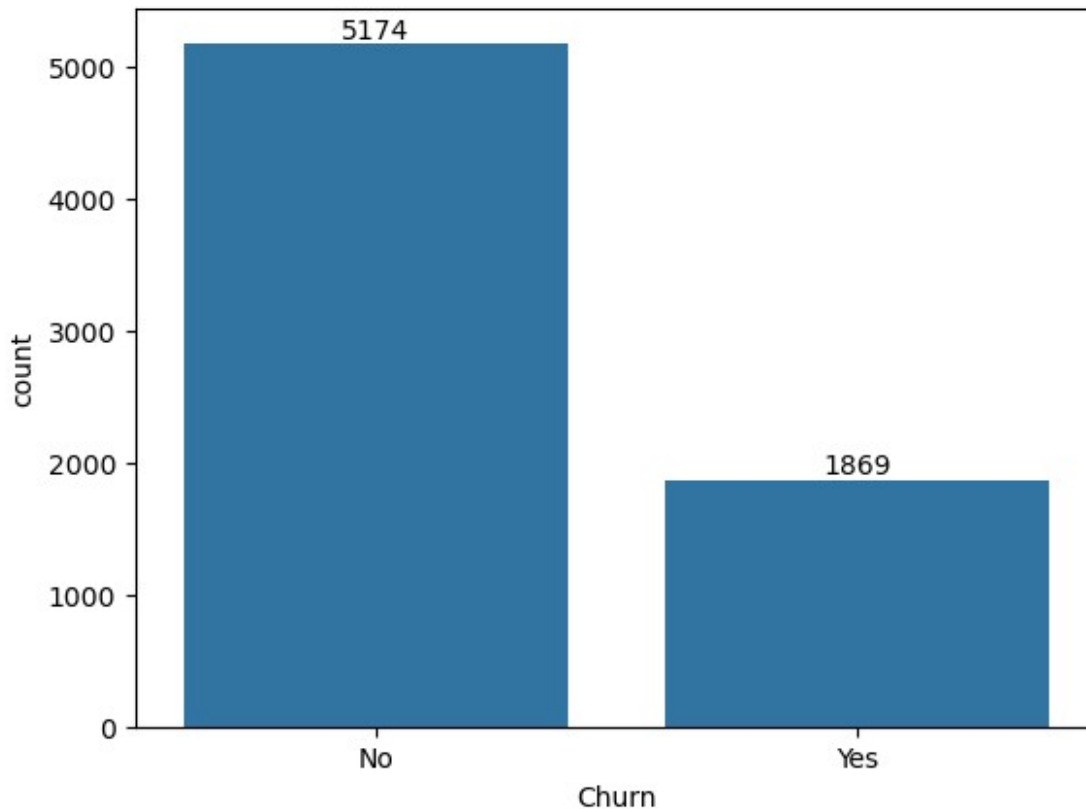
	TechSupport	StreamingTV	StreamingMovies	Contract
PaperlessBilling \				
0	No	No	No	Month-to-month
Yes				
1	No	No	No	One year
No				
2	No	No	No	Month-to-month
Yes				
3	Yes	No	No	One year
No				
4	No	No	No	Month-to-month
Yes				

	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

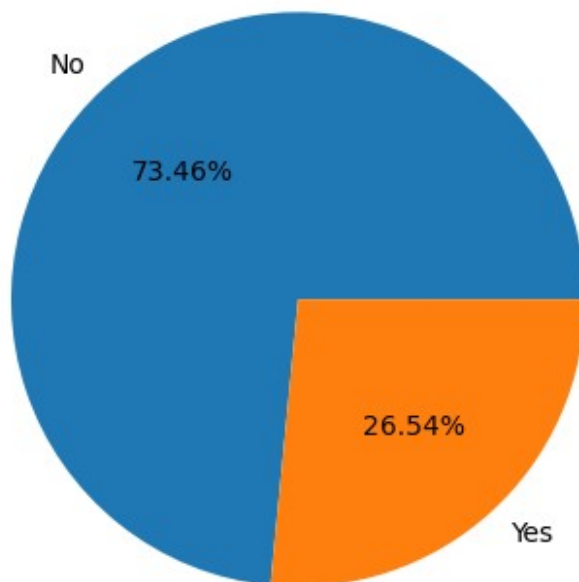
[5 rows x 21 columns]

#convert 0 and 1 values of sinercitizen to yes/no for better under standing

```
ax = sns.countplot(x = 'Churn' , data = df)
ax.bar_label(ax.containers[0])
plt.show()
```

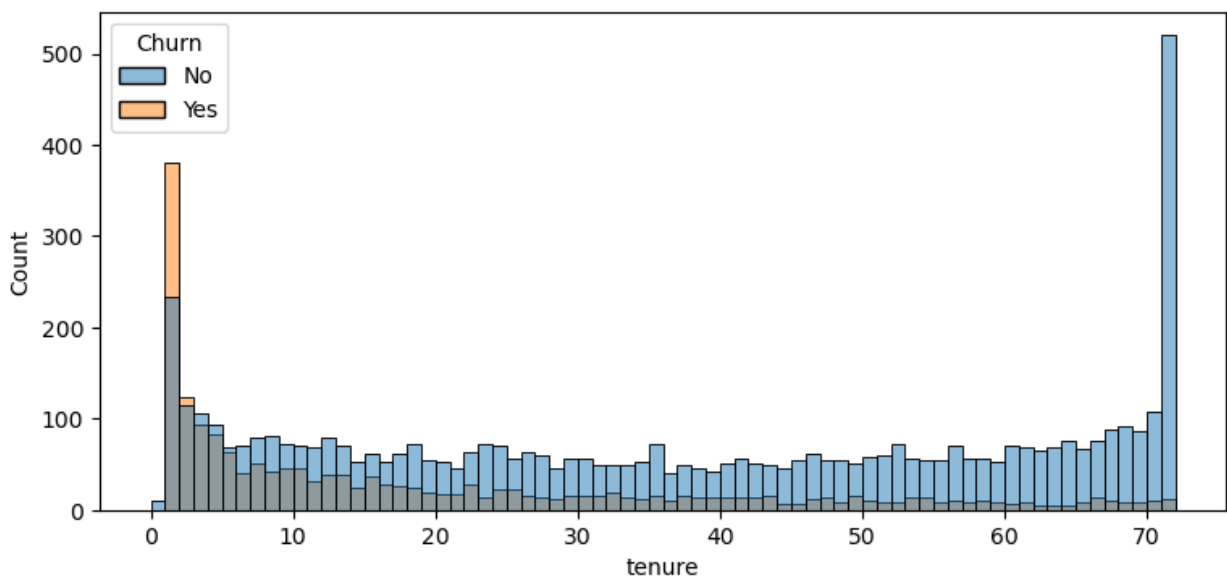


```
gb = df.groupby("Churn").agg({'Churn':"count"})
plt.pie(gb['Churn'] , labels = gb.index, autopct = "%1.2f%%")
plt.show()
```



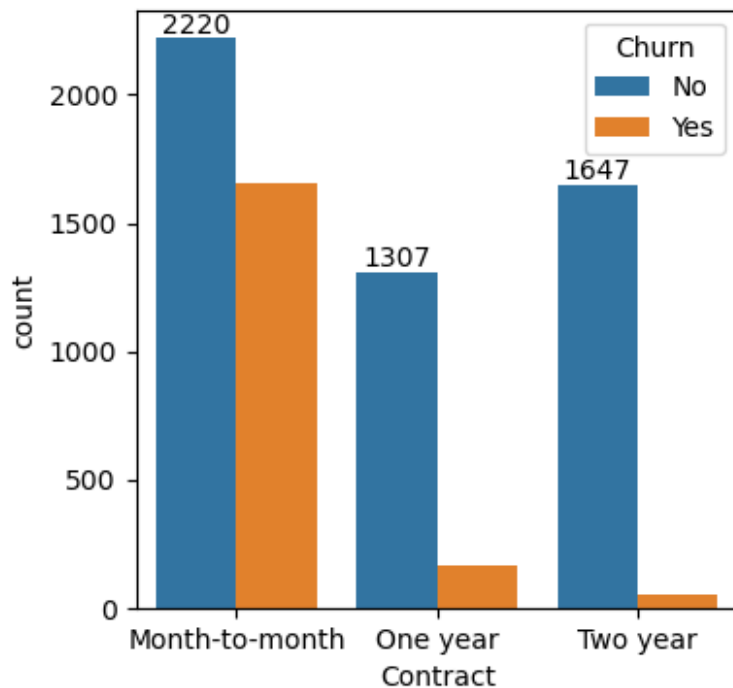
#in this pie chart we are understand and analysis how many customer left Churn

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



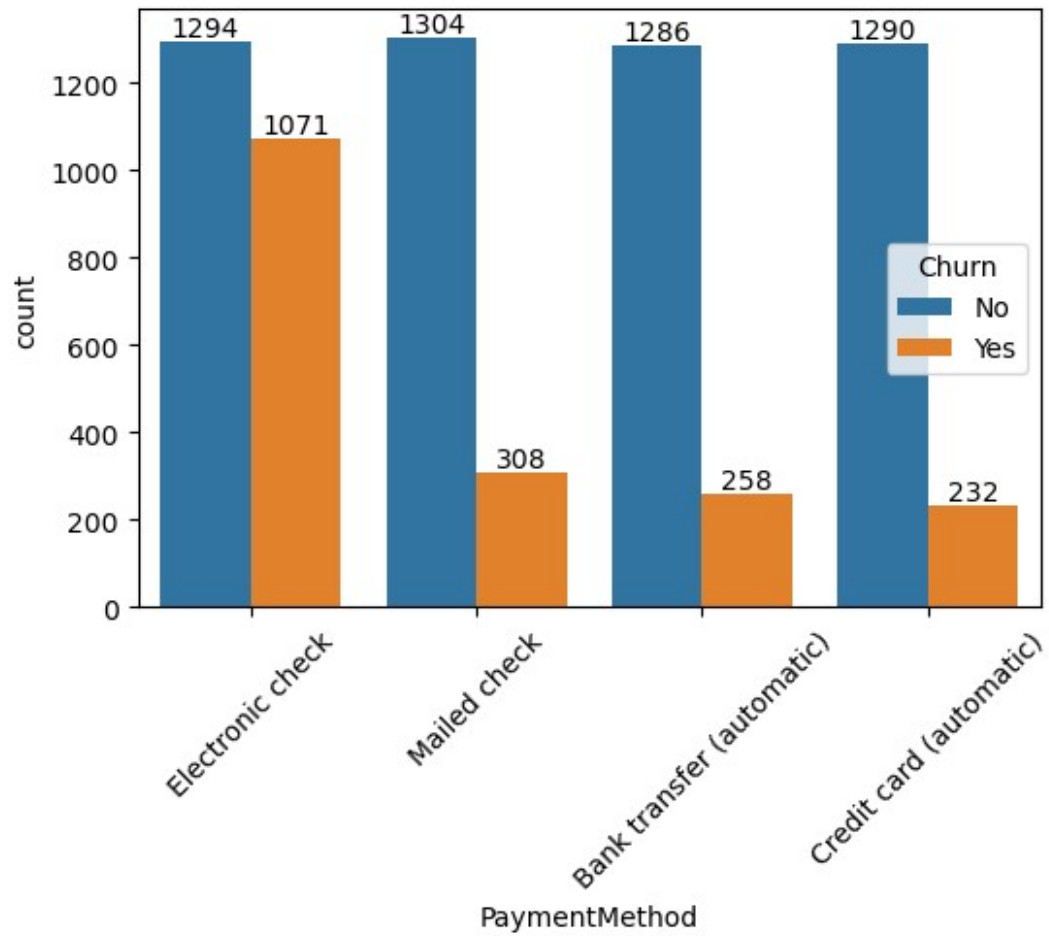
#people who have used our services for a long time have stayed and people who used our services 1 and 2 month have chured

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



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

```
plt.figure(figsize = (6,4))
ax = sns.countplot(x = 'PaymentMethod' , data = df , hue = 'Churn')
ax.bar_label(ax.containers[0])
ax.bar_label(ax.containers[1])
plt.xticks(rotation = 45 )
plt.show()
```

#customer as likely to Churn when he is using electronic check as a payment method

