How to Place Matplotlib Charts on a Tkinter GUI

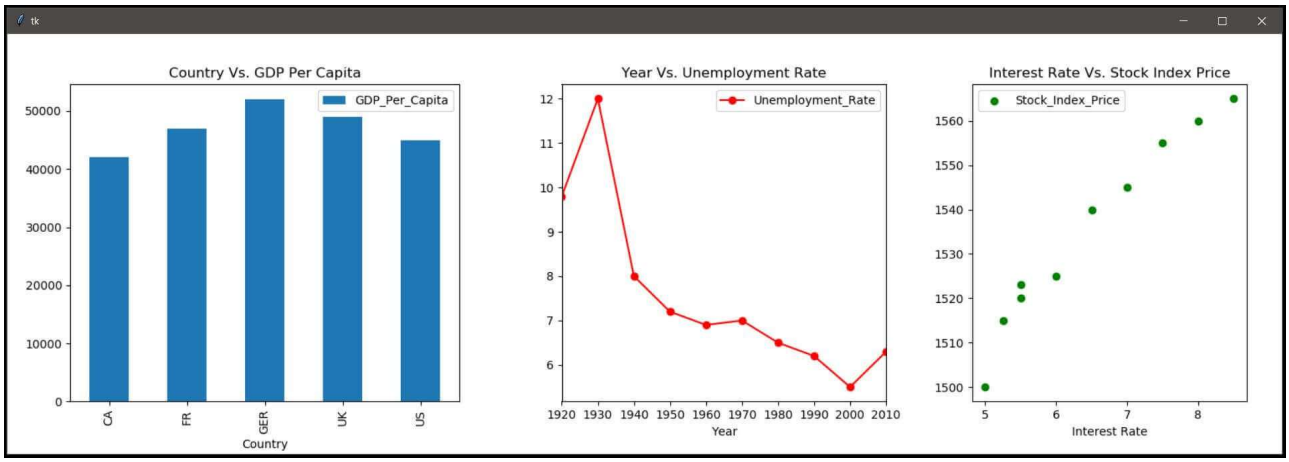
Often times, you may need to place *matplotlib* charts on a tkinter GUI. This feature is especially useful for users who deal with front-end GUIs.

And so, in this tutorial, I'll show you the steps to place matplotlib charts on a tkinter GUI.

More specifically, I'll show you how to embed the following charts on your GUI:

- Bar
- Line
- Scatter

By the end of this tutorial, you'll be able to create the following tkinter GUI with the embedded charts:



Let's now review the steps to achieve this goal.

Steps to place *matplotlib* charts on a tkinter GUI

Step 1: Prepare the datasets for the charts

Firstly, you'll need to prepare the datasets to be used as the input for the charts.

For illustration purposes, I created the following 3 datasets for our charts:

Data for the Bar Chart

Country	GDP_Per_Capita
US	45000
CA	42000
GER	52000
UK	49000
FR	47000

Data for the Line Chart

Year	Unemployment_Rate
1920	9.8
1930	12
1940	8
1950	7.2
1960	6.9
1970	7
1980	6.5

1990	6.2
2000	5.5
2010	6.3

Data for the Scatter Diagram

Interest_Rate	Stock_Index_Price
5	1500
5.5	1520
6	1525
5.5	1523
5.25	1515
6.5	1540
7	1545
8	1560
7.5	1555
8.5	1565

Step 2: Create the DataFrames in Python

Next, you may utilize pandas DataFrame to capture the above data in Python.

Here is a template that you may use to create a DataFrame in Python:

```
from pandas import DataFrame

data = {'First Column Name': ['First value', 'Second
value',…],

        'Second Column Name': ['First value', 'Second
value',…],

         …

        }

df = DataFrame(data, columns = ['First Column
Name','Second Column Name',…])
```

For our example, the datasets can be captured as follows:

```
from pandas import DataFrame

data1 = {'Country': ['US','CA','GER','UK','FR'],

         'GDP_Per_Capita':
[45000,42000,52000,49000,47000]

        }

df1 =
DataFrame(data1,columns=['Country','GDP_Per_Capita'])

print (df1)

data2 = {'Year':
[1920,1930,1940,1950,1960,1970,1980,1990,2000,2010],
```

```
          'Unemployment_Rate':
[9.8,12,8,7.2,6.9,7,6.5,6.2,5.5,6.3]

        }

df2 =
DataFrame(data2,columns=['Year','Unemployment_Rate'])

print (df2)

data3 = {'Interest_Rate':
[5,5.5,6,5.5,5.25,6.5,7,8,7.5,8.5],

          'Stock_Index_Price':
[1500,1520,1525,1523,1515,1540,1545,1560,1555,1565]

        }

df3 =
DataFrame(data3,columns=['Interest_Rate','Stock_Index_Pri
ce'])

print (df3)
```

If you run the above code, you'll get these 3 DataFrames:

	Country	GDP_Per_Capita
0	US	45000
1	CA	42000
2	GER	52000
3	UK	49000
4	FR	47000

	Year	Unemployment_Rate
0	1920	9.8
1	1930	12.0
2	1940	8.0
3	1950	7.2
4	1960	6.9
5	1970	7.0
6	1980	6.5
7	1990	6.2
8	2000	5.5
9	2010	6.3

	Interest_Rate	Stock_Index_Price
0	5.00	1500
1	5.50	1520
2	6.00	1525
3	5.50	1523
4	5.25	1515
5	6.50	1540
6	7.00	1545
7	8.00	1560
8	7.50	1555
9	8.50	1565

Step 3: Create the GUI

Next, you'll need to create the tkinter GUI, so that you can place the charts on it.

To begin, you'll need to import the tkinter and matplotlib modules as follows:

```
import tkinter as tk

import matplotlib.pyplot as plt

from matplotlib.backends.backend_tkagg import
FigureCanvasTkAgg
```

Then, add the charts to the GUI by using this generic template:

```
figure = plt.Figure(figsize=(6,5), dpi=100)
```

```
ax = figure.add_subplot(111)

chart_type = FigureCanvasTkAgg(figure, root)

chart_type.get_tk_widget().pack()

df = df[['First Column','Second Column']].groupby('First
Column').sum()

df.plot(kind='Chart Type such as bar', legend=True,
ax=ax)

ax.set_title('The Title for your chart')
```

Slight variations may be applied to the above template, depending on the chart that you need to plot.

Putting everything together, your full Python code would look like this:

```
import tkinter as tk

from pandas import DataFrame

import matplotlib.pyplot as plt

from matplotlib.backends.backend_tkagg import
FigureCanvasTkAgg



data1 = {'Country': ['US','CA','GER','UK','FR'],

         'GDP_Per_Capita':
[45000,42000,52000,49000,47000]

        }

df1 =
DataFrame(data1,columns=['Country','GDP_Per_Capita'])
```

```
data2 = {'Year':
[1920,1930,1940,1950,1960,1970,1980,1990,2000,2010],

         'Unemployment_Rate':
[9.8,12,8,7.2,6.9,7,6.5,6.2,5.5,6.3]

        }

df2 =
DataFrame(data2,columns=['Year','Unemployment_Rate'])




data3 = {'Interest_Rate':
[5,5.5,6,5.5,5.25,6.5,7,8,7.5,8.5],

         'Stock_Index_Price':
[1500,1520,1525,1523,1515,1540,1545,1560,1555,1565]

        }

df3 =
DataFrame(data3,columns=['Interest_Rate','Stock_Index_Pri
ce'])

root= tk.Tk()



figure1 = plt.Figure(figsize=(6,5), dpi=100)

ax1 = figure1.add_subplot(111)

bar1 = FigureCanvasTkAgg(figure1, root)

bar1.get_tk_widget().pack(side=tk.LEFT, fill=tk.BOTH)
```

```
df1 =
df1[['Country','GDP_Per_Capita']].groupby('Country').sum(
)

df1.plot(kind='bar', legend=True, ax=ax1)

ax1.set_title('Country Vs. GDP Per Capita')



figure2 = plt.Figure(figsize=(5,4), dpi=100)

ax2 = figure2.add_subplot(111)

line2 = FigureCanvasTkAgg(figure2, root)

line2.get_tk_widget().pack(side=tk.LEFT, fill=tk.BOTH)

df2 =
df2[['Year','Unemployment_Rate']].groupby('Year').sum()

df2.plot(kind='line', legend=True, ax=ax2,
color='r',marker='o', fontsize=10)

ax2.set_title('Year Vs. Unemployment Rate')



figure3 = plt.Figure(figsize=(5,4), dpi=100)

ax3 = figure3.add_subplot(111)

ax3.scatter(df3['Interest_Rate'],df3['Stock_Index_Price']
, color = 'g')

scatter3 = FigureCanvasTkAgg(figure3, root)

scatter3.get_tk_widget().pack(side=tk.LEFT, fill=tk.BOTH)

ax3.legend(['Stock_Index_Price'])
```

```
ax3.set_xlabel('Interest Rate')

ax3.set_title('Interest Rate Vs. Stock Index Price')



root.mainloop()
```

Step 4: Run the Python code

Run the above Python code, and you'll see the matplotlib charts placed on the GUI: