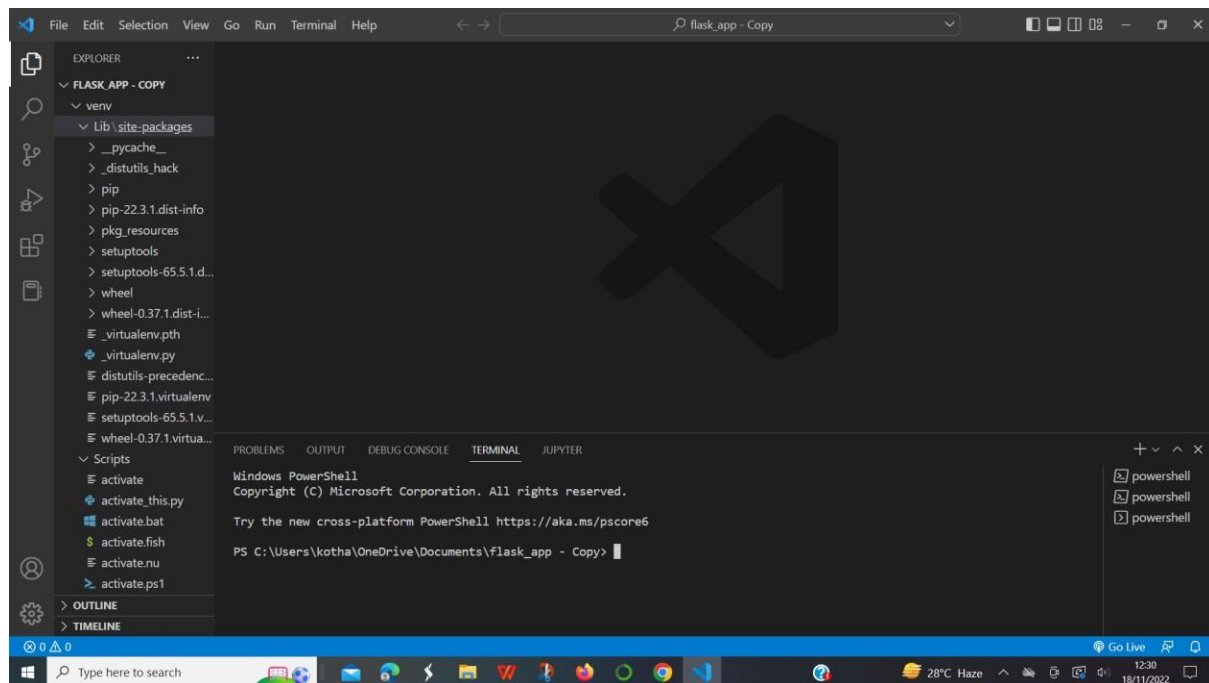


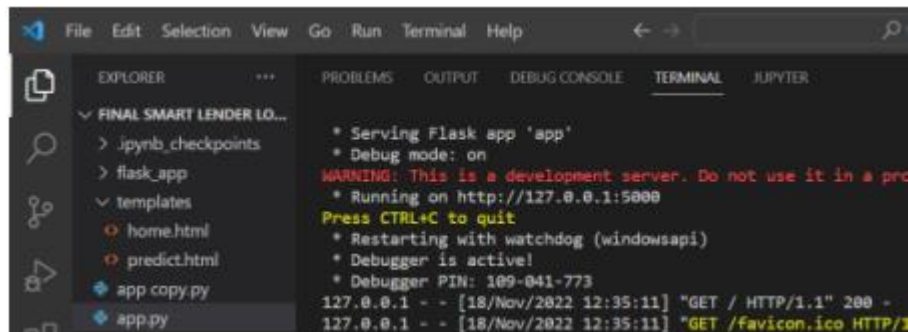
Integrate Flask with Scoring End Point

Team ID	PNT2022TMID10653
Project Name	Car resale value prediction

Create the virtual environment using the visual studio code



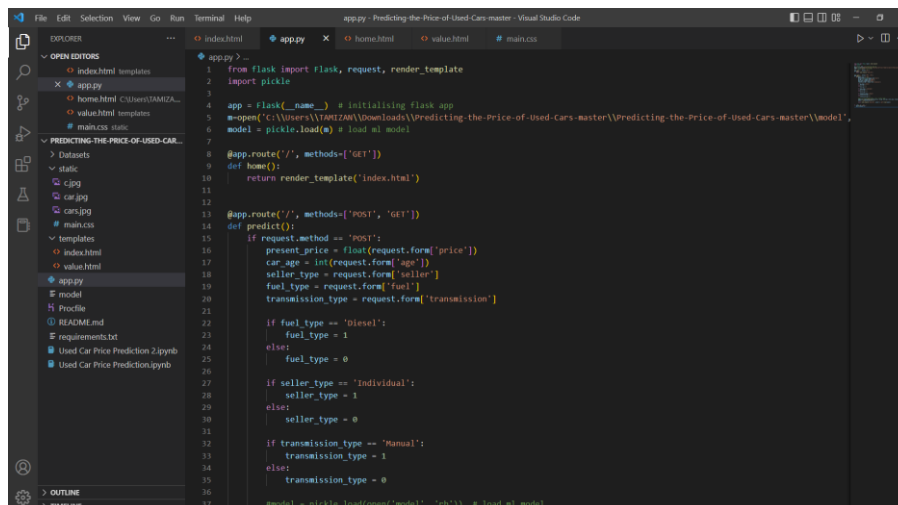
Run the python file



```
File Edit Selection View Go Run Terminal Help
EXPLORER PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
FINAL SMART LENDER LO...
> .ipynb_checkpoints
> flask_app
> templates
  home.html
  predict.html
  app copy.py
  app.py

* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with watchdog (windowsapi)
* Debugger is active!
* Debugger PIN: 109-041-773
127.0.0.1 - - [18/Nov/2022 12:35:11] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [18/Nov/2022 12:35:11] "GET /favicon.ico HTTP/1.1" 404 -
```

Use the API key that have generated in the IBM cloud in the python code



```
File Edit Selection View Go Run Terminal Help
app.py - Predicting the Price of Used Cars-master - Visual Studio Code
EXPLORER
  OPEN EDITORS
    app.py
  PREDICTING THE PRICE OF USED CAR...
    datasets
    static
    r.jpg
    car.jpg
    cars.jpg
    main.css
    templates
      index.html
      value.html
    app.py
    model
    profile
    README.md
    requirements.txt
    Used Car Price Prediction 2.ipynb
    Used Car Price Prediction.ipynb
  OUTLINE
  TIMELINE

1 from flask import flask, request, render_template
2 import pickle
3
4 app = flask(__name__) # initializing flask app
5 #open('C:\Users\VAAGI\Downloads\Predicting the Price of Used Cars-master\Predicting the Price of Used Cars-master\model',
6 model = pickle.load(s) # load ai model
7
8 @app.route('/', methods=['GET'])
9 def home():
10     return render_template('index.html')
11
12
13 @app.route('/', methods=['POST', 'GET'])
14 def predict():
15     if request.method == 'POST':
16         present_price = float(request.form['price'])
17         car_age = int(request.form['age'])
18         seller_type = request.form['seller']
19         fuel_type = request.form['fuel']
20         transmission_type = request.form['transmission']
21
22         if fuel_type == 'Diesel':
23             fuel_type = 1
24         else:
25             fuel_type = 0
26
27         if seller_type == 'Individual':
28             seller_type = 1
29         else:
30             seller_type = 0
31
32         if transmission_type == 'Manual':
33             transmission_type = 1
34         else:
35             transmission_type = 0
36
37     model = pickle.load(open('model', 'rb')) # load ai model
```

Using the Api key, we are integrating the flask application

