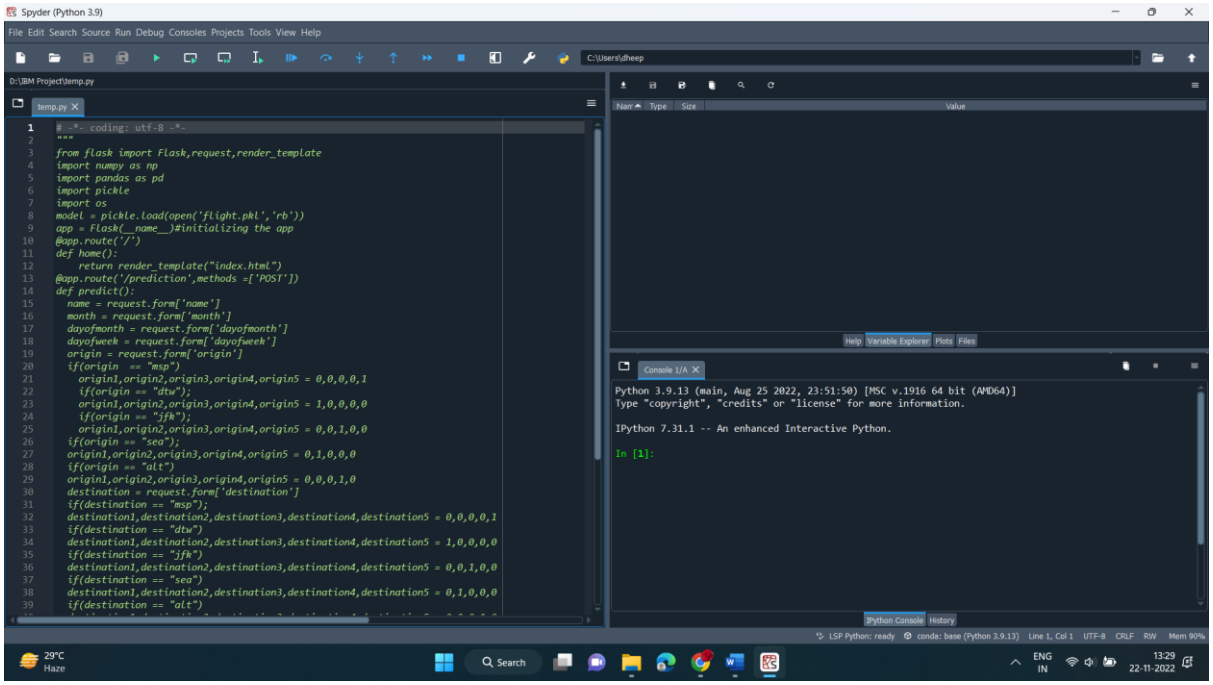
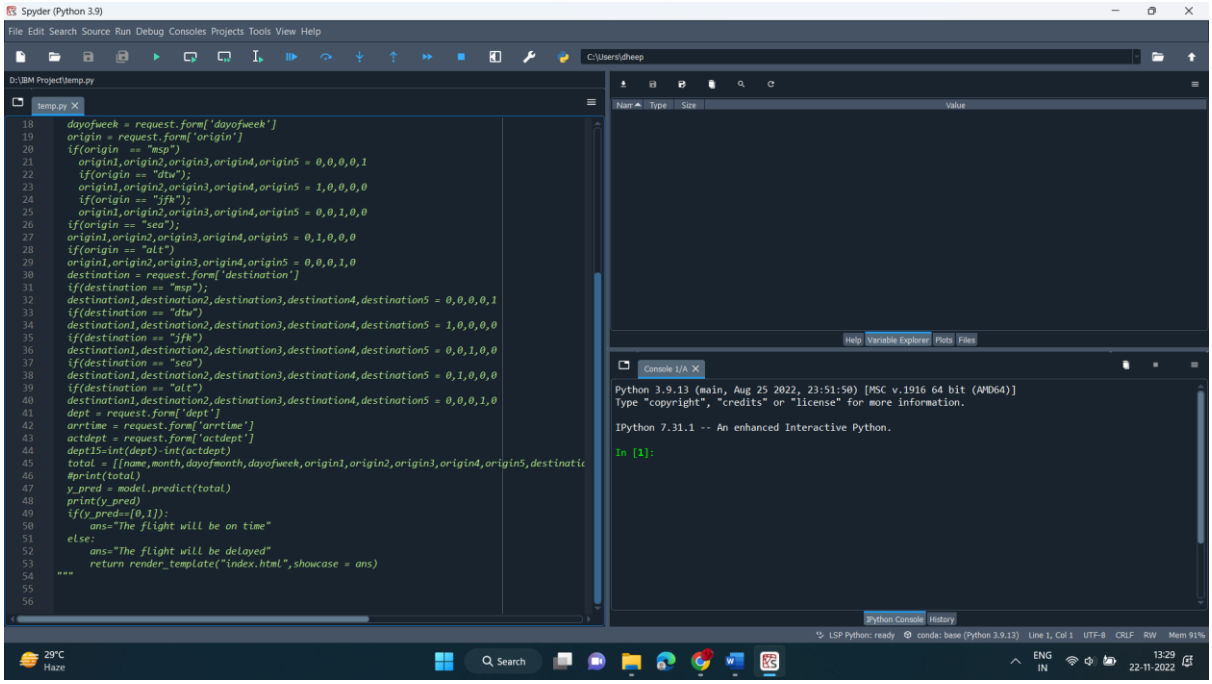
INTEGRATE FLASK WITH SCORING END POINT

DATE	01 NOVEMBER 2022
TEAM ID	PNT2022TMID17753
PROJECT NAME	DEVELOPING A FLIGHT DELAY PREDICTION MODEL BY USING MACHINE LEARNING





def home():

return render_template('index.html')

def y_predict():

geography = request.form("geography")

gender= request.form("gender")

age= request.form("age")

tenure= request.form("tenure")

creditscore= request.form("creditscore")

balance= request.form("balance")

noof = request.form["no of"]

hascreditcard = request.form{"has credit card"]

isactivemember = request.form{"isactivemember"]

estimatedsalary = request.form["estimated salary"]

if(geography == "Spain"):

s1,s2,s3 = 0,0,1

if(geography == "Germany"):

s1,s2,s3 = 0,1,0

```
if(geography == “Newyork”):
s1,s2,s3 = 0,1,0
if(gender ==”female”):
gender = 0
if(gender == “male”):
gender = 1
if(isactivemember == “no”):
isactivemember = 0
if(isactivemember == “yes”):
isactivemember = 1
if(hascreditcard == “no”):
hascreditcard = 0
if(hascreditcard == “yes”):
hascreditcard = 1
t=[[int(s1),int(s2),int(s3),int(creditscore),int(gender),int(age),int(tenure),int(balance),int(noof)
,int(hascreditcard),int(isactivemember),int(estimatedsalary)]]
print(t)
payload_scoring =
(“input_data”:[{“field”:[[“G1”,”G2”,”G3”,”CreditScore”,”Gender”,”Age”,”Tenure”,”Balance
”,”numofproducts”,”HasCrCard”,”IsActiveMember”)
print(“Scoring response’)
predictions = response_scoring.json()
print(predictions)
pred = predictions[‘predictions’][0][‘values’][0][0]
if(pred == 0):
output = “he will not get excited”
print(“ he will not get excited”)
else:
output = “he gets excited”
print(“he gets excited”)
return render_template(“index.html’,prediction_text=output)
```

```
if_name_ == “_main_”:
app.run(debug=True)
```