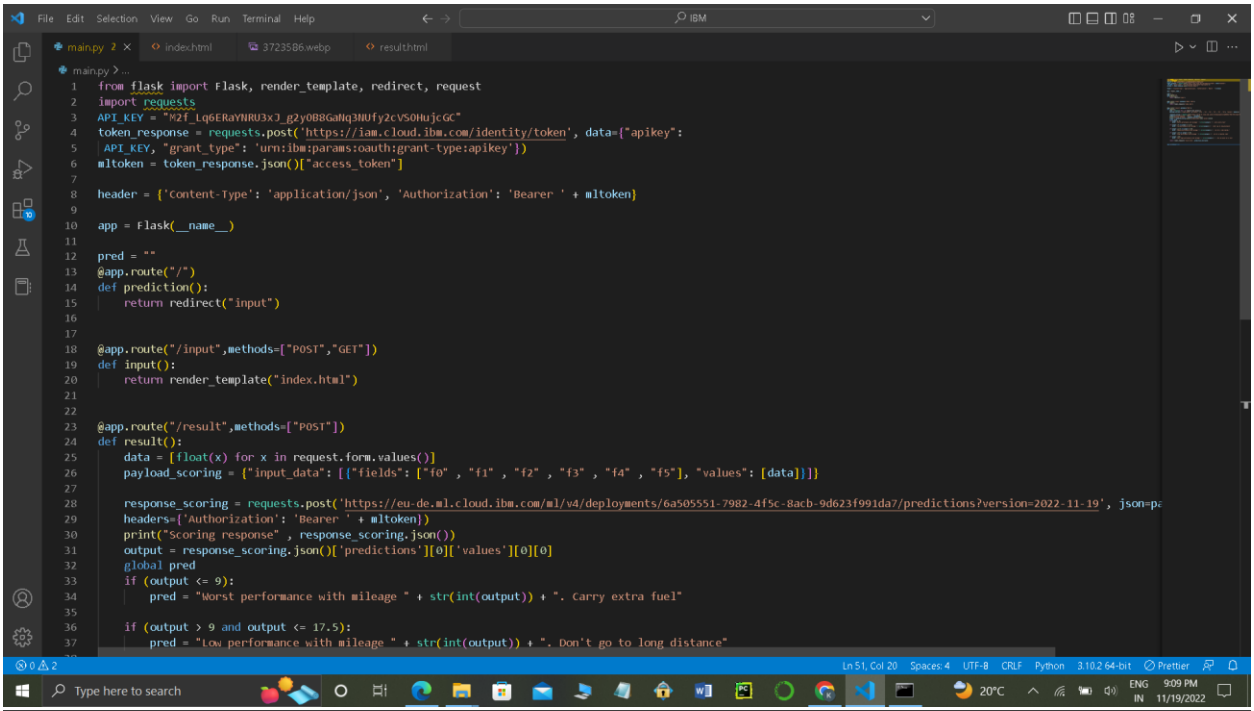
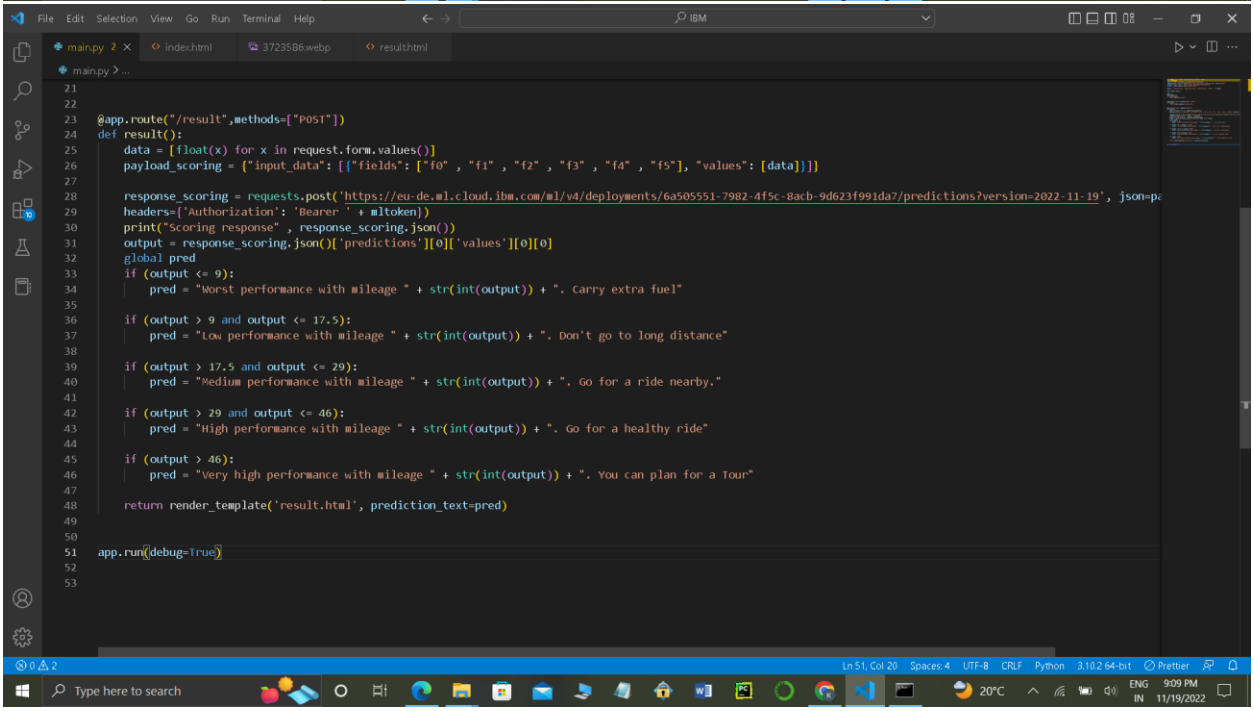
Integrate Flask with Scoring End Point :

Team ID — **PNT2022TMID22357**



```python
from flask import Flask, render_template, redirect, request
import requests
API_KEY = "M2f_Lq6ERaYNRU3xJ_g2yOB8GaNq3NUfy2cVSOHujcGC"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
 API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}

app = Flask(__name__)

pred = ""
@app.route("/")
def prediction():
    return redirect("input")


@app.route("/input",methods=["POST","GET"])
def input():
    return render_template("index.html")


@app.route("/result",methods=["POST"])
def result():
    data = [float(x) for x in request.form.values()]
    payload_scoring = {"input_data": [{"fields": ["f0" , "f1" , "f2" , "f3" , "f4" , "f5"], "values": [data]}]}

    response_scoring = requests.post('https://eu-de.ml.cloud.ibm.com/ml/v4/deployments/6a505551-7982-4f5c-8acb-9d623f991da7/predictions?version=2022-11-19', json=pa
    headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response" , response_scoring.json())
    output = response_scoring.json()['predictions'][0]['values'][0][0]
    global pred
    if (output <= 9):
        pred = "Worst performance with mileage " + str(int(output)) + ". Carry extra fuel"

    if (output > 9 and output <= 17.5):
        pred = "Low performance with mileage " + str(int(output)) + ". Don't go to long distance"
```



```python
@app.route("/result",methods=["POST"])
def result():
    data = [float(x) for x in request.form.values()]
    payload_scoring = {"input_data": [{"fields": ["f0" , "f1" , "f2" , "f3" , "f4" , "f5"], "values": [data]}]}

    response_scoring = requests.post('https://eu-de.ml.cloud.ibm.com/ml/v4/deployments/6a505551-7982-4f5c-8acb-9d623f991da7/predictions?version=2022-11-19', json=pa
    headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response" , response_scoring.json())
    output = response_scoring.json()['predictions'][0]['values'][0][0]
    global pred
    if (output <= 9):
        pred = "Worst performance with mileage " + str(int(output)) + ". Carry extra fuel"

    if (output > 9 and output <= 17.5):
        pred = "Low performance with mileage " + str(int(output)) + ". Don't go to long distance"

    if (output > 17.5 and output <= 29):
        pred = "Medium performance with mileage " + str(int(output)) + ". Go for a ride nearby."

    if (output > 29 and output <= 46):
        pred = "High performance with mileage " + str(int(output)) + ". Go for a healthy ride"

    if (output > 46):
        pred = "Very high performance with mileage " + str(int(output)) + ". You can plan for a Tour"

    return render_template('result.html', prediction_text=pred)


app.run(debug=True)
```