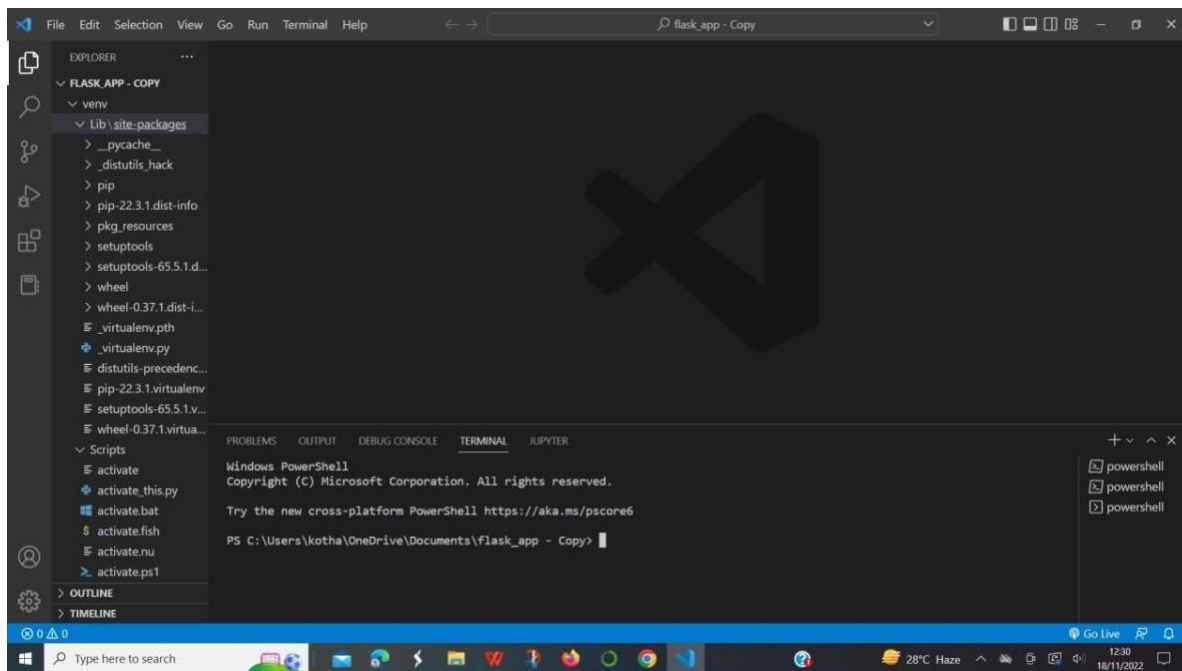



## Integrate Flask with Scoring End Point

Team ID	PNT2022TMID47865
Project Name	Car resale value prediction

Create the virtual environment using the visual studio code



Run the python file



The screenshot shows the VS Code interface with the following panels and content:

- Explorer:**
  - FINAL SMART LENDER LO...
    - jpynb\_checkpoints
    - flask\_app
    - templates
      - home.html
      - predict.html
    - app.copy.py
    - app.py
- Problems:** (Empty)
- Output:** (Empty)
- Debug Console:** (Empty)
- Terminal:**

```

* 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

The screenshot shows a Visual Studio Code editor with a Python Flask application. The file explorer on the left lists the project structure, including templates, static files, and a 'predicting-the-price-of-used-car-master' folder. The main editor displays the code for app.py, which includes imports for Flask and pickle, initialization of the app, loading of a model, and routes for GET and POST requests to predict car prices based on form inputs.

```

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

Using the Api key, we are integrating the flask application

