Document
Week4: Deployment on Flask

Name: Fatimah Asiri
Username: alassirifatima@gmail.com
Batch code: LISUM14 30 Sep – 30 Dec 2022
Submission Date: 10/23/2022
Submitted to: Data Glacier.

This document contains a snapshot of each step of the deployment

(Deployment Machine Learning on Flask)

ML Model Deployment using Flask

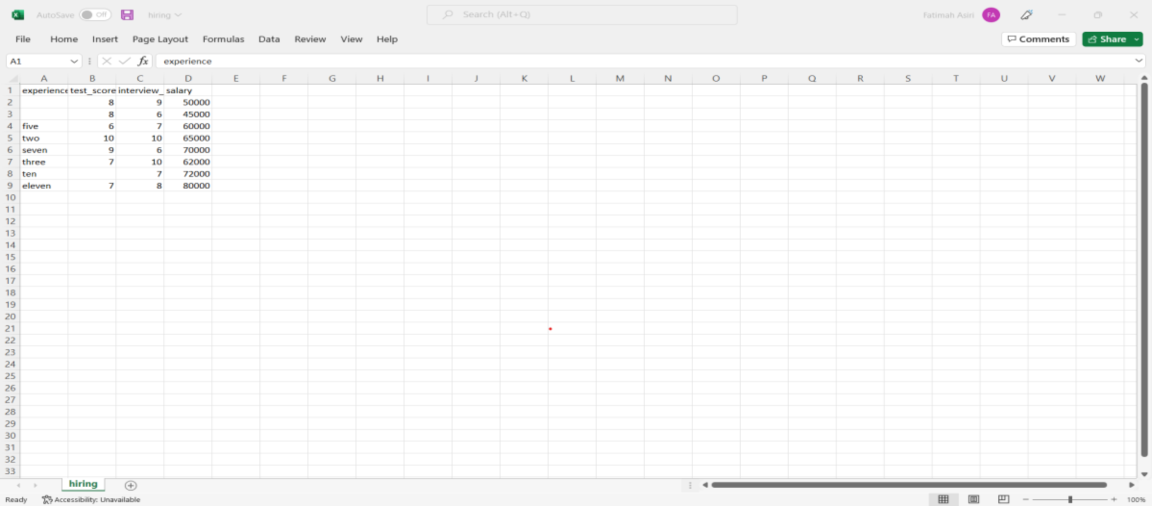
1. Build ML Model.
2. Deploy using Flask.

Files to be created

1. model.py (ML model)
2. model.pkl (Pickle file of ML model)
3. app.py (Flask Application)
4. index.html (inside the folder templates)
5. hiring dataset (data to build ML model)

1. Choose a toy dataset.

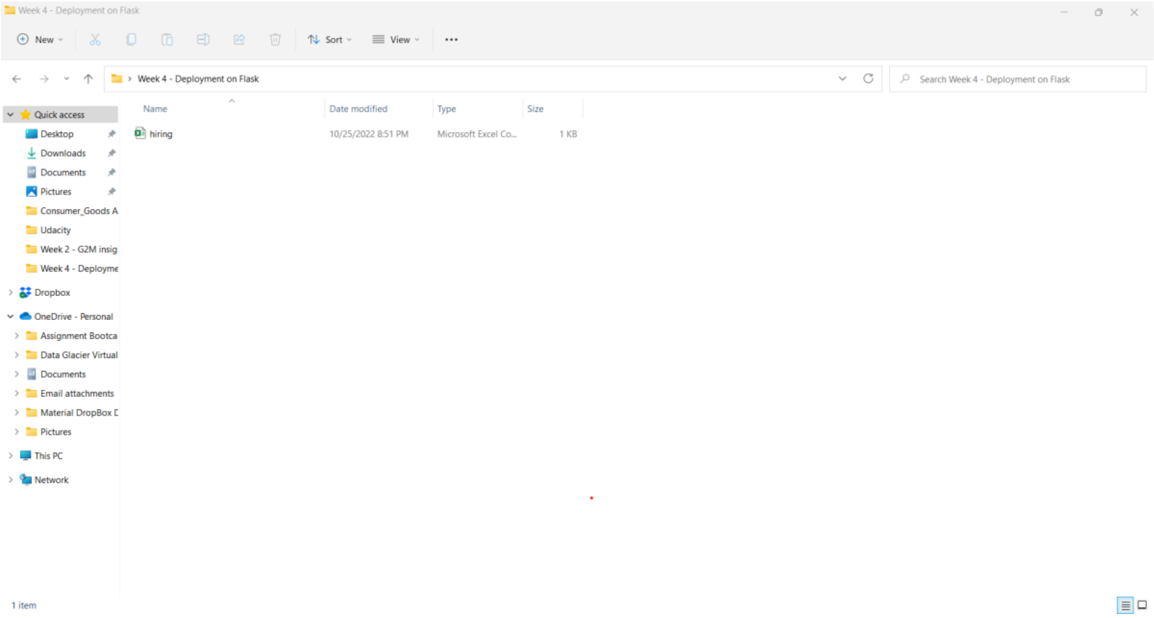
Select the hiring dataset.



AutoSave Off
hiring
Search (Alt+Q)
Fatimah Asiri
File Home Insert Page Layout Formulas Data Review View Help
Comments
Share
A1
experience
hiring
Ready
Accessibility: Unavailable
100%

experience	test_score	interview_	salary
	8	9	50000
	8	6	45000
five	6	7	60000
two	10	10	65000
seven	9	6	70000
three	7	10	62000
ten		7	72000
eleven	7	8	80000

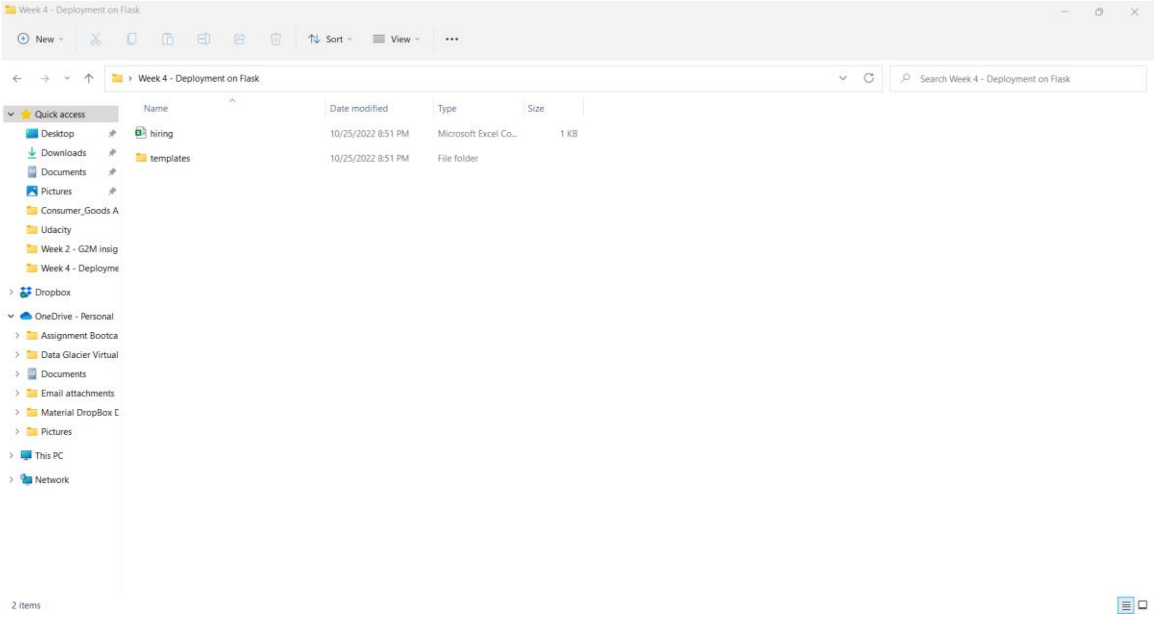
2. **Download csv file inside the folder**



Name	Date modified	Type	Size
hiring	10/25/2022 8:51 PM	Microsoft Excel Co...	1 KB

1 item

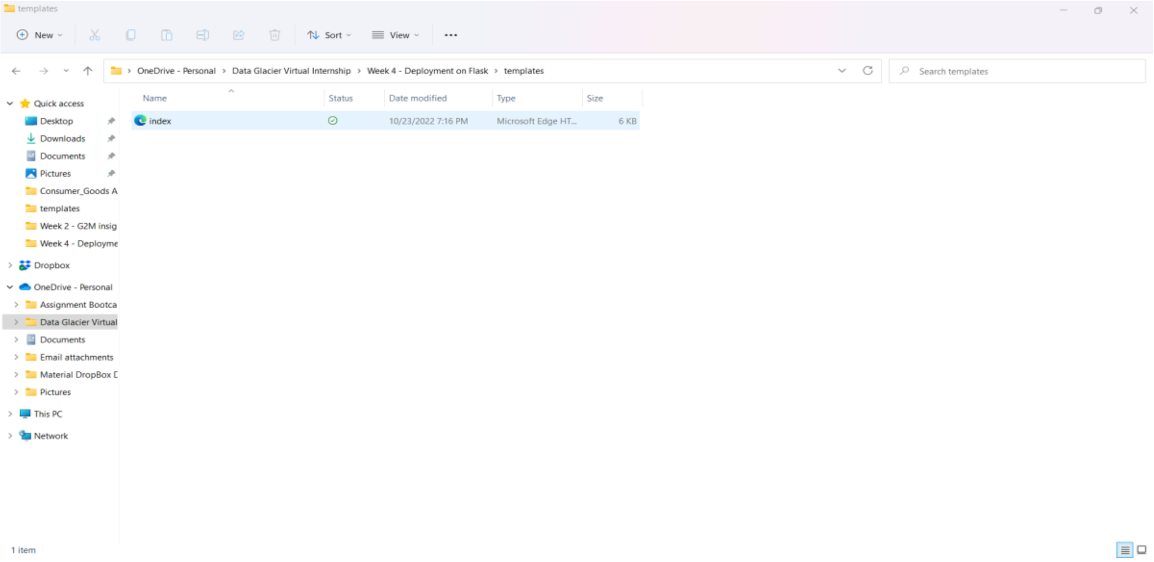
3. Create new folder (templates).



Name	Date modified	Type	Size
hiring	10/25/2022 8:51 PM	Microsoft Excel Co...	1 KB
templates	10/25/2022 8:51 PM	File folder	

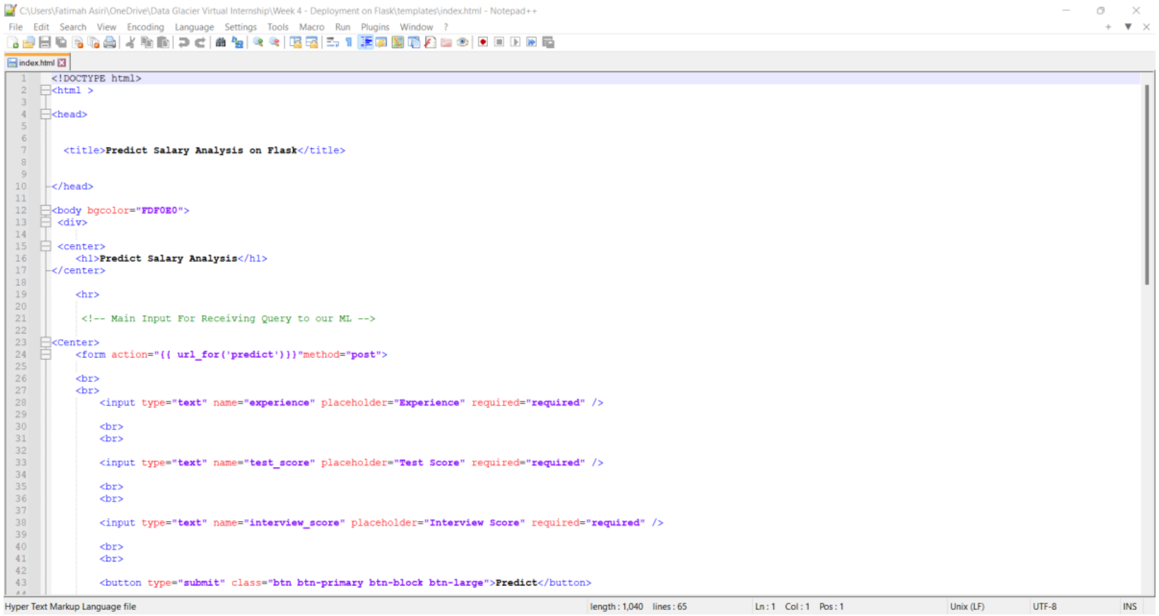
2 items

4. Create index.html file in the templates folder.



OneDrive - Personal › Data Glacier Virtual Internship › Week 4 - Deployment on Flask › templates

Name	Status	Date modified	Type	Size
index	✓	10/23/2022 7:16 PM	Microsoft Edge HT...	6 KB

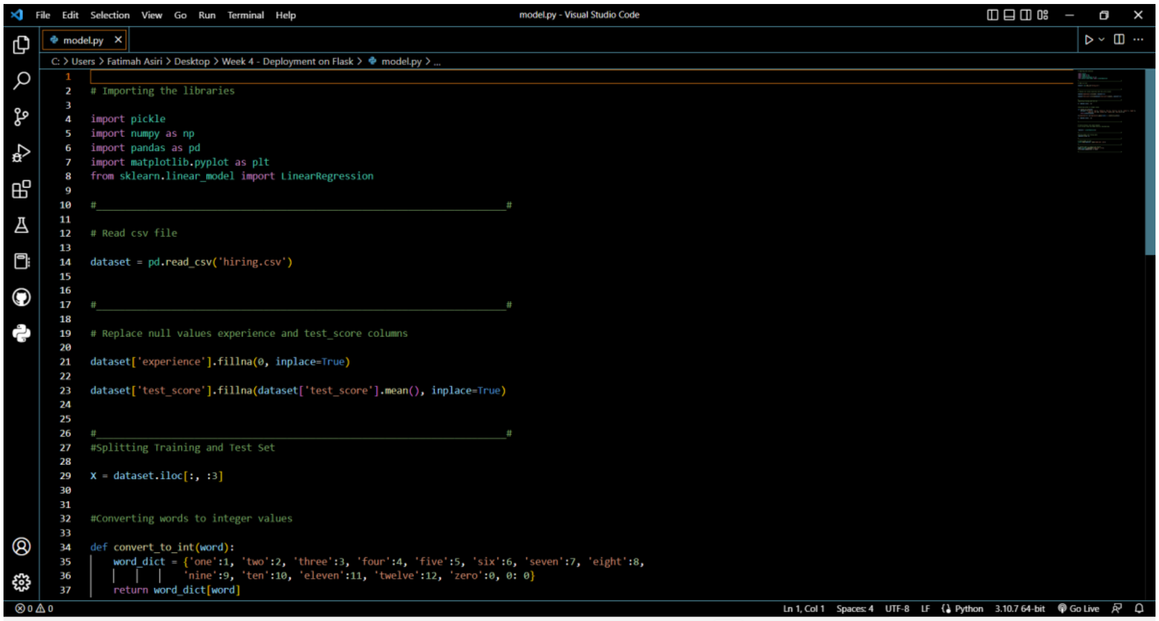


C:\Users\Fatimah Asiri\OneDrive\Data Glacier Virtual Internship\Week 4 - Deployment on Flask\templates\index.html - Notepad++

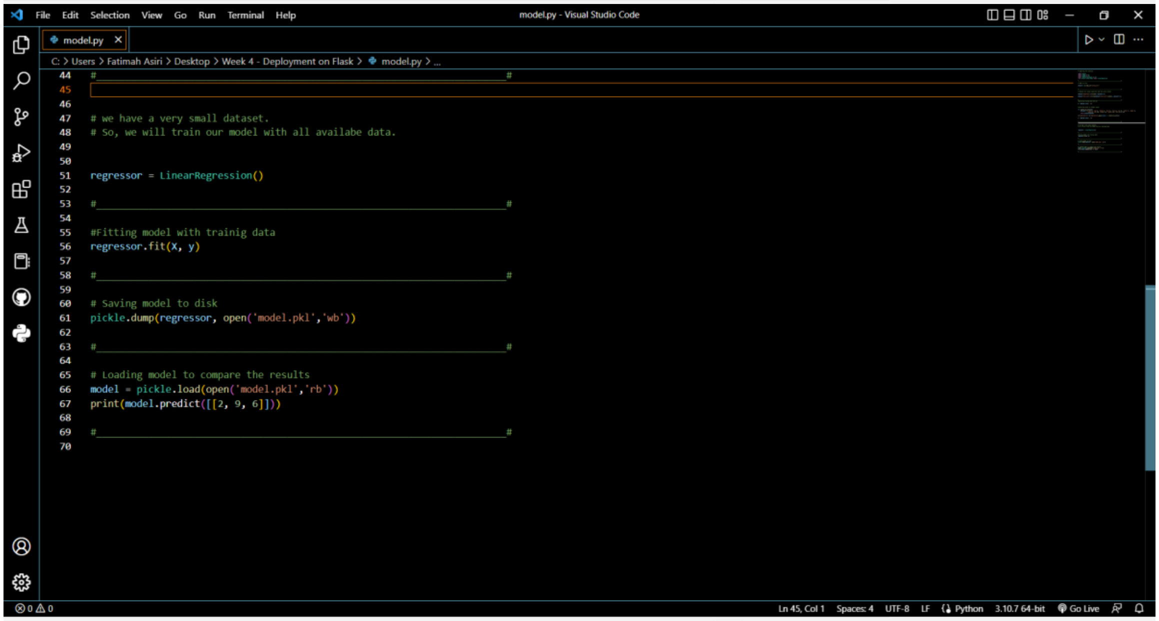
```html
<!DOCTYPE html>
<html >

<head>


  <title>Predict Salary Analysis on Flask</title>


</head>

<body bgcolor="FDF0E0">
 <div>

 <center>
    <h1>Predict Salary Analysis</h1>
</center>

    <hr>

     <!-- Main Input For Receiving Query to our ML -->

<Center>
    <form action="{{ url_for('predict')}}"method="post">

    <br>
    <br>
        <input type="text" name="experience" placeholder="Experience" required="required" />

        <br>
        <br>

        <input type="text" name="test_score" placeholder="Test Score" required="required" />

        <br>
        <br>

        <input type="text" name="interview_score" placeholder="Interview Score" required="required" />

        <br>
        <br>

        <button type="submit" class="btn btn-primary btn-block btn-large">Predict</button>
```

Hyper Text Markup Language file   length : 1,040   lines : 65   Ln : 1   Col : 1   Pos : 1   Unix (LF)   UTF-8   INS

5. Create model.py file


File
Edit
Selection
View
Go
Run
Terminal
Help
model.py - Visual Studio Code
model.py
C: > Users > Fatimah Asiri > Desktop > Week 4 - Deployment on Flask > model.py > ...
⊗ 0 ⚠ 0
Ln 1, Col 1
Spaces: 4
UTF-8
LF
Python
3.10.7 64-bit
Go Live

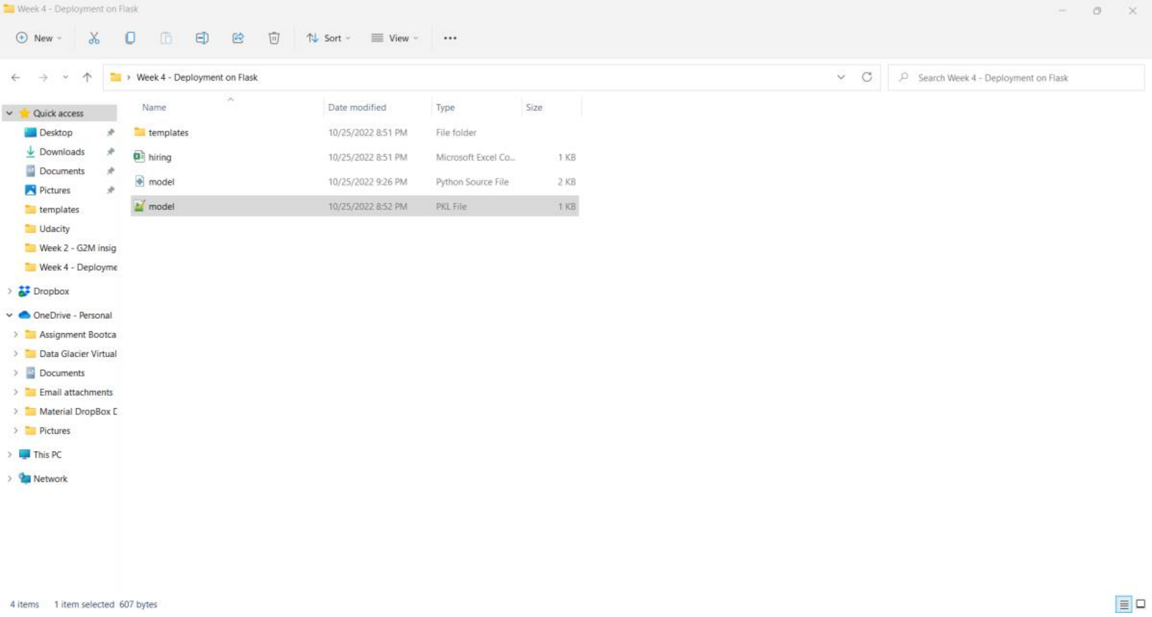
```python

# Importing the libraries

import pickle
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

#______________________________________________________________________#

# Read csv file

dataset = pd.read_csv('hiring.csv')


#______________________________________________________________________#

# Replace null values experience and test_score columns

dataset['experience'].fillna(0, inplace=True)

dataset['test_score'].fillna(dataset['test_score'].mean(), inplace=True)


#______________________________________________________________________#
#Splitting Training and Test Set

X = dataset.iloc[:, :3]


#Converting words to integer values

def convert_to_int(word):
    word_dict = {'one':1, 'two':2, 'three':3, 'four':4, 'five':5, 'six':6, 'seven':7, 'eight':8,
                'nine':9, 'ten':10, 'eleven':11, 'twelve':12, 'zero':0, 0: 0}
    return word_dict[word]
```


File
Edit
Selection
View
Go
Run
Terminal
Help
model.py - Visual Studio Code
model.py
C: > Users > Fatimah Asiri > Desktop > Week 4 - Deployment on Flask > model.py > ...
⊗ 0 ⚠ 0
Ln 45, Col 1
Spaces: 4
UTF-8
LF
Python
3.10.7 64-bit
Go Live

```python
#______________________________________________________________________#


# we have a very small dataset.
# So, we will train our model with all availabe data.


regressor = LinearRegression()

#______________________________________________________________________#

#Fitting model with trainig data
regressor.fit(X, y)

#______________________________________________________________________#

# Saving model to disk
pickle.dump(regressor, open('model.pkl','wb'))

#______________________________________________________________________#

# Loading model to compare the results
model = pickle.load(open('model.pkl','rb'))
print(model.predict([[2, 9, 6]]))

#______________________________________________________________________#

```

6. Create Pickle file of our model

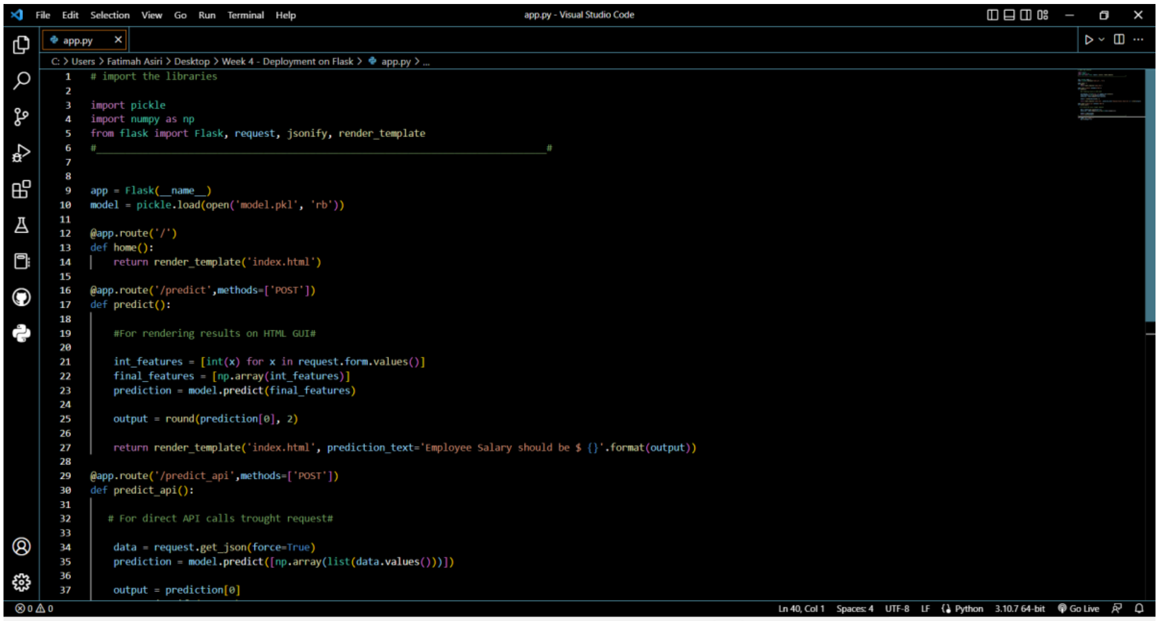
model.pkl



Name	Date modified	Type	Size
templates	10/25/2022 8:51 PM	File folder	
hiring	10/25/2022 8:51 PM	Microsoft Excel Co...	1 KB
model	10/25/2022 9:26 PM	Python Source File	2 KB
model	10/25/2022 8:52 PM	PKL File	1 KB

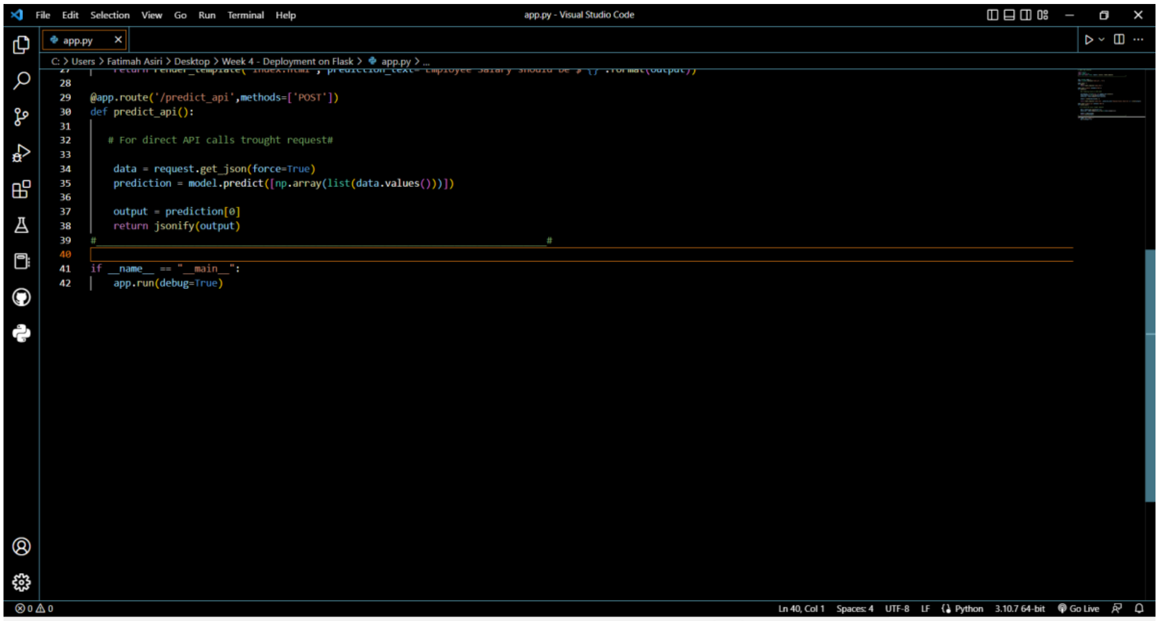
4 items 1 item selected 607 bytes

7. Create app.py files



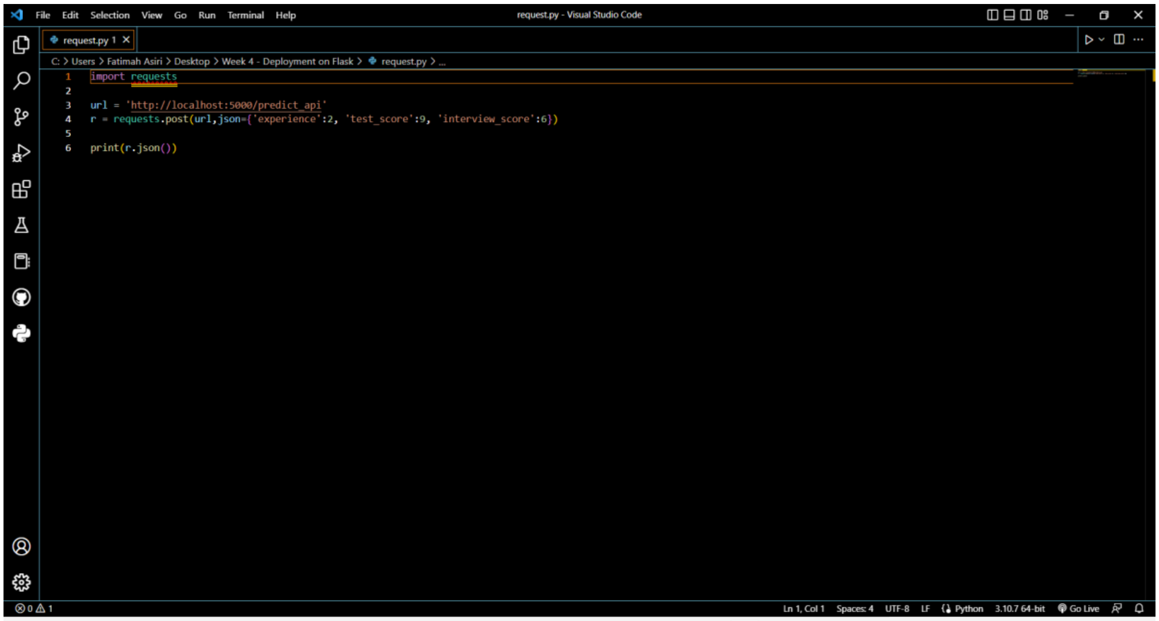
File
Edit
Selection
View
Go
Run
Terminal
Help
app.py - Visual Studio Code
app.py
C: > Users > Fatimah Asiri > Desktop > Week 4 - Deployment on Flask > app.py > ...
⊗ 0 ⚠ 0
Ln 40, Col 1
Spaces: 4
UTF-8
LF
Python
3.10.7 64-bit
Go Live

```python
# import the libraries

import pickle
import numpy as np
from flask import Flask, request, jsonify, render_template
#_____________________________________________________________________#


app = Flask(__name__)
model = pickle.load(open('model.pkl', 'rb'))

@app.route('/')
def home():
    return render_template('index.html')

@app.route('/predict',methods=['POST'])
def predict():

    #For rendering results on HTML GUI#

    int_features = [int(x) for x in request.form.values()]
    final_features = [np.array(int_features)]
    prediction = model.predict(final_features)

    output = round(prediction[0], 2)

    return render_template('index.html', prediction_text='Employee Salary should be $ {}'.format(output))

@app.route('/predict_api',methods=['POST'])
def predict_api():

   # For direct API calls trought request#

    data = request.get_json(force=True)
    prediction = model.predict([np.array(list(data.values()))])

    output = prediction[0]
```


File Edit Selection View Go Run Terminal Help
app.py - Visual Studio Code
app.py
C: > Users > Fatimah Asiri > Desktop > Week 4 - Deployment on Flask > app.py > ...
Ln 40, Col 1 Spaces: 4 UTF-8 LF Python 3.10.7 64-bit Go Live

```python
    return render_template('index.html', prediction_text='Employee Salary should be $ {}'.format(output))

@app.route('/predict_api',methods=['POST'])
def predict_api():

    # For direct API calls trought request#

    data = request.get_json(force=True)
    prediction = model.predict([np.array(list(data.values()))])

    output = prediction[0]
    return jsonify(output)
#_____________________________________________________________#

if __name__ == "__main__":
    app.run(debug=True)
```

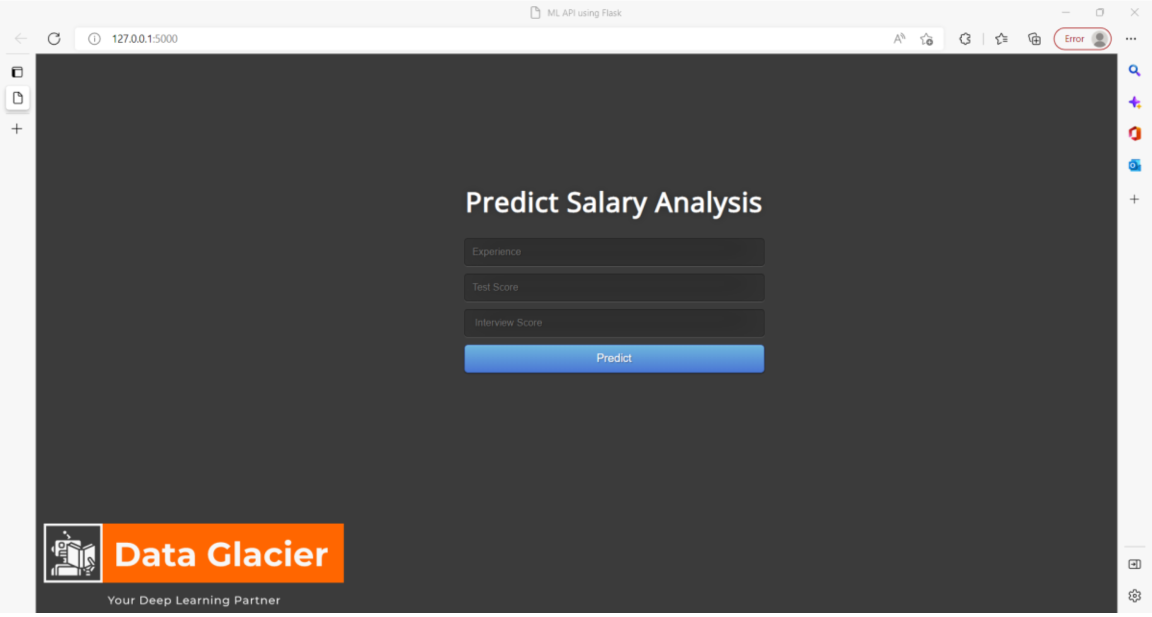
8.Create request.py


File Edit Selection View Go Run Terminal Help
request.py - Visual Studio Code
request.py 1
C: > Users > Fatimah Asiri > Desktop > Week 4 - Deployment on Flask > request.py > ...
Ln 1, Col 1 Spaces: 4 UTF-8 LF Python 3.10.7 64-bit Go Live

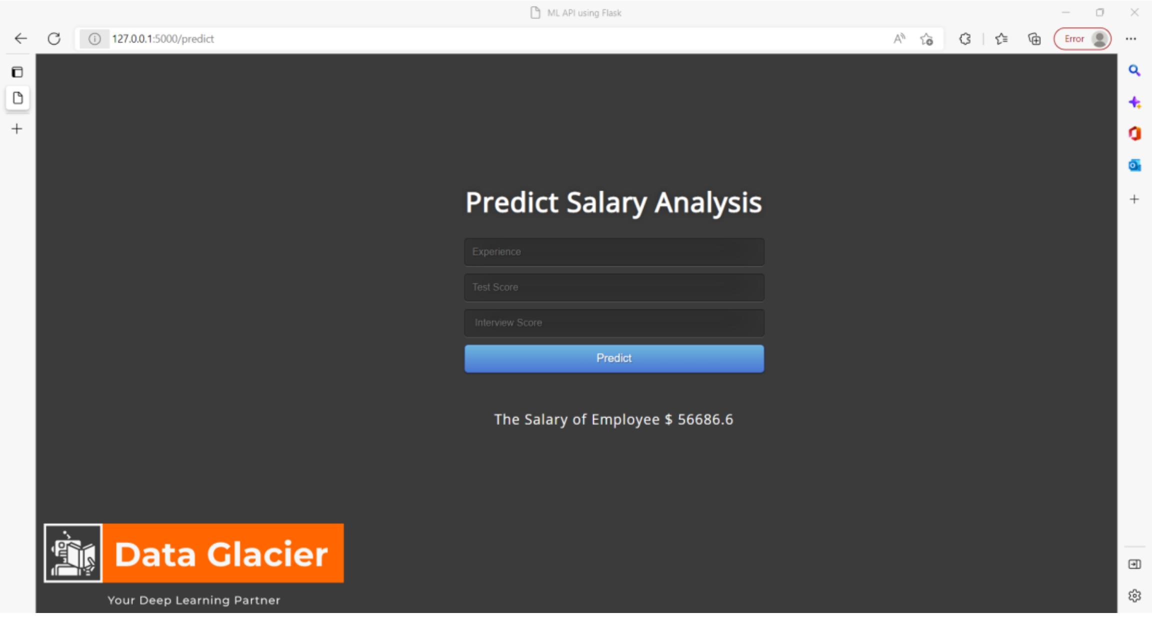
```python
import requests

url = 'http://localhost:5000/predict_api'
r = requests.post(url,json={'experience':2, 'test_score':9, 'interview_score':6})

print(r.json())
```

9. Read me file for GitHub

10. The Result of running app



Predict Salary Analysis



Predict Salary Analysis

The Salary of Employee $ 56686.6