In [6]:

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix
import pandas as pd

df = pd.read_csv('data/interview.csv')
y_train = df['admitted']
x_train = df[['english','gpa','experience']]
n_data = y_train.shape[0]

model = LogisticRegression(random_state =0)
model.fit(x_train,y_train)

y_pred = model.predict(x_train)
cm = confusion_matrix(y_train,y_pred)
print("\nnConfusion matrix\n",cm)

from sklearn import metrics
print("Accuracy: %.2f"%(metrics.accuracy_score(y_train,y_pred)))

x_new = np.array([[400,2.2,1],[255,3.5,5],[390,2.8,6],[100,4,10]])
y_pred_new = model.predict(x_new)
print("\nPredicted response of X :")
print(y_pred_new)
```

```
nConfusion matrix
 [[27  2]
 [ 1 28]]
Accuracy: 0.95

Predicted response of X :
['Rejected' 'Rejected' 'Accepted' 'Rejected']
```

In [68]:

```
from tkinter import *

root = Tk()
root.geometry("560x430")
l=Label(text='Show image',font='none 20',width=35,anchor=CENTER).place(x=10,y=5)
catBT = Button(text ='Cat',width = 15,anchor=CENTER)
dogtBT = Button(text ='Dog',width = 15,anchor=CENTER)
pigBT = Button(text ='Pig',width = 15,anchor=CENTER)
clearBT = Button(text ='Clear',width = 15,anchor=CENTER)
outputLabel = Label(width=76,height=20,borderwidth=2,relief='groove')

catBT.place(x=10,y=50)
dogtBT.place(x=150,y=50)
pigBT.place(x=290,y=50)
clearBT.place(x=430,y=50)
outputLabel.place(x=10,y=90)

root.mainloop()
```

In [1]:

```
from tkinter import *

root = Tk()


def IncreaseNumber():
    text = outputLabel.cget("text")
    number = (int)(text) +1
    outputLabel.config(text=str(number))
def DecreaseNumber():
    text = outputLabel.cget("text")
    number = (int)(text) -1
    outputLabel.config(text=str(number))
root.geometry("280x100")

plusB = Button(text = 'Plus',width =15,anchor=CENTER,command =IncreaseNumber)
minusB = Button(text = 'Minus',width =15,anchor=CENTER,command =DecreaseNumber)
outputLabel = Label(text='1',font='none 20',width=16 ,borderwidth=2,relief='groove' ,anchor=CENTER)

plusB.place(x=10,y=5)
minusB.place(x=150,y=5)
outputLabel.place(x=10,y=50)

root.title('Lab 2: Button trigger')
root:mainloop()
```

In [3]:

```
from tkinter import *
from PIL import ImageTk,Image
import random

def showImage():
    img = Image.open("data/Image/dog.jpg")
    img_resize = img.resize((400,320), Image.ANTIALIAS)
    file_img = ImageTk.PhotoImage(img_resize)
    canvas.create_image(0,0,anchor=NW,image=file_img)
    canvas.image =file_img
def clearImage():
    canvas.delete('all')

root = Tk()
root.geometry("425x375")
showButton = Button(text = 'Show',width =10,anchor=CENTER,command=showImage)
clearButton = Button(text = 'Clear',width =10,anchor=CENTER,command=clearImage)
canvas = Canvas(width=400,height=320,borderwidth=2,relief='groove')

showButton.place(x=120,y=8)
clearButton.place(x=220,y=8)
canvas.place(x=8,y=40)
root.mainloop()
```

In [69]:

```
import numpy as np
from tkinter import*
import matplotlib.pyplot as plt
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg

root = Tk()
root.title("Demo matplotlib Graphs")

x = np.random.randint(0,10,10)
y = np.random.randint(0,50,10)

fig = plt.Figure(figsize=(8,8), dpi=100)

plot1 = fig.add_subplot(121)
plot1.plot(x,y,'ro')

plot2 = fig.add_subplot(122)
plot2.plot(x,y,'g-')

canvas = FigureCanvasTkAgg(fig, master = root)
canvas.get_tk_widget().pack()

root.geometry('800x400')
root.mainloop()
```

In [77]:

```
from tkinter import *
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg

df1 = pd.DataFrame({
    'Major' :['CS','IT','CAG','CA','GIS'],
    'Students':[250,180,200,220,150]
})

df2 = pd.DataFrame({
    'Year':[2011,2012,2013,2014,2015,2016,2017,2018,2019,2020],
    'F_students':[20,15,18,10,5,6,13,5,19,25],
    'A_students':[10,20,30,40,50,40,30,20,10,10]
})

root =Tk()
root.title("Demo Graph from DatFrame")

fig =plt.Figure(figsize=(8,8),dpi=100)

plot1 = fig.add_subplot(221)
data = df1[['Major','Students']].groupby('Major').sum()
data.plot(kind="bar",color='green',legend=True,ax=plot1)
plot1.set_title('Students in Major')

plot2 =fig.add_subplot(222)
data = df2[['Year','F_students']].groupby('Year').sum()
data.plot(kind="line",color='red',legend=True,ax=plot2)
plot2.set_title('F students')

plot3 =fig.add_subplot(223)
data = df2[['Year','A_students']].groupby('Year').sum()
plot3.scatter(df2['Year'],df2['A_students'],color='green')
plot3.set_title('A students')

plot4 =fig.add_subplot(224)
data_a = df2[['Year','A_students']].groupby('Year').sum()
data_f = df2[['Year','F_students']].groupby('Year').sum()
data_a.plot(kind="line",color='red',legend=True,ax=plot2)
data_f.plot(kind="line",color='green',legend=True,ax=plot2)
plot3.set_title('A students')

canvas = FigureCanvasTkAgg(fig, master = root)
canvas.get_tk_widget().pack()

root.geometry('800x800')
root.mainloop()
```

In [76]:

```

import numpy as np
import pandas as pd
from tkinter import*
import matplotlib.pyplot as plt
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg

root = Tk()
root.title("Demo Graph matplotlib from Dataframe")

df1 = pd.DataFrame({
    'Major':['CS','IT','CAG','CA','GIS'],
    'Students':[250,180,200,220,150]
})
df2 = pd.DataFrame({
    'Year':[2011,2012,2013,2014,2015,2019,2017,2018,2019,2020],
    'F_Students':[20,15,18,10,5,6,13,5,19,25],
    'A_Students':[10,20,30,40,50,40,30,20,10,10]
})
```

```
fig = plt.Figure(figsize=(8,8), dpi=100)

plot1 = fig.add_subplot(221)
data = df1[['Major','Students']].groupby('Major').sum()
data.plot(kind='bar',color='green',legend=True, ax=plot1)
plot1.set_title('Students in Major.')

plot2= fig.add_subplot(222)
data = df2[['Year','F_Students']].groupby('Year').sum()
data.plot(kind='line',color='red',legend=True, ax=plot2)
plot2.set_title('F-students.')

plot3= fig.add_subplot(223)
data = df2[['Year','A_Students']].groupby('Year').sum()
plot3.scatter(df2['Year'],df2['A_Students'],color='red')
plot3.set_title('A-students.')

plot4= fig.add_subplot(224)
data1 = df2[['Year','A_Students']].groupby('Year').sum()
data2 = df2[['Year','F_Students']].groupby('Year').sum()
data1.plot(kind='line',color='red',legend=True, ax=plot4)
data2.plot(kind='line',color='red',legend=True, ax=plot4)
plot4.set_title('A and F students.')

# plot4 = fig.add_subplot(221)
# data = df1[['Major','Students']].groupby('Major').sum()
# data.plot(kind='bar',color='green',legend=True, ax=plot4)
# plot4.setTitle('Students in Major.')

canvas = FigureCanvasTkAgg(fig, master = root)
canvas.get_tk_widget().pack()

root.geometry('800x800')
root.mainloop()
```

In [ ]:

```
from tkinter import *
from tkinter.font import *
from PIL import ImageTk,Image
from functools import partial

root = Tk()
root.geometry("560x410")
def showImage(text):
    if(text == 'Cat'):
        img = Image.open("data/Image/cat.png")
    elif(text == 'Dog'):
        img = Image.open("data/Image/dog.jpg")
    img_resize = img.resize((530,300), Image.ANTIALIAS)
    file_img = ImageTk.PhotoImage(img_resize)
    outputLabel.create_image(0,0,anchor=NW,image=file_img)
    outputLabel.image =file_img
def clear():
    outputLabel.delete('all')
l=Label(text='Show image',font='none 20',width=35,anchor=CENTER).place(x=10,y=5)
catBT = Button(text ='Cat',width = 15,anchor=CENTER,command=partial(showImage,'Cat'))
dogtBT = Button(text ='Dog',width = 15,anchor=CENTER,command=partial(showImage,'Dog'))
pigBT = Button(text ='Pig',width = 15,anchor=CENTER)
clearBT = Button(text ='Clear',width = 15,anchor=CENTER,command = clear)
outputLabel = Canvas(width=530,height=300,borderwidth=2,relief='groove')

catBT.place(x=10,y=50)
dogtBT.place(x=150,y=50)
pigBT.place(x=290,y=50)
clearBT.place(x=430,y=50)
outputLabel.place(x=10,y=90)
```

```
root.mainloop()
```