

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv("/content/telco customer churn.csv")

df

{"type": "dataframe", "variable_name": "df"}

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.isnull().sum().sum()

np.int64(0)

df.describe()

{"summary": "{\n  \"name\": \"df\",\n  \"rows\": 8,\n  \"fields\": [\n    {\n      \"column\": \"SeniorCitizen\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 2489.9992387084,\n        \"min\": 0,\n        \"max\": 1,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"Partner\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"Dependents\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"tenure\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 1.4142135623730951,\n        \"min\": 0,\n        \"max\": 2,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"PhoneService\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"MultipleLines\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"InternetService\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"OnlineSecurity\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"OnlineBackup\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"DeviceProtection\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"TechSupport\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"StreamingTV\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"StreamingMovies\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"Contract\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"PaperlessBilling\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"PaymentMethod\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"MonthlyCharges\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 0.000125892541179,\n        \"min\": 0,\n        \"max\": 0,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"TotalCharges\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    },\n    {\n      \"column\": \"Churn\",\n      \"properties\": {\n        \"dtype\": \"object\",\n        \"std\": null,\n        \"min\": null,\n        \"max\": null,\n        \"num_null\": 0,\n        \"num_non_null\": 8\n      }\n    }\n  ]\n}"

```

```

{"min": 0.0, "max": 7043.0, "num_unique_values": 5, "samples": [0.1621468124378816, 1.0, 0.36861160561002687], "semantic_type": "", "description": ""}, {"column": "tenure", "properties": {"dtype": "number", "std": 2478.9752758409018, "min": 0.0, "max": 7043.0, "num_unique_values": 8, "samples": [32.37114865824223, 29.0, 7043.0]}, "semantic_type": "", "description": ""}, {"column": "MonthlyCharges", "properties": {"dtype": "number", "std": 2468.7047672837775, "min": 18.25, "max": 7043.0, "num_unique_values": 8, "samples": [64.76169246059918, 70.35, 7043.0]}, "semantic_type": "", "description": ""}]
}, {"type": "dataframe"}

```

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

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

```
df.columns
```

```

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

```

```
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["SeniorCitizen"][1:21]
```

```

1    no
2    no
3    no

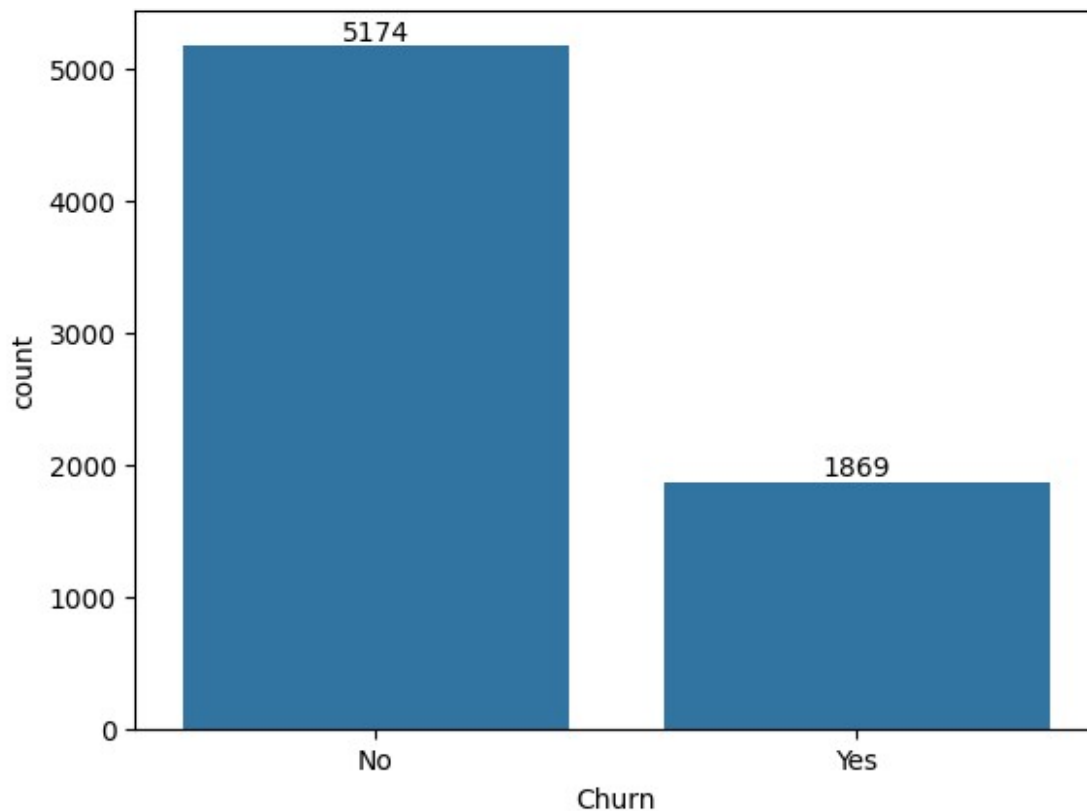
```

```
4      no
5      no
6      no
7      no
8      no
9      no
10     no
11     no
12     no
13     no
14     no
15     no
16     no
17     no
18     no
19     no
20     no
Name: SeniorCitizen, dtype: object
```

#know data is clean

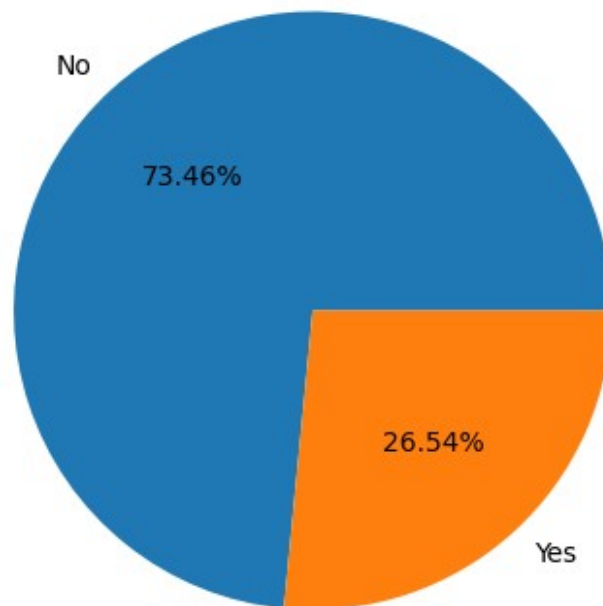
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

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

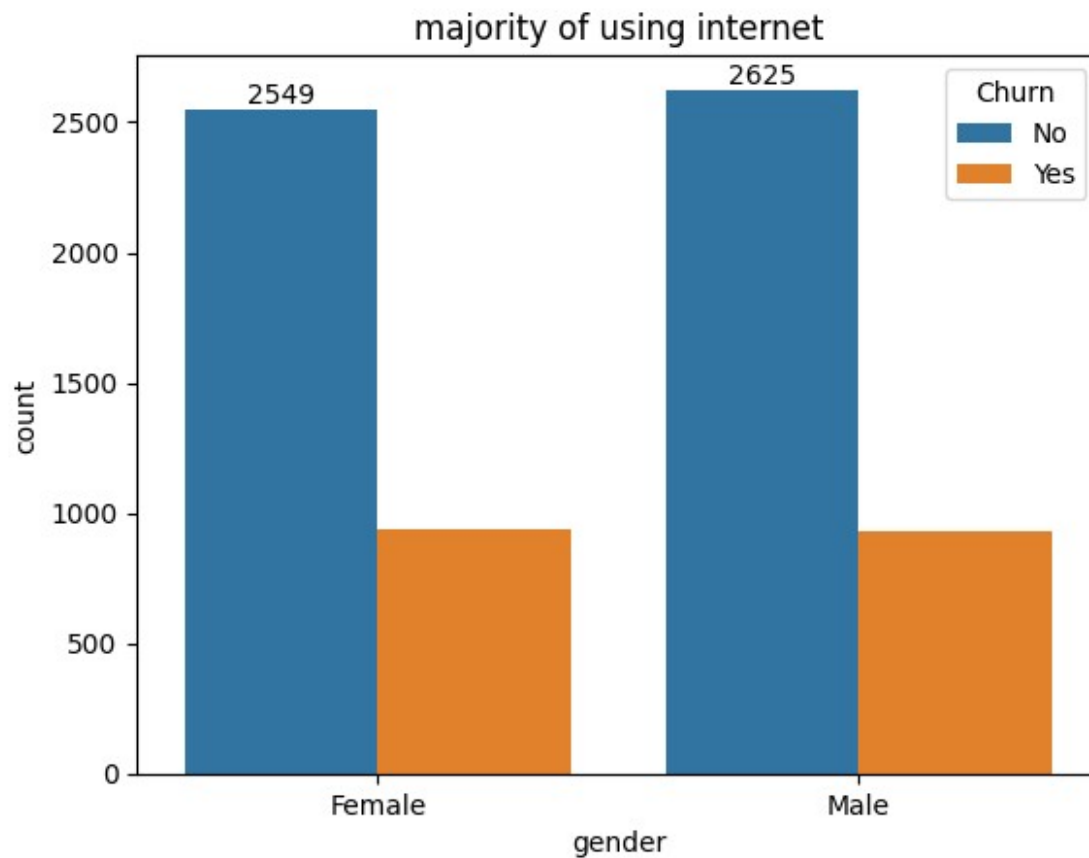


```
plt.figure(figsize = (10,5))
plt.title("percentage of churned customers",fontsize = 20)
gb = df.groupby("Churn").agg({"Churn" : "count"})
plt.pie(gb["Churn"],labels = gb.index,autopct = "%.2f%%")
plt.show()
```

## percentage of churned customers

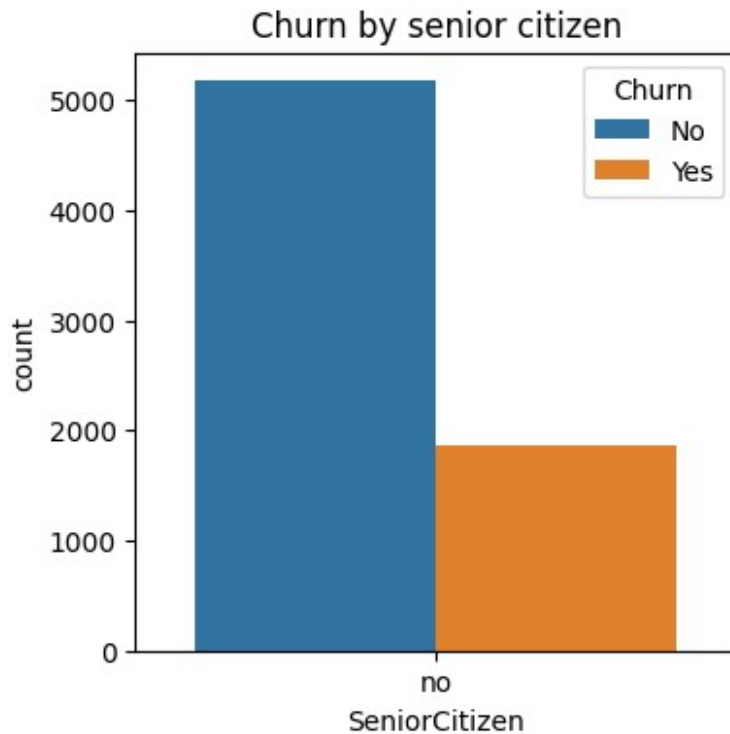


```
plt.title("majority of using internet")
y = sns.countplot(x = "gender", data = df, hue = "Churn")
y.bar_label(y.containers[0])
plt.show()
```



```
#senior citizen
```

```
plt.figure(figsize = (3,3))  
sns.countplot(x = "SeniorCitizen",data = df, hue = "Churn")  
plt.title("Churn by senior citizen")  
plt.show()
```



```
import pandas as pd
import matplotlib.pyplot as plt

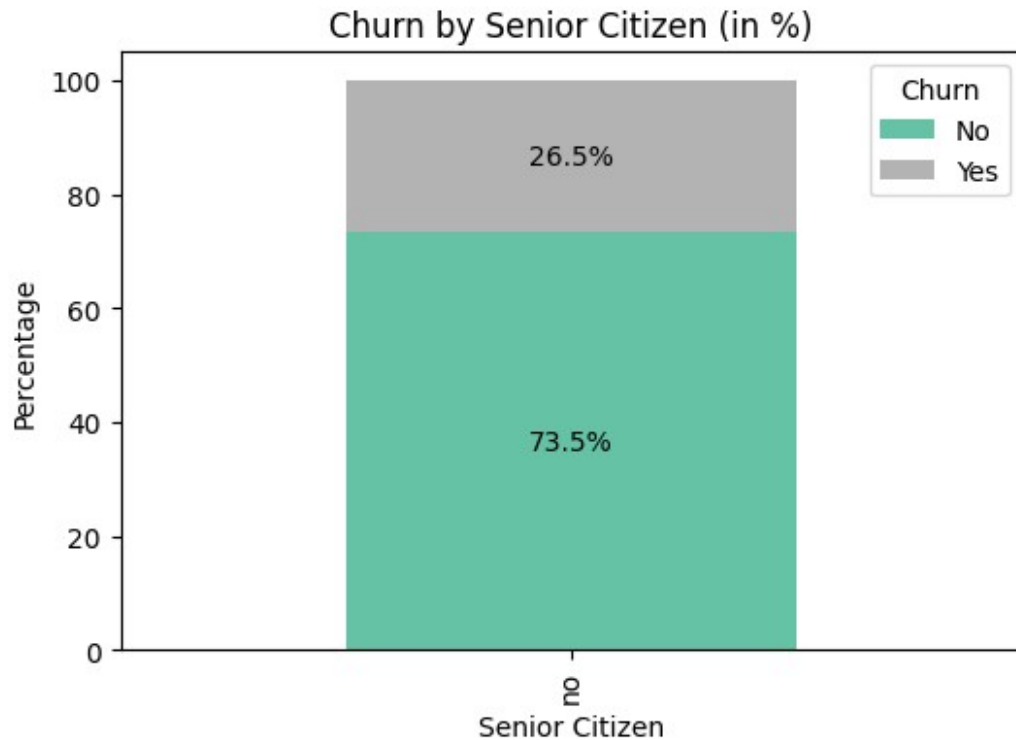
# Create a crosstab of counts
ct = pd.crosstab(df['SeniorCitizen'], df['Churn'])

# Convert counts to percentages (row-wise)
ct_percent = ct.div(ct.sum(axis=1), axis=0) * 100

# Plot stacked bar chart
ax = ct_percent.plot(kind='bar', stacked=True, figsize=(6,4),
colormap="Set2")

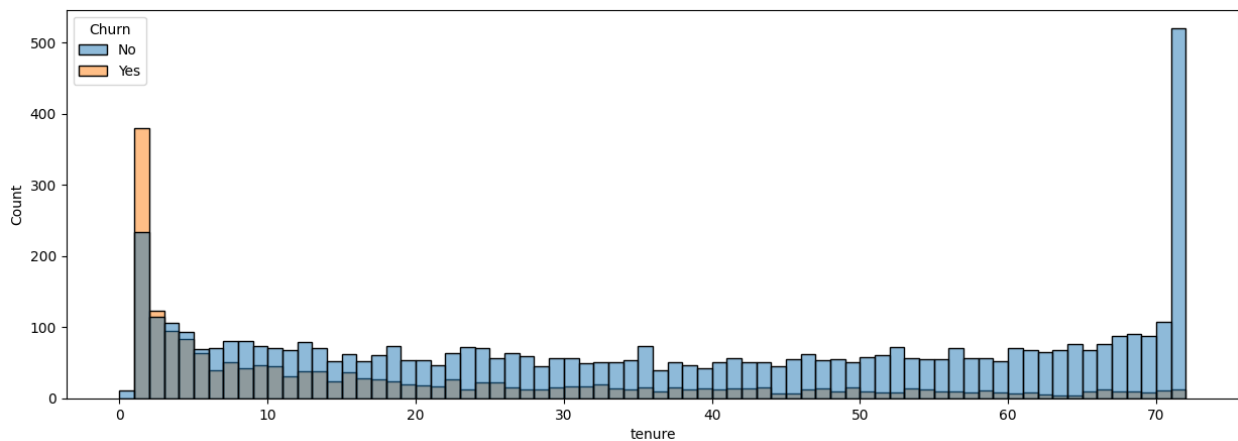
# Add percentage labels
for c in ax.containers:
    ax.bar_label(c, fmt="%.1f%%", label_type="center")

plt.title("Churn by Senior Citizen (in %)")
plt.ylabel("Percentage")
plt.xlabel("Senior Citizen")
plt.legend(title="Churn")
plt.show()
```



*#tenure chart*

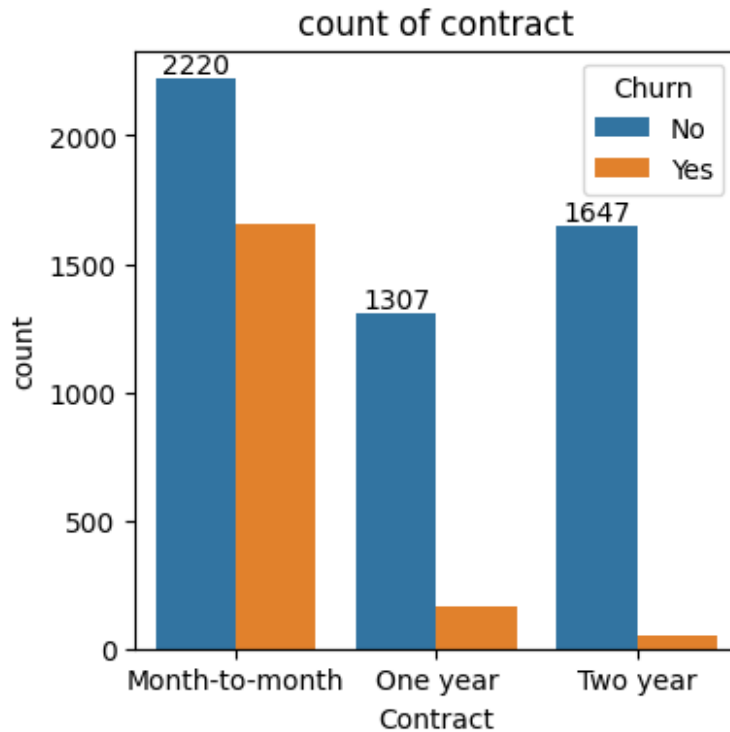
```
plt.figure(figsize = (15,5))
sns.histplot(x = "tenure",data = df, bins = 72, hue = "Churn")
plt.show()
```



*#contract ka chart*

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





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

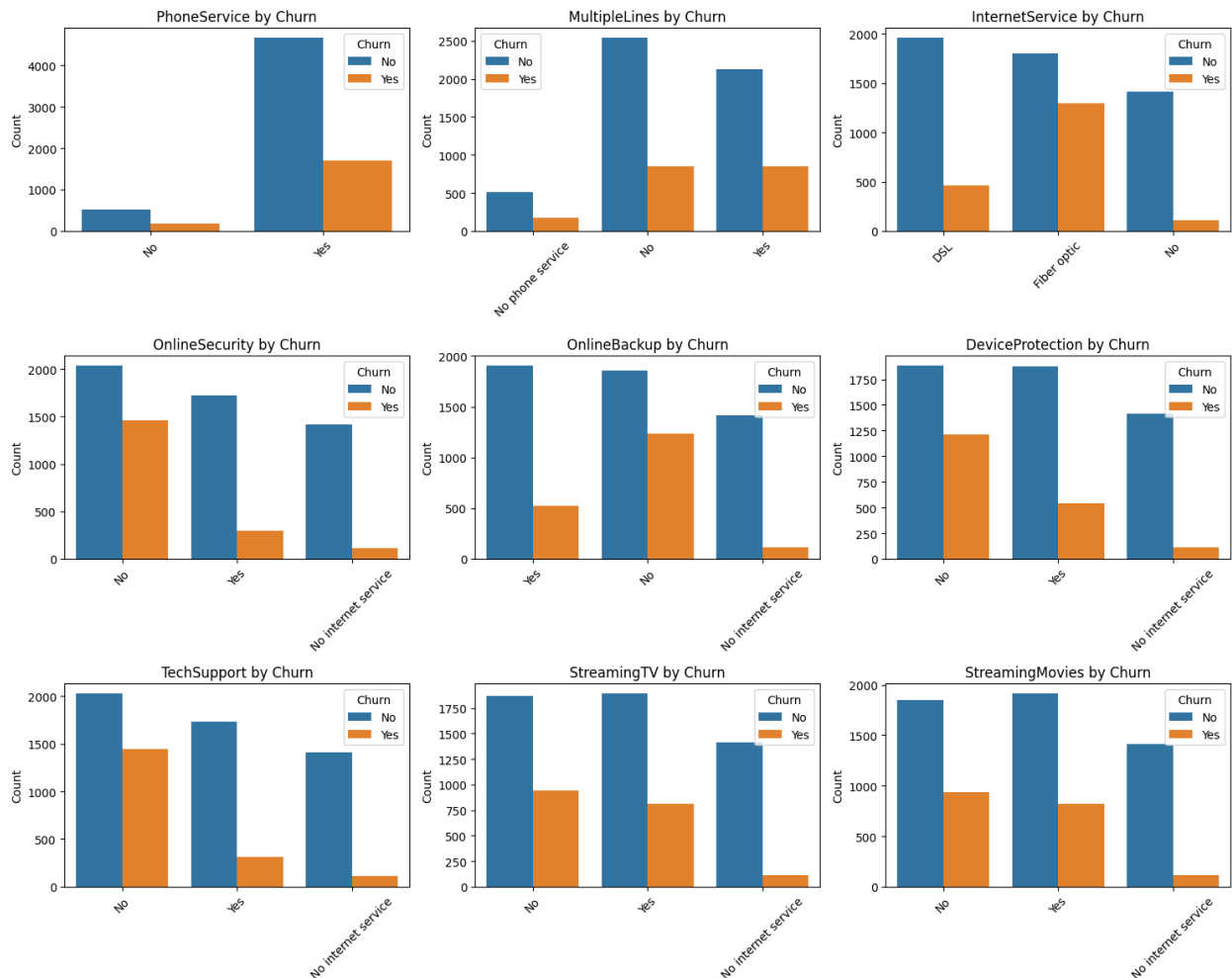
cols = [
    'PhoneService', 'MultipleLines', 'InternetService',
    'OnlineSecurity', 'OnlineBackup', 'DeviceProtection',
    'TechSupport', 'StreamingTV', 'StreamingMovies'
]

# Define subplot grid (3x3 because 9 features)
fig, axes = plt.subplots(3, 3, figsize=(15, 12))
axes = axes.flatten() # flatten to easily iterate

for i, col in enumerate(cols):
    sns.countplot(x=col, data=df, hue="Churn", ax=axes[i])
    axes[i].set_title(f"{col} by Churn")
    axes[i].set_xlabel("") # cleaner look
    axes[i].set_ylabel("Count")
```

```
axes[i].tick_params(axis='x', rotation=45)
```

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



#From these plots, we can see:

Customers with No PhoneService are very few, and most churn happens among those with service.

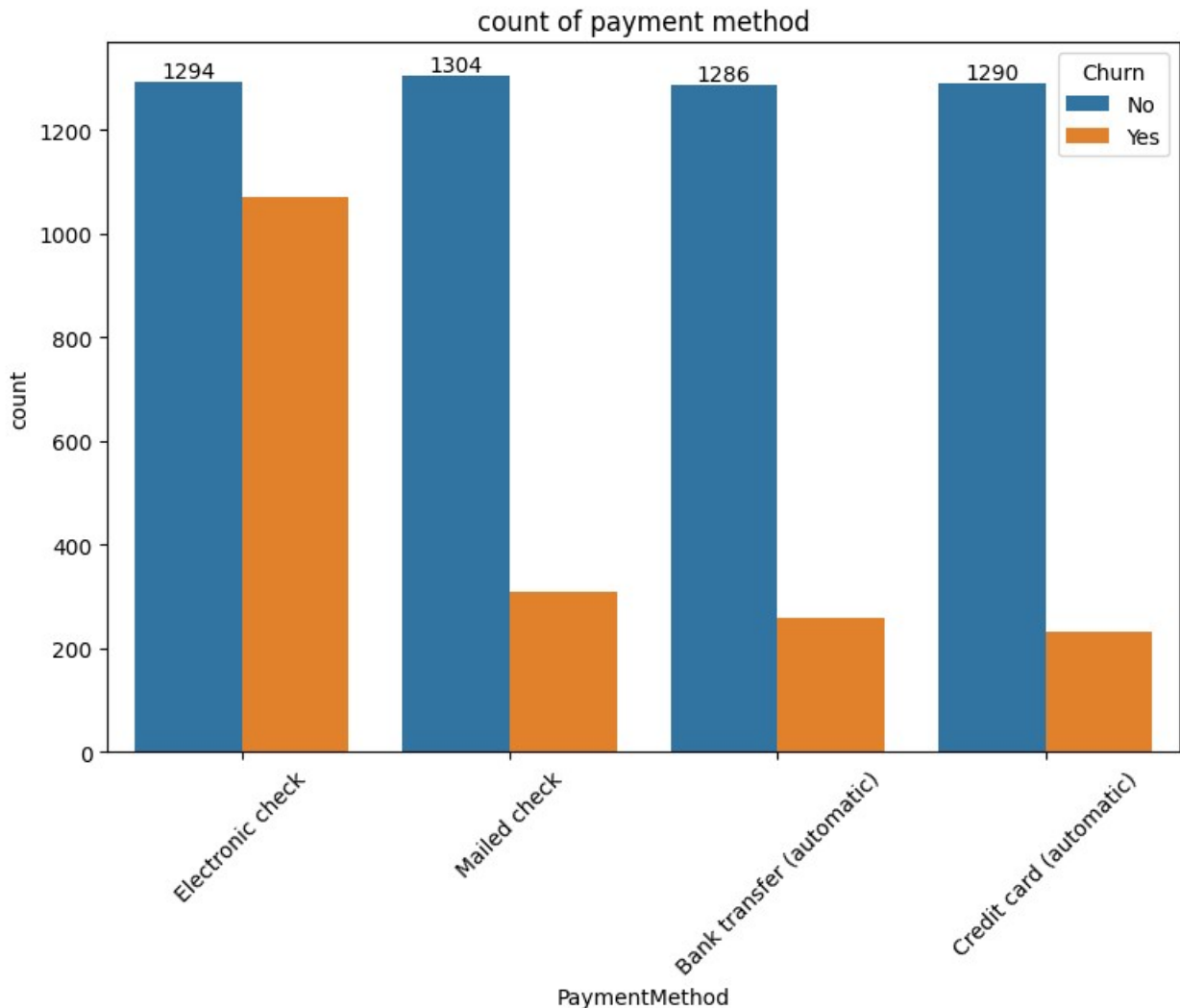
For internet-related features (security, backup, device protection, tech support), churn is much higher when the service is not subscribed.

Fiber optic internet users show notably higher churn compared to DSL or no internet service.

Streaming services (TV, Movies) don't show a strong churn difference, though users without these services still churn significantly

*#payment method chart*

```
plt.figure(figsize = (9,6))
ax = sns.countplot(x = "PaymentMethod" ,data = df, hue = "Churn")
ax.bar_label(ax.containers[0])
plt.title("count of payment method")
plt.xticks(rotation = 45)
plt.show()
```



### #Executive Summary

This analysis explores key factors associated with customer churn in the telco dataset. Initial data inspection revealed no missing values or duplicate entries.

### #Key findings from the visualizations include:

**Overall Churn Rate:** Approximately 26.54% of customers in the dataset have churned, indicating a significant churn problem.

**Gender and Churn:** The churn rate appears to be similar between male and female customers.

**Senior Citizens and Churn:** Senior citizens show a notably higher churn rate compared to non-senior citizens.

**Tenure and Churn:** Customers with shorter tenures (especially those in the first few months) have a higher propensity to churn. Churn also appears elevated among customers with very long tenures.

**Contract Type and Churn:** Customers on month-to-month contracts have a significantly higher churn rate compared to those with one-year or two-year contracts. Internet Service and Churn: Customers with Fiber optic internet service have a higher churn rate than those with DSL or no internet service.

**Other Service Features:** For most internet-related services (Online Security, Online Backup, Device Protection, Tech Support), customers who do not subscribe to these services are more likely to churn. Streaming services (TV and Movies) show less of a clear difference in churn rates.

**Payment Method and Churn:** Electronic check users have the highest churn rate among all payment methods.

#*In summary*, customers who are senior citizens, have shorter tenures, are on month-to-month contracts, use fiber optic internet, do not subscribe to security/backup/support services, and use electronic checks are more likely to churn. These insights can be valuable for developing targeted customer retention strategies.

```
from google.colab import files
files.download("telco_customer_churn.csv")

<IPython.core.display.Javascript object>

<IPython.core.display.Javascript object>
```