

```

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.head()
#replacing the blanks into "0" values
df["TotalCharges"] = df["TotalCharges"].replace(" ", "0")
df["TotalCharges"] = df["TotalCharges"].astype("float") #overwriting the data type from object(string)=float

#re-verifying
df.info()
df.isnull().sum() #if null values exist or not
df.describe()
df.duplicated().sum()
df["customerID"].duplicated().sum() #should also check with unique ID
#converted senior citizen into yes and no
def conv(value):
    if value == 1:
        return "yes"
    else:
        return "no"
df['SeniorCitizen'] = df['SeniorCitizen'].apply(conv)
df

gb = df.groupby("Churn").agg({'Churn': 'count'}) #grouping all the values in churn into yes/no
gb
gb = df.groupby("Churn").agg({'Churn': 'count'}) #grouping all the values in churn into yes/no
gb
plt.figure(figsize=(3, 4)) #pie chart size
plt.title("percentage of churn")
plt.pie(gb['Churn'], labels=gb.index, autopct="%1.0f%%")
plt.show()
from the given pie chart we can conclude that 27% of customers are churned or they ported to a different company
lets explore the reason behind it
sns.countplot(x="gender", data=df, hue="Churn")
plt.title("gender")
plt.show()
hence this churning is not gender specialized
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
df = pd.read_csv('Customer Churn.csv')
ax = sns.countplot(x="SeniorCitizen", data=df)
ax.bar_label(ax.containers[0])
plt.title("SeniorCitizen")
plt.figure(figsize=(3, 3))
plt.show()
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Load your data
df = pd.read_csv('Customer Churn.csv')

# Convert 0 and 1 to "No" and "Yes"
def conv(value):
    if value == 1:
        return "Yes"
    else:
        return "No"
df['SeniorCitizen'] = df['SeniorCitizen'].apply(conv)

# Calculate the normalized counts (percentage)
total_counts = df.groupby('SeniorCitizen')['Churn'].value_counts(normalize=True).unstack() * 100

# Plot
fig, ax = plt.subplots(figsize=(4, 4)) # Adjust figsize for better visualization

# Plot the bars
total_counts.plot(kind='bar', stacked=True, ax=ax, color=['#1f77b4', '#ff7f0e']) # Customize colors if desired

# Add percentage labels on the bars
for p in ax.patches:
    width, height = p.get_width(), p.get_height()
    x, y = p.get_xy()
    ax.text(x + width / 2, y + height / 2, f'{height:.1f}%', ha='center', va='center')

# Customize x-axis labels
ax.set_xticklabels(['No', 'Yes']) # Replace 0 and 1 with 'No' and 'Yes'

plt.title('Churn by Senior Citizen (Stacked Bar Chart)')
plt.xlabel('Senior Citizen')
plt.ylabel('Percentage (%)')
plt.xticks(rotation=0)
plt.legend(title='Churn', bbox_to_anchor=(0.9, 0.9)) # Customize legend location

plt.show()

this concludes that more amount of senior citizen converts churns

sns.histplot(x="tenure", data=df, bins=72, hue="Churn")
plt.show()
we realise that majority of people who have churned out is in 1st month
ax = sns.countplot(x="Contract", data=df, hue="Churn")
ax.bar_label(ax.containers[0])
plt.title("Contract")
plt.figure(figsize=(3, 3))
plt.show()
columns = ['PhoneService', 'MultipleLines', 'InternetService', 'OnlineSecurity',
           'OnlineBackup', 'DeviceProtection', 'TechSupport', 'StreamingTV', 'StreamingMovies']

# Number of columns for the subplot grid (you can change this)
n_cols = 3

```

```

n_rows = (len(columns) + n_cols - 1) // n_cols # Calculate number of rows needed

# Create subplots
fig, axes = plt.subplots(n_rows, n_cols, figsize=(15, n_rows * 4)) # Adjust figsize as needed

# Flatten the axes array for easy iteration (handles both 1D and 2D arrays)
axes = axes.flatten()

# Iterate over columns and plot count plots
for i, col in enumerate(columns):
    sns.countplot(x=col, data=df, ax=axes[i], hue = df["Churn"])
    axes[i].set_title(f'Count Plot of {col}')
    axes[i].set_xlabel(col)
    axes[i].set_ylabel('Count')

# Remove empty subplots (if any)
for j in range(i + 1, len(axes)):
    fig.delaxes(axes[j])

ply.tight_layout()
ply.show()

```

The majority of customers who do **not** churn tend to have services like PhoneService, InternetService (particularly DSL), **and** OnlineSecurity enabled. For service

```

ax=sns.countplot(x="PaymentMethod",data=df,hue="Churn")
ax.bar_label(ax.containers[0])
ply.title("Count of customer by Payment Method")
ply.xticks(rotation=45)
ply.show()

```

Customers using automatic payment methods (bank transfer **or** credit card) appear more loyal, **while** those using electronic checks are more likely to churn. This