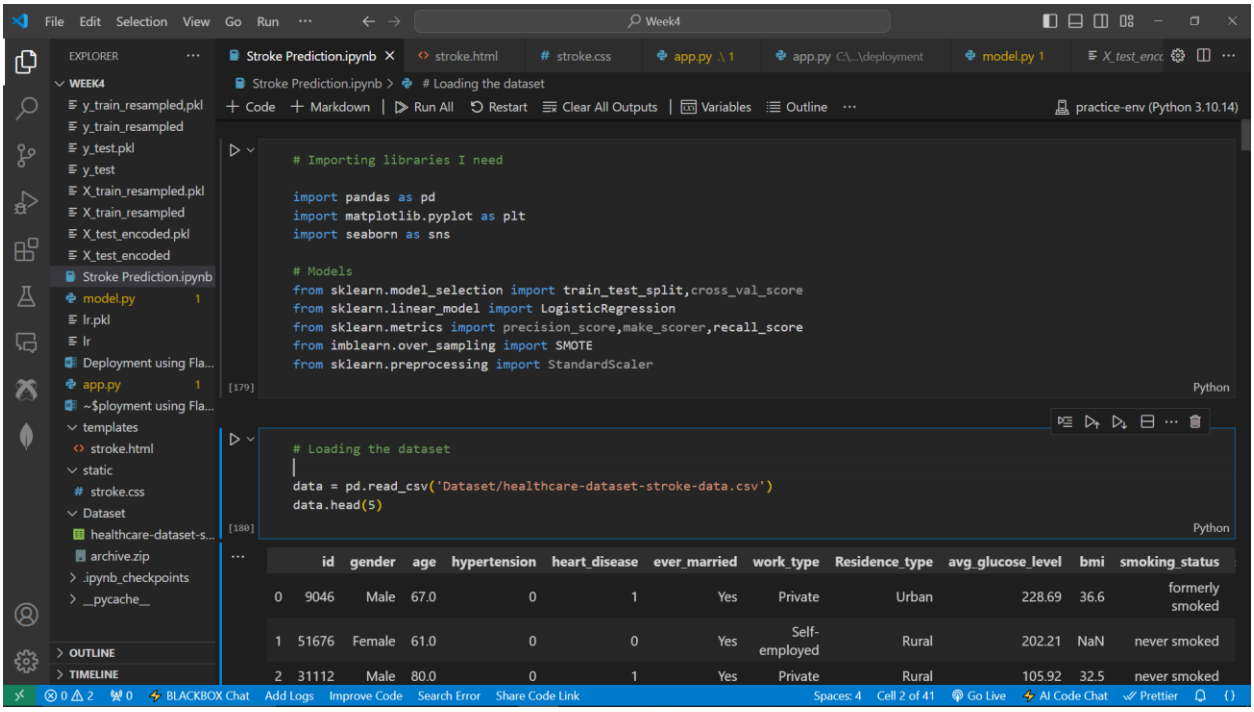
STEPS OF DEPLOYMENT USING FLASK

NAME: VIVIAN KERUBO MOSOMI

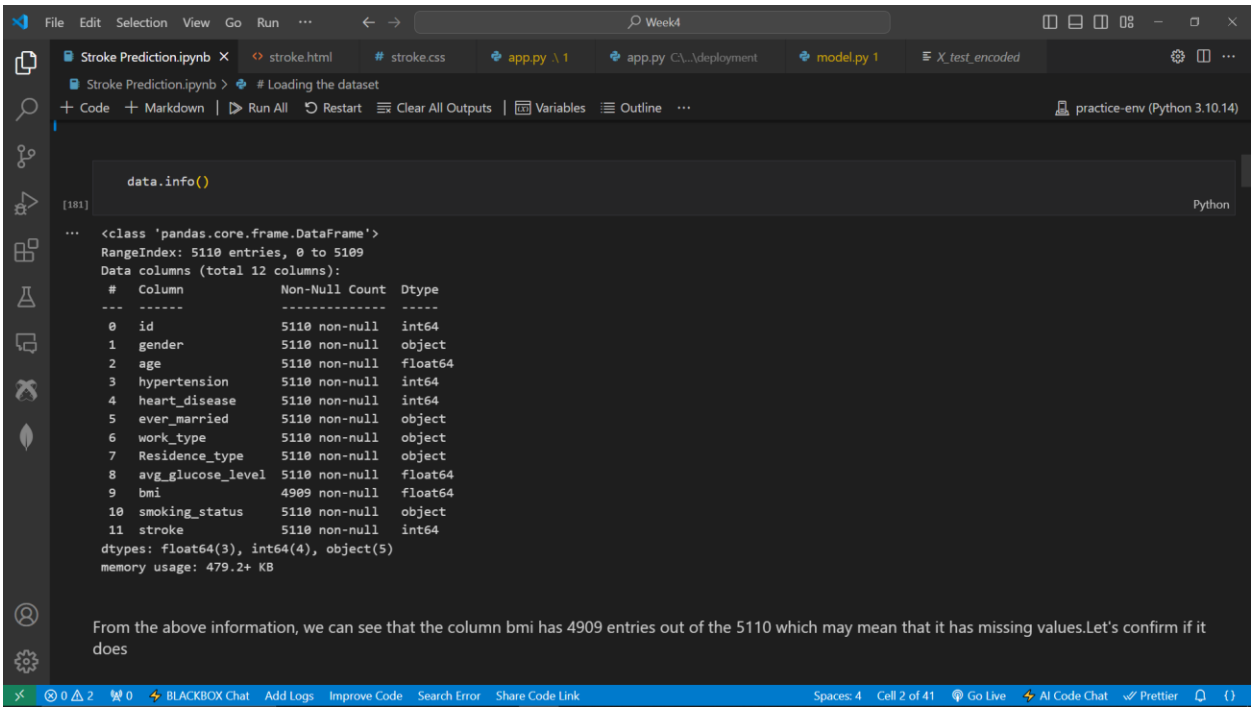
BATCH CODE: LISUM39

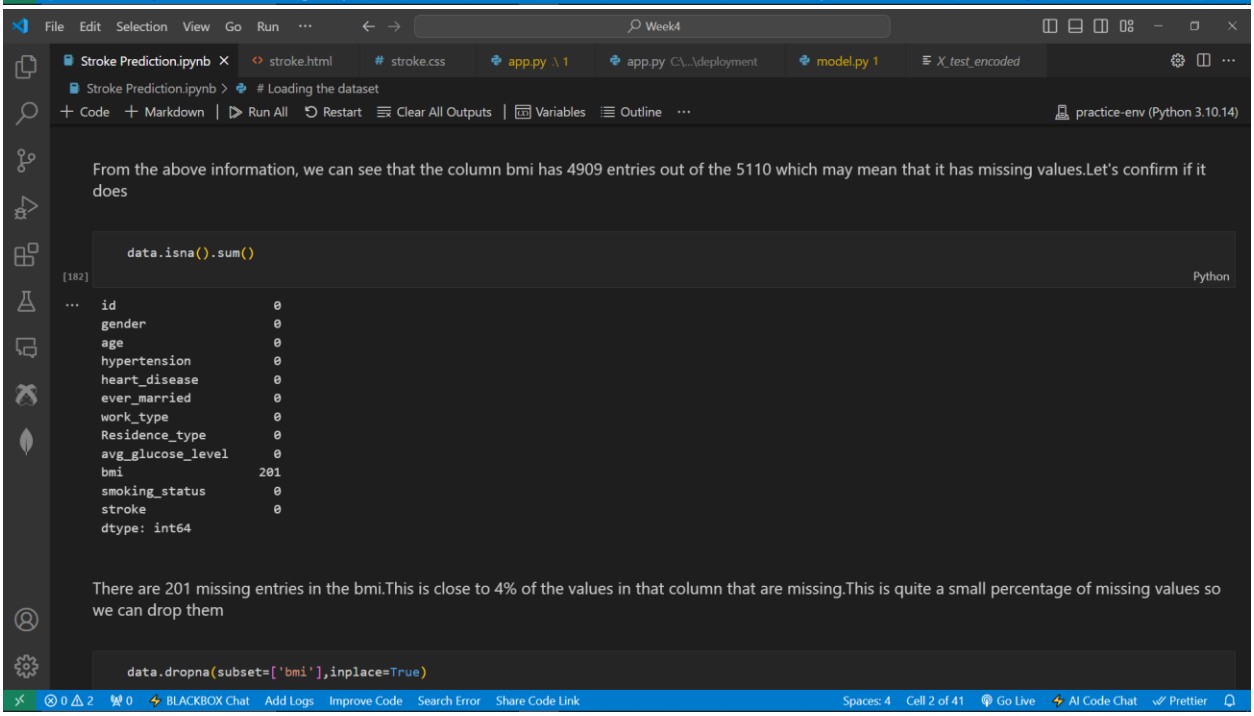
SUBMISSION DATE: 27/11/2024

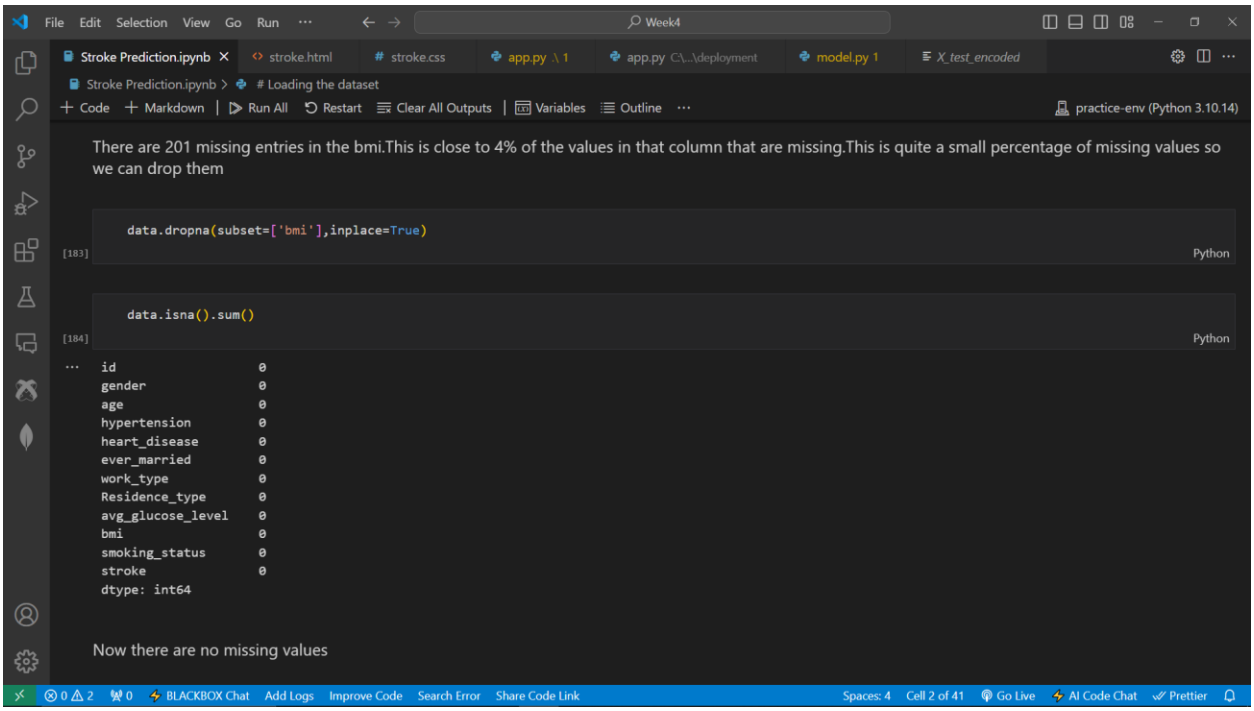
1.Loading the data

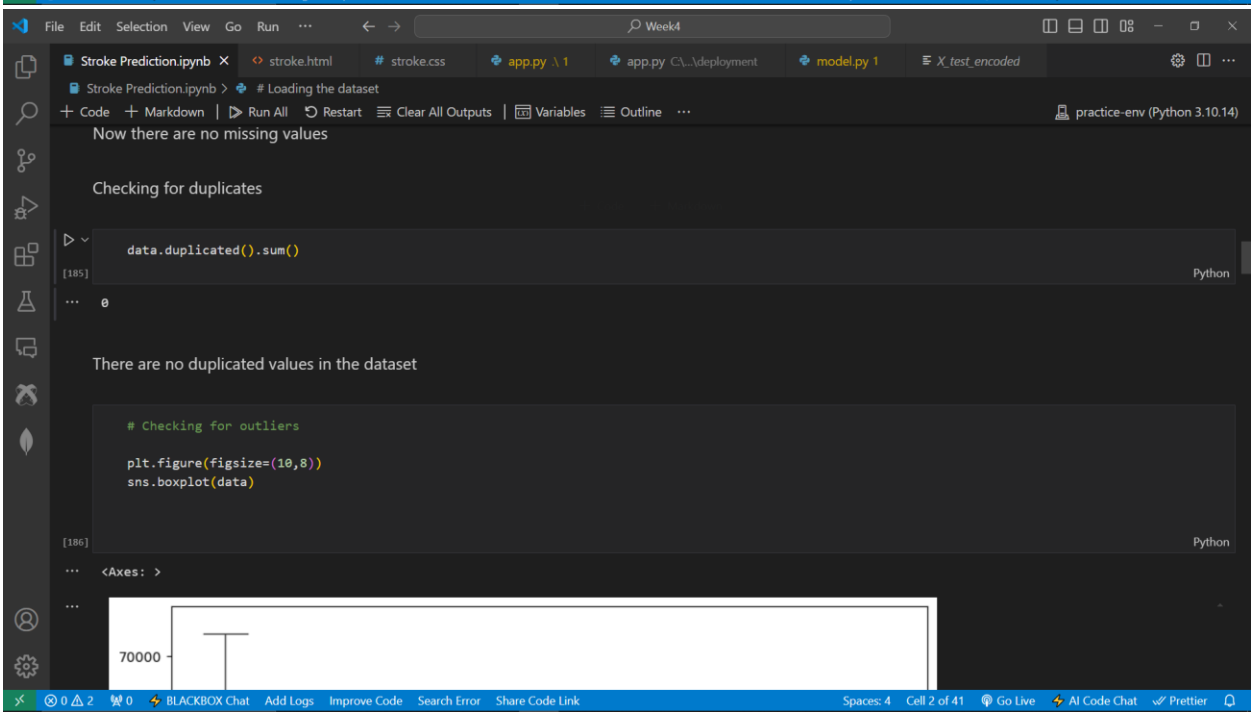


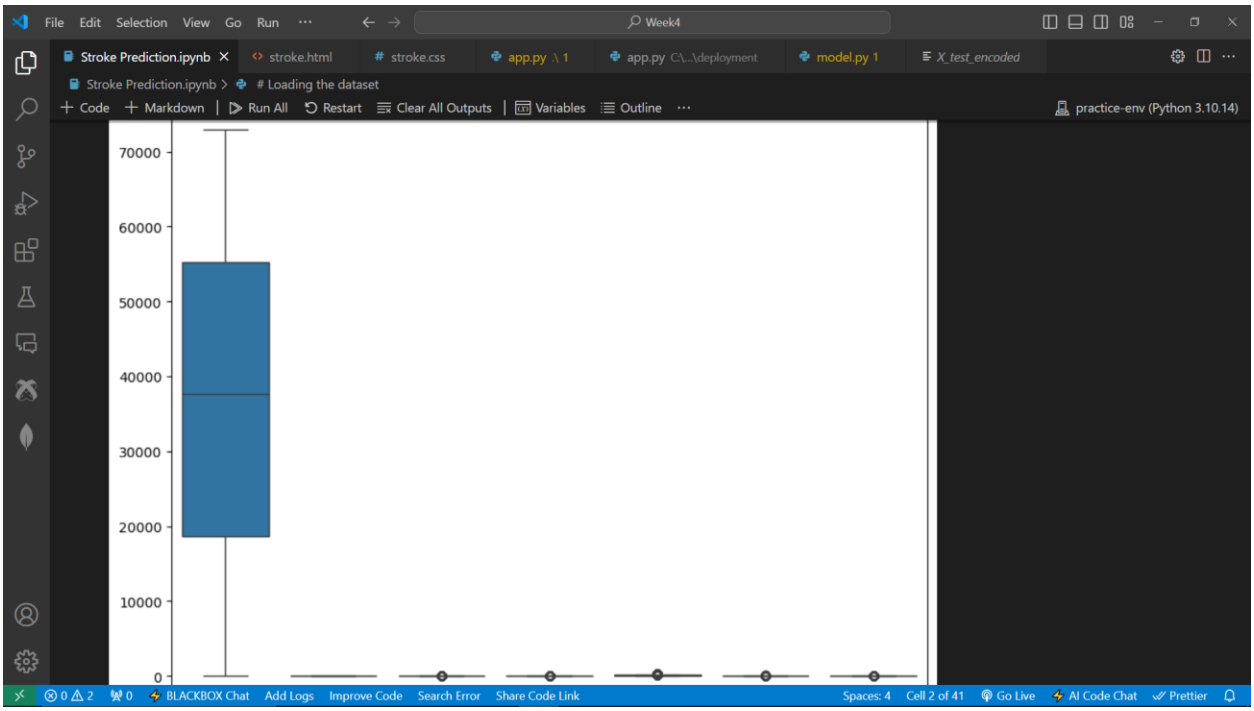
2. Data Preparation

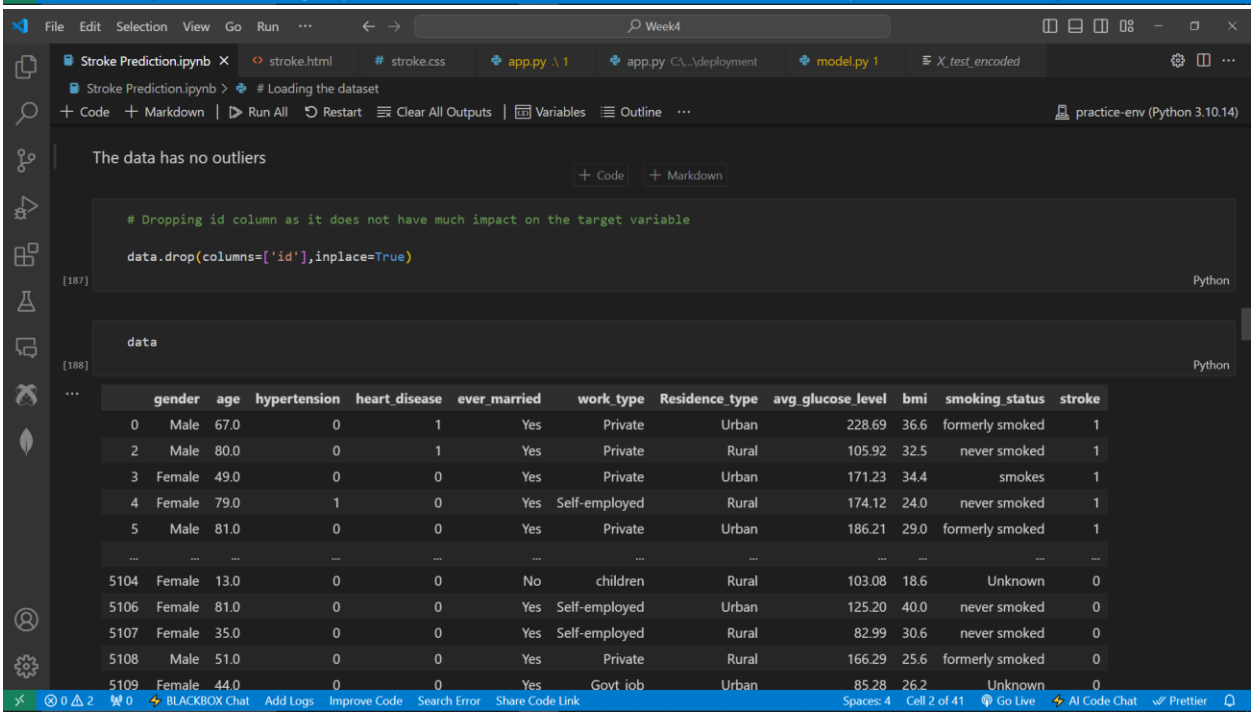

File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py
app.py C:\...\deployment
model.py
X_test_encoded
Stroke Prediction.ipynb > # Loading the dataset
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline
practice-env (Python 3.10.14)
data.info()
[181]
Python
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5110 entries, 0 to 5109
Data columns (total 12 columns):
Column Non-Null Count Dtype
--- ------ -------------- -----
0 id 5110 non-null int64
1 gender 5110 non-null object
2 age 5110 non-null float64
3 hypertension 5110 non-null int64
4 heart_disease 5110 non-null int64
5 ever_married 5110 non-null object
6 work_type 5110 non-null object
7 Residence_type 5110 non-null object
8 avg_glucose_level 5110 non-null float64
9 bmi 4909 non-null float64
10 smoking_status 5110 non-null object
11 stroke 5110 non-null int64
dtypes: float64(3), int64(4), object(5)
memory usage: 479.2+ KB
From the above information, we can see that the column bmi has 4909 entries out of the 5110 which may mean that it has missing values.Let's confirm if it does
BLACKBOX Chat Add Logs Improve Code Search Error Share Code Link
Spaces: 4 Cell 2 of 41 Go Live AI Code Chat Prettier


File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py
app.py C:\...\deployment
model.py
X_test_encoded
Stroke Prediction.ipynb > # Loading the dataset
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline
practice-env (Python 3.10.14)
From the above information, we can see that the column bmi has 4909 entries out of the 5110 which may mean that it has missing values.Let's confirm if it does
data.isna().sum()
[182]
Python
id 0
gender 0
age 0
hypertension 0
heart_disease 0
ever_married 0
work_type 0
Residence_type 0
avg_glucose_level 0
bmi 201
smoking_status 0
stroke 0
dtype: int64
There are 201 missing entries in the bmi.This is close to 4% of the values in that column that are missing.This is quite a small percentage of missing values so we can drop them
data.dropna(subset=['bmi'],inplace=True)
BLACKBOX Chat Add Logs Improve Code Search Error Share Code Link
Spaces: 4 Cell 2 of 41 Go Live AI Code Chat Prettier

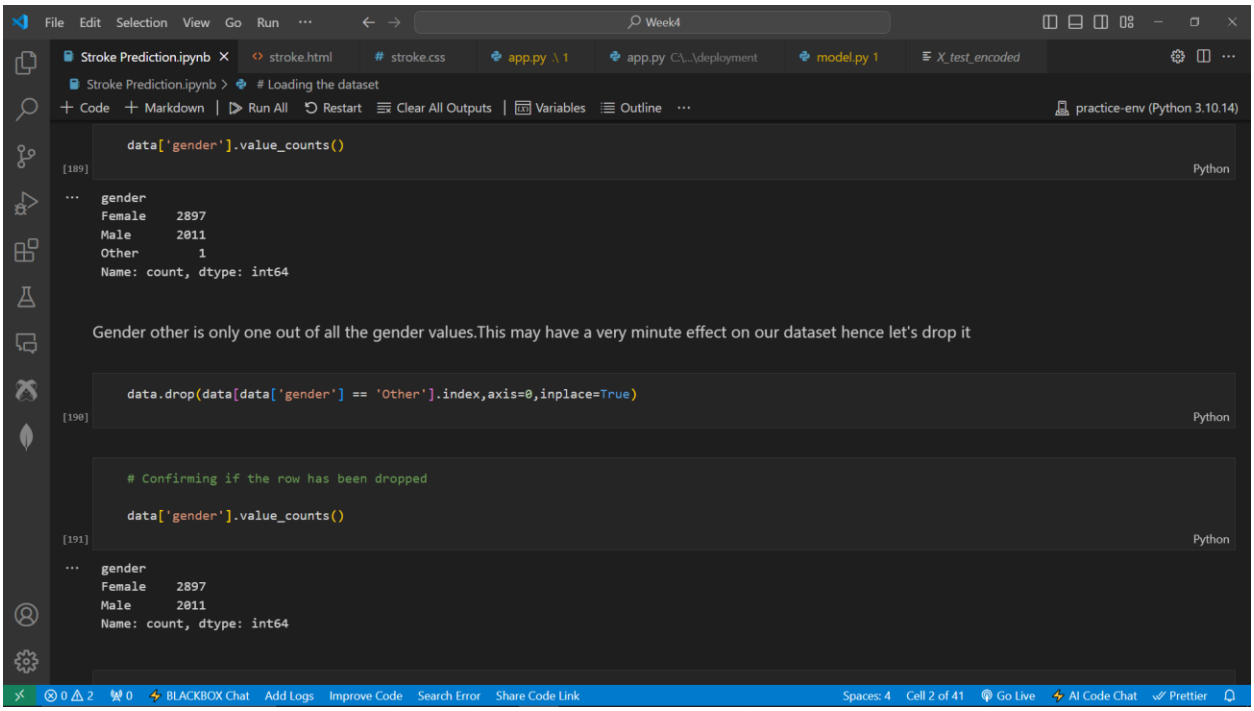

File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py
app.py C:\...\deployment
model.py
X_test_encoded
Stroke Prediction.ipynb > # Loading the dataset
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline
practice-env (Python 3.10.14)
There are 201 missing entries in the bmi.This is close to 4% of the values in that column that are missing.This is quite a small percentage of missing values so we can drop them
data.dropna(subset=['bmi'],inplace=True)
[183]
Python
data.isna().sum()
[184]
Python
id 0
gender 0
age 0
hypertension 0
heart_disease 0
ever_married 0
work_type 0
Residence_type 0
avg_glucose_level 0
bmi 0
smoking_status 0
stroke 0
dtype: int64
Now there are no missing values
BLACKBOX Chat Add Logs Improve Code Search Error Share Code Link
Spaces: 4 Cell 2 of 41 Go Live AI Code Chat Prettier

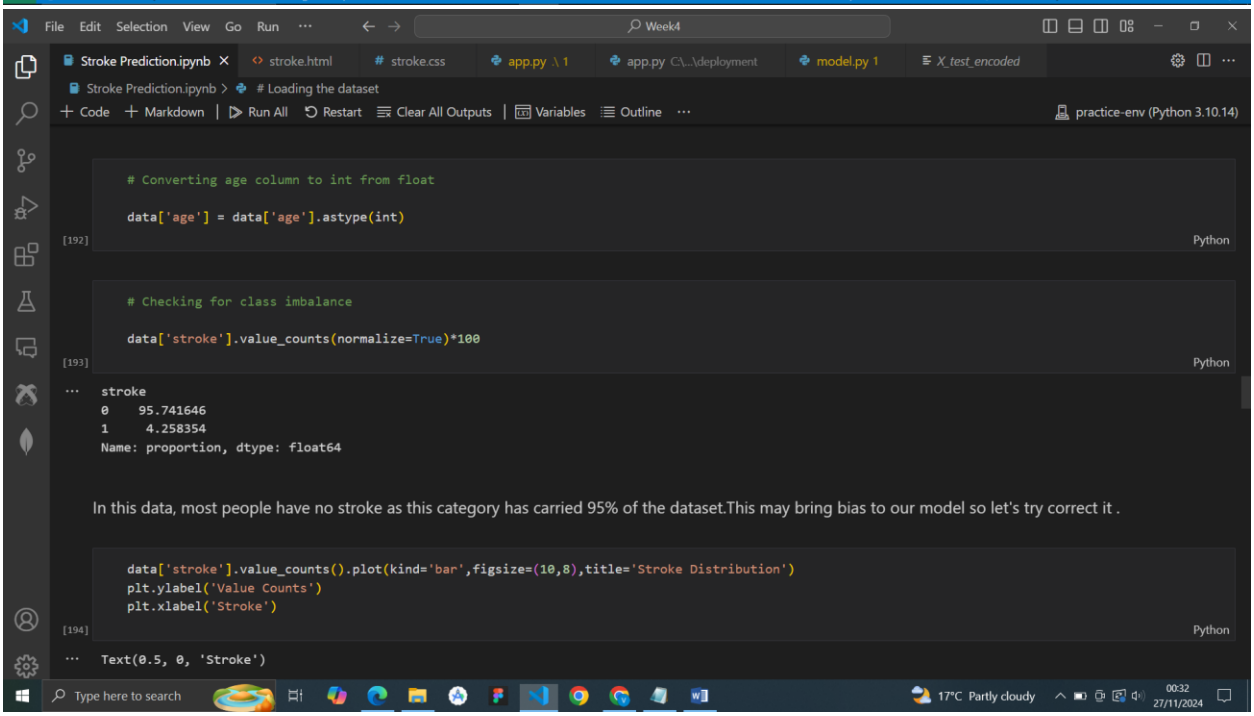

File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py
app.py C:\...\deployment
model.py
X_test_encoded
Stroke Prediction.ipynb > # Loading the dataset
+ Code + Markdown | Run All Restart Clear All Outputs | Variables Outline
practice-env (Python 3.10.14)
Now there are no missing values
Checking for duplicates
data.duplicated().sum()
[185]
Python
0
There are no duplicated values in the dataset
# Checking for outliers
plt.figure(figsize=(10,8))
sns.boxplot(data)
[186]
Python
<Axes: >
70000
BLACKBOX Chat Add Logs Improve Code Search Error Share Code Link
Spaces: 4 Cell 2 of 41 Go Live AI Code Chat Prettier

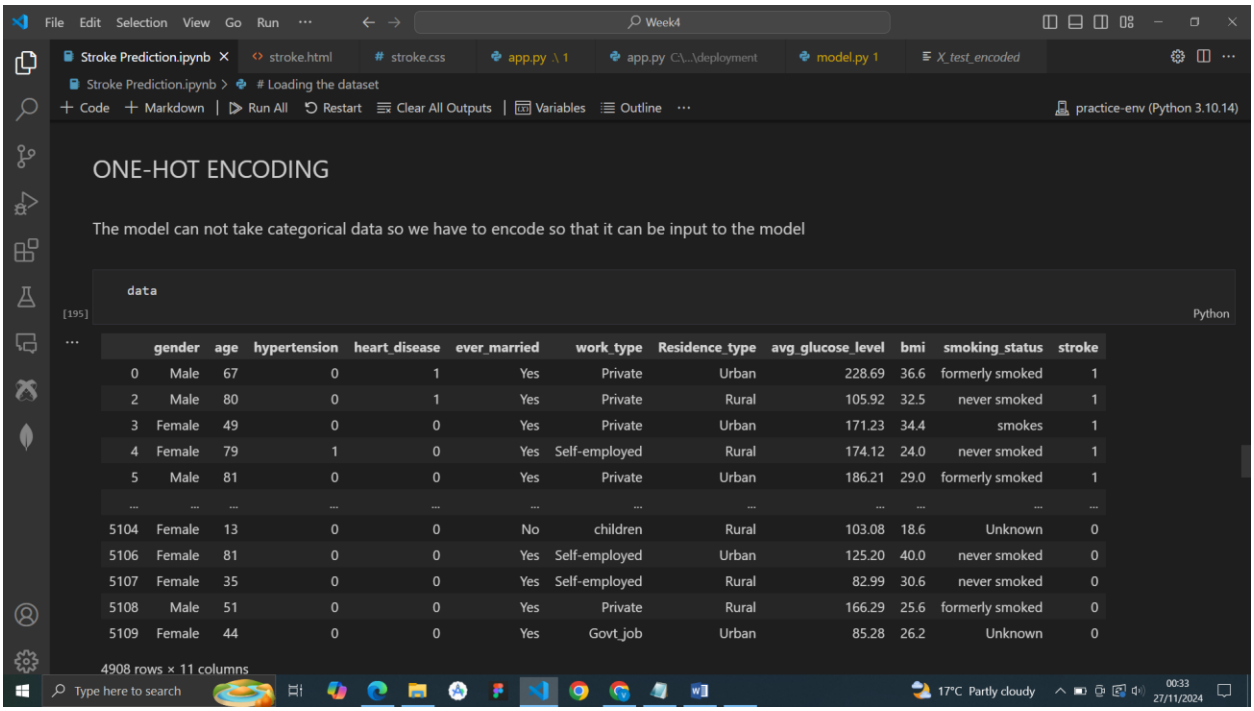


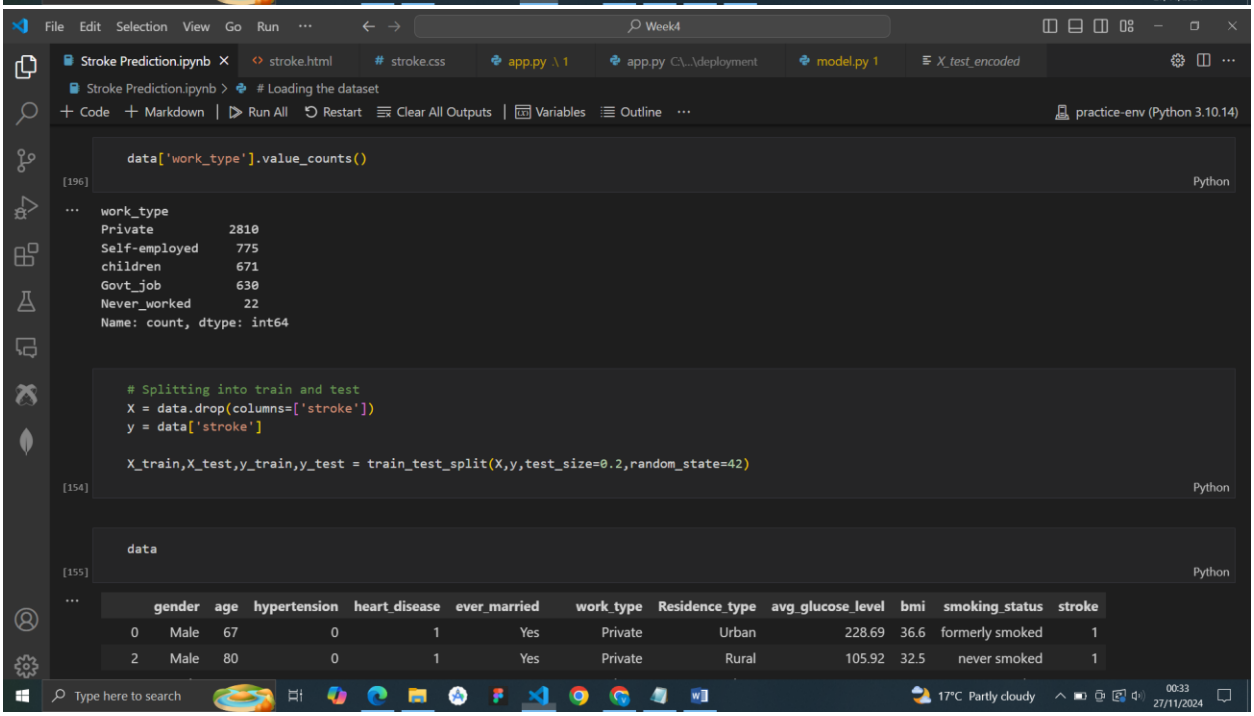


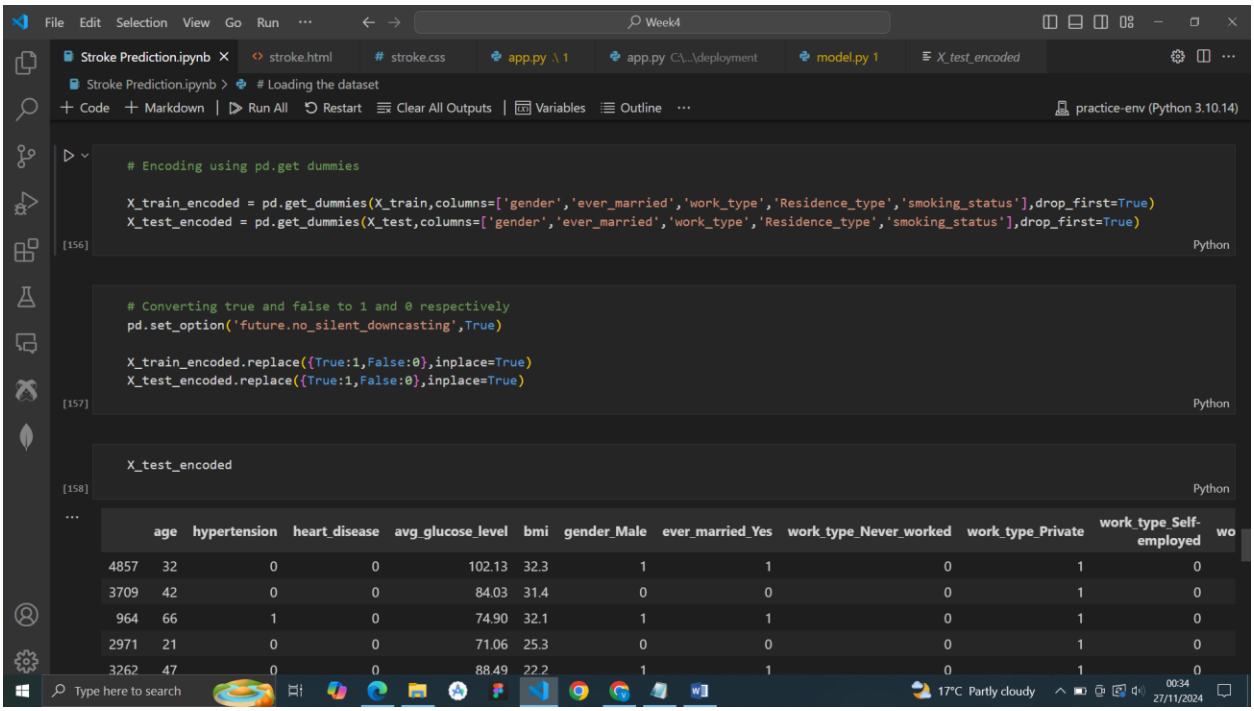
	gender	age	hypertension	heart_disease	ever_married	work_type	Residence_type	avg_glucose_level	bmi	smoking_status	stroke
0	Male	67.0	0	1	Yes	Private	Urban	228.69	36.6	formerly smoked	1
2	Male	80.0	0	1	Yes	Private	Rural	105.92	32.5	never smoked	1
3	Female	49.0	0	0	Yes	Private	Urban	171.23	34.4	smokes	1
4	Female	79.0	1	0	Yes	Self-employed	Rural	174.12	24.0	never smoked	1
5	Male	81.0	0	0	Yes	Private	Urban	186.21	29.0	formerly smoked	1
...	...	...	...	...	...	...	...	...	...	...	...
5104	Female	13.0	0	0	No	children	Rural	103.08	18.6	Unknown	0
5106	Female	81.0	0	0	Yes	Self-employed	Urban	125.20	40.0	never smoked	0
5107	Female	35.0	0	0	Yes	Self-employed	Rural	82.99	30.6	never smoked	0
5108	Male	51.0	0	0	Yes	Private	Rural	166.29	25.6	formerly smoked	0
5109	Female	44.0	0	0	Yes	Govt_job	Urban	85.28	26.2	Unknown	0


File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py
app.py C:\...\deployment
model.py
X_test_encoded
Stroke Prediction.ipynb > # Loading the dataset
Code Markdown Run All Restart Clear All Outputs Variables Outline
practice-env (Python 3.10.14)
data['gender'].value_counts()
[189]
Python
gender
Female 2897
Male 2011
Other 1
Name: count, dtype: int64
Gender other is only one out of all the gender values.This may have a very minute effect on our dataset hence let's drop it
data.drop(data[data['gender'] == 'Other'].index,axis=0,inplace=True)
[190]
Python
Confirming if the row has been dropped
data['gender'].value_counts()
[191]
Python
gender
Female 2897
Male 2011
Name: count, dtype: int64
BLACKBOX Chat Add Logs Improve Code Search Error Share Code Link
Spaces: 4 Cell 2 of 41 Go Live AI Code Chat Prettier


File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py
app.py C:\...\deployment
model.py
X_test_encoded
Stroke Prediction.ipynb > # Loading the dataset
Code Markdown Run All Restart Clear All Outputs Variables Outline
practice-env (Python 3.10.14)
Converting age column to int from float
data['age'] = data['age'].astype(int)
[192]
Python
Checking for class imbalance
data['stroke'].value_counts(normalize=True)*100
[193]
Python
stroke
0 95.741646
1 4.258354
Name: proportion, dtype: float64
In this data, most people have no stroke as this category has carried 95% of the dataset.This may bring bias to our model so let's try correct it .
data['stroke'].value_counts().plot(kind='bar',figsize=(10,8),title='Stroke Distribution')
plt.ylabel('Value Counts')
plt.xlabel('Stroke')
[194]
Python
Text(0.5, 0, 'Stroke')
Type here to search
17°C Partly cloudy
00:32
27/11/2024


File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py
app.py C:\...\deployment
model.py
X_test_encoded
Stroke Prediction.ipynb > # Loading the dataset
Code Markdown Run All Restart Clear All Outputs Variables Outline
practice-env (Python 3.10.14)
ONE-HOT ENCODING
The model can not take categorical data so we have to encode so that it can be input to the model
[195]
data
Python
gender age hypertension heart_disease ever_married work_type Residence_type avg_glucose_level bmi smoking_status stroke
0 Male 67 0 1 Yes Private Urban 228.69 36.6 formerly smoked 1
2 Male 80 0 1 Yes Private Rural 105.92 32.5 never smoked 1
3 Female 49 0 0 Yes Private Urban 171.23 34.4 smokes 1
4 Female 79 1 0 Yes Self-employed Rural 174.12 24.0 never smoked 1
5 Male 81 0 0 Yes Private Urban 186.21 29.0 formerly smoked 1
5104 Female 13 0 0 No children Rural 103.08 18.6 Unknown 0
5106 Female 81 0 0 Yes Self-employed Urban 125.20 40.0 never smoked 0
5107 Female 35 0 0 Yes Self-employed Rural 82.99 30.6 never smoked 0
5108 Male 51 0 0 Yes Private Rural 166.29 25.6 formerly smoked 0
5109 Female 44 0 0 Yes Govt_job Urban 85.28 26.2 Unknown 0
4908 rows × 11 columns
Type here to search
17°C Partly cloudy
00:33
27/11/2024


File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py
app.py C:\...\deployment
model.py
X_test_encoded
Stroke Prediction.ipynb > # Loading the dataset
Code Markdown Run All Restart Clear All Outputs Variables Outline
practice-env (Python 3.10.14)
[196]
data['work_type'].value_counts()
Python
work_type
Private 2810
Self-employed 775
children 671
Govt_job 630
Never_worked 22
Name: count, dtype: int64
[154]
# Splitting into train and test
X = data.drop(columns=['stroke'])
y = data['stroke']
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=42)
Python
[155]
data
Python
gender age hypertension heart_disease ever_married work_type Residence_type avg_glucose_level bmi smoking_status stroke
0 Male 67 0 1 Yes Private Urban 228.69 36.6 formerly smoked 1
2 Male 80 0 1 Yes Private Rural 105.92 32.5 never smoked 1
Type here to search
17°C Partly cloudy
00:33
27/11/2024



```
# Encoding using pd.get dummies

X_train_encoded = pd.get_dummies(X_train,columns=['gender','ever_married','work_type','Residence_type','smoking_status'],drop_first=True)
X_test_encoded = pd.get_dummies(X_test,columns=['gender','ever_married','work_type','Residence_type','smoking_status'],drop_first=True)
```

[156] Python

```
# Converting true and false to 1 and 0 respectively
pd.set_option('future.no_silent_downcasting',True)

X_train_encoded.replace({True:1,False:0},inplace=True)
X_test_encoded.replace({True:1,False:0},inplace=True)
```

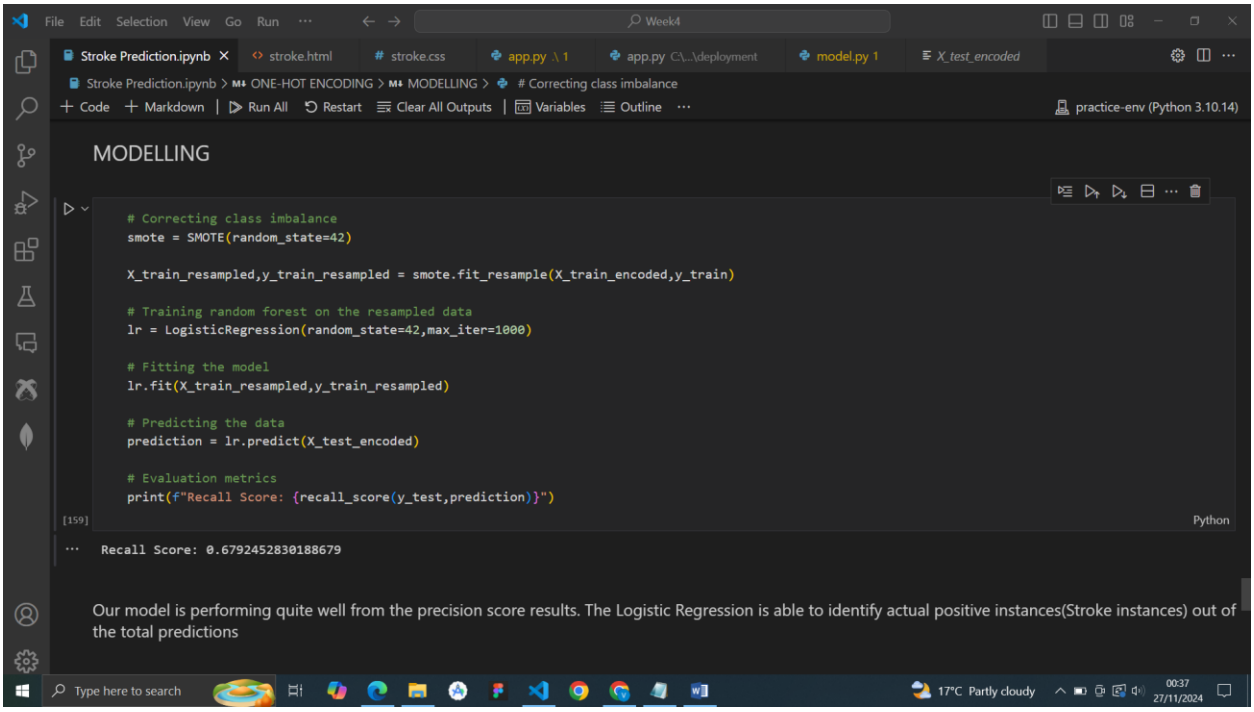
[157] Python

```
X_test_encoded
```

[158] Python

	age	hypertension	heart_disease	avg_glucose_level	bmi	gender_Male	ever_married_Yes	work_type_Never_worked	work_type_Private	work_type_Self-employed	wo
4857	32	0	0	102.13	32.3	1	1	0	1	0	
3709	42	0	0	84.03	31.4	0	0	0	1	0	
964	66	1	0	74.90	32.1	1	1	0	1	0	
2971	21	0	0	71.06	25.3	0	0	0	1	0	
3262	47	0	0	88.49	22.2	1	1	0	1	0	

3. Modelling



MODELLING

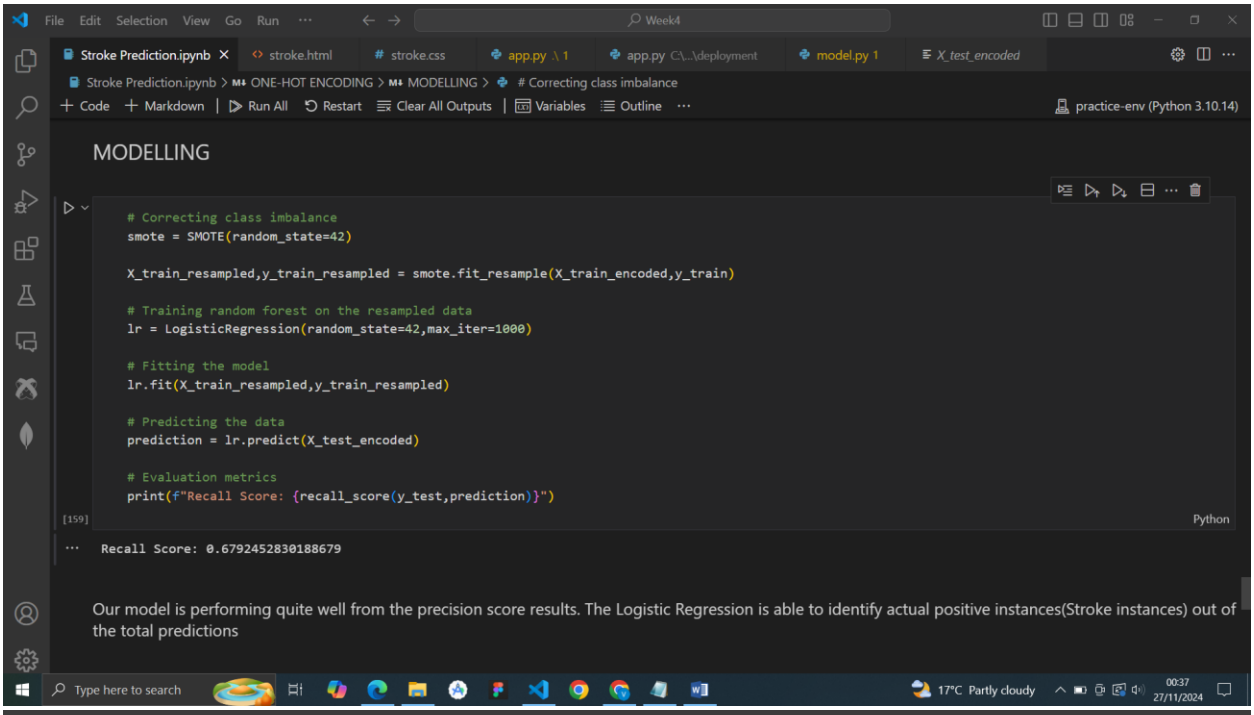
```
# Correcting class imbalance
smote = SMOTE(random_state=42)

X_train_resampled,y_train_resampled = smote.fit_resample(X_train_encoded,y_train)

# Training random forest on the resampled data
lr = LogisticRegression(random_state=42,max_iter=1000)

# Fitting the model
lr.fit(X_train_resampled,y_train_resampled)

# Predicting the data
prediction = lr.predict(X_test_encoded)

# Evaluation metrics
print(f"Recall Score: {recall_score(y_test,prediction)}")
```

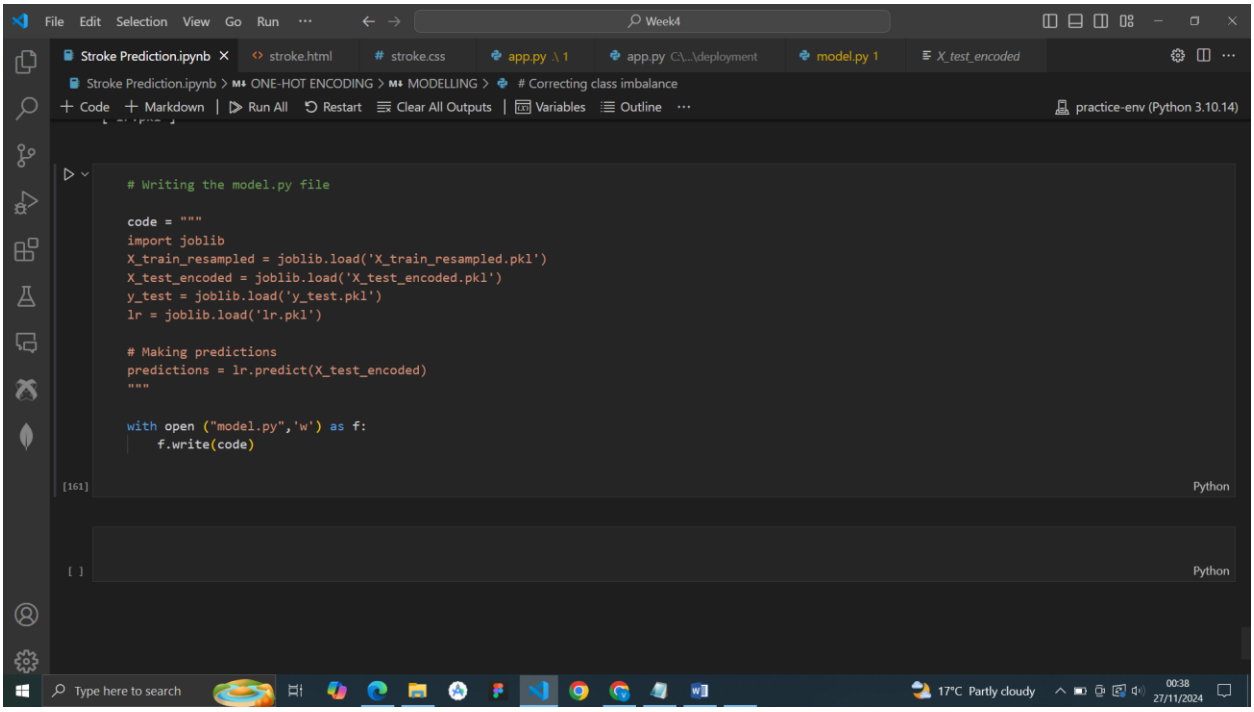
[159] Python

```
Recall Score: 0.6792452830188679
```

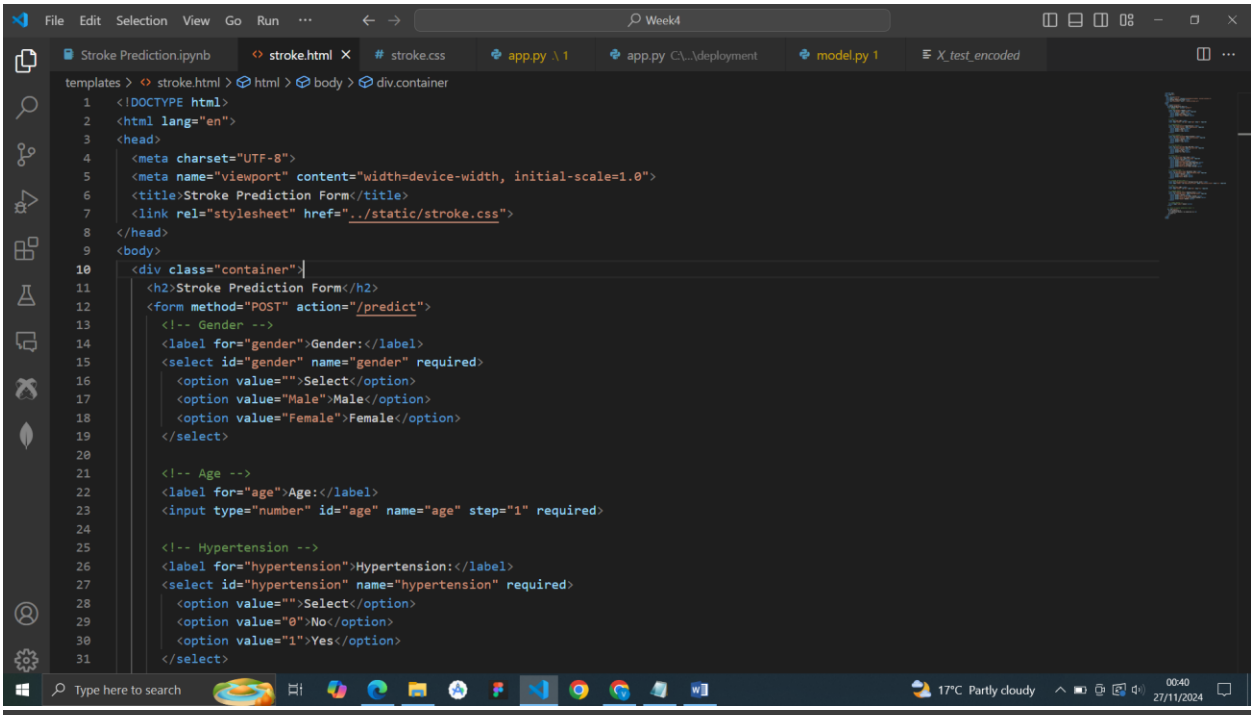
Our model is performing quite well from the precision score results. The Logistic Regression is able to identify actual positive instances(Stroke instances) out of the total predictions

4. Deployment



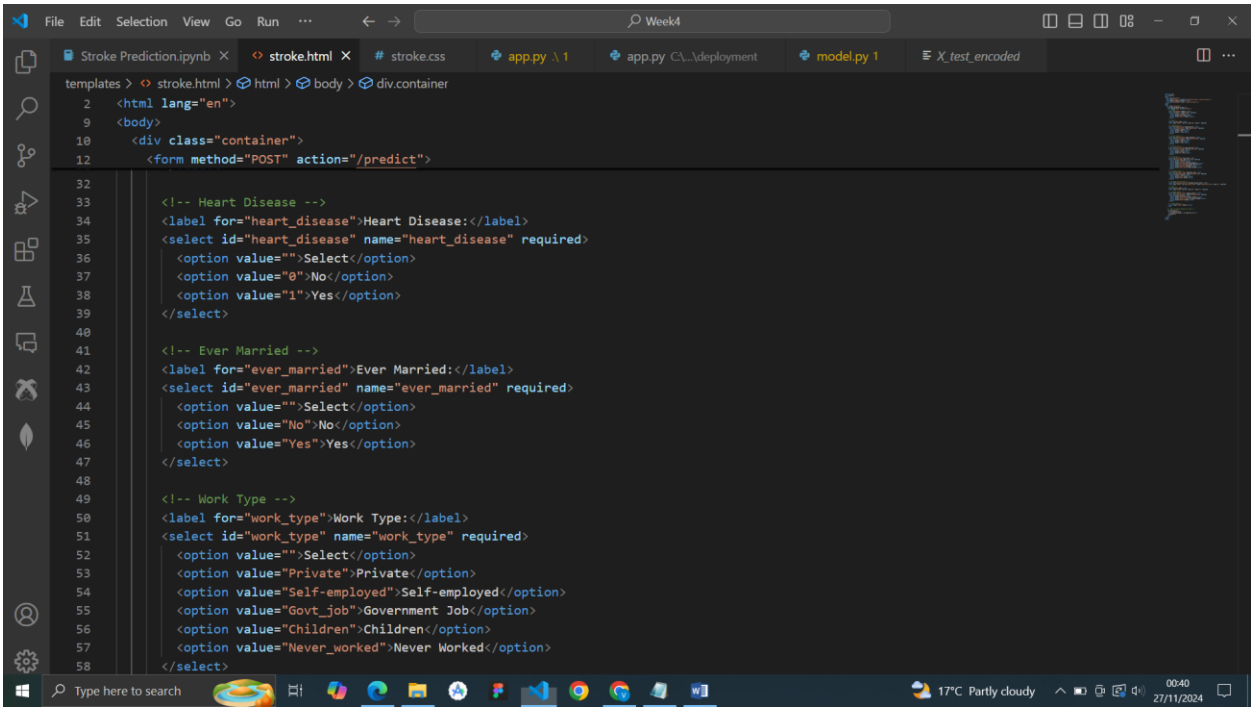


i) Html File



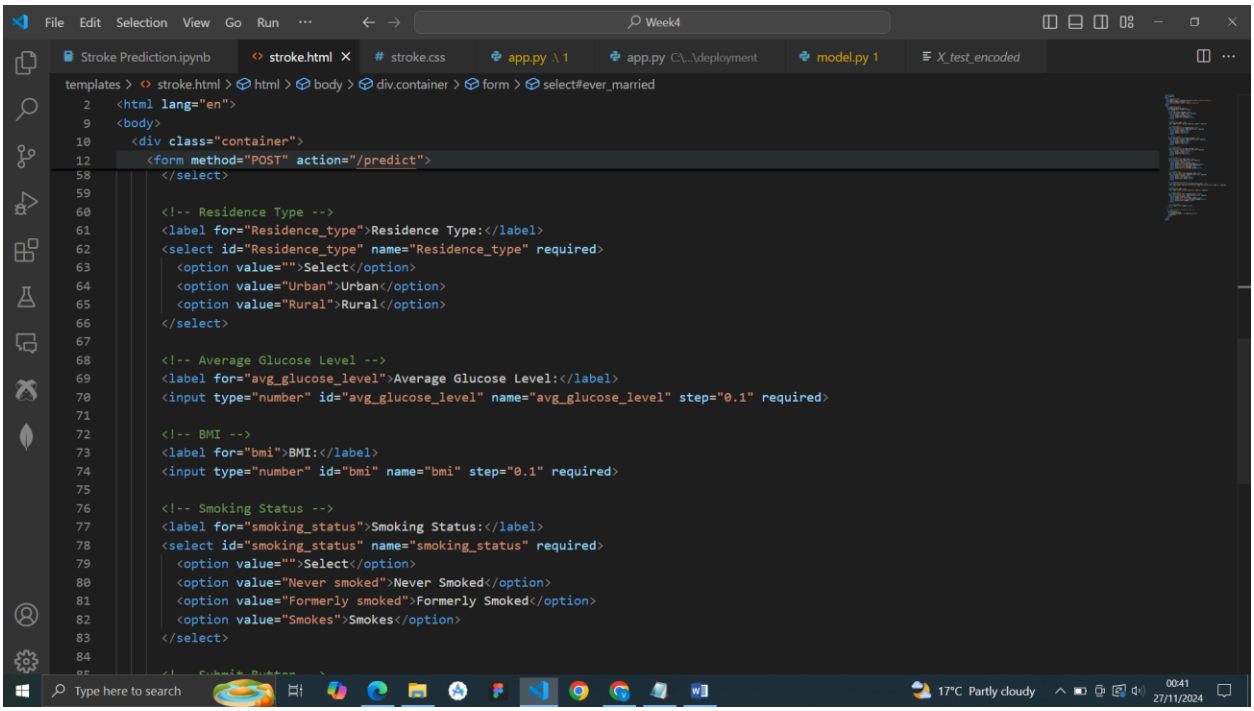
templates > stroke.html > html > body > div.container

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Stroke Prediction Form</title>
  <link rel="stylesheet" href="../static/stroke.css">
</head>
<body>
  <div class="container">
    <h2>Stroke Prediction Form</h2>
    <form method="POST" action="/predict">
      <!-- Gender -->
      <label for="gender">Gender:</label>
      <select id="gender" name="gender" required>
        <option value="">Select</option>
        <option value="Male">Male</option>
        <option value="Female">Female</option>
      </select>

      <!-- Age -->
      <label for="age">Age:</label>
      <input type="number" id="age" name="age" step="1" required>

      <!-- Hypertension -->
      <label for="hypertension">Hypertension:</label>
      <select id="hypertension" name="hypertension" required>
        <option value="">Select</option>
        <option value="0">No</option>
        <option value="1">Yes</option>
      </select>
```

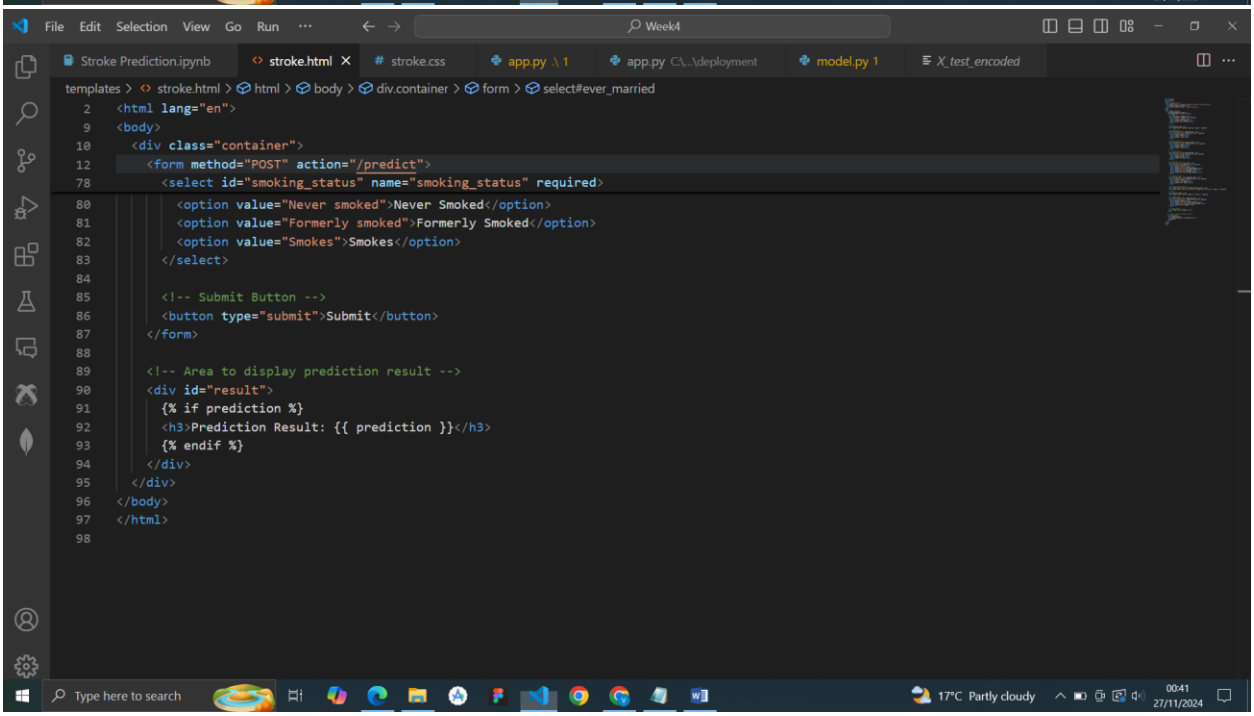


templates > stroke.html > html > body > div.container

```
<html lang="en">
<body>
  <div class="container">
    <form method="POST" action="/predict">

      <!-- Heart Disease -->
      <label for="heart_disease">Heart Disease:</label>
      <select id="heart_disease" name="heart_disease" required>
        <option value="">Select</option>
        <option value="0">No</option>
        <option value="1">Yes</option>
      </select>

      <!-- Ever Married -->
      <label for="ever_married">Ever Married:</label>
      <select id="ever_married" name="ever_married" required>
        <option value="">Select</option>
        <option value="No">No</option>
        <option value="Yes">Yes</option>
      </select>

      <!-- Work Type -->
      <label for="work_type">Work Type:</label>
      <select id="work_type" name="work_type" required>
        <option value="">Select</option>
        <option value="Private">Private</option>
        <option value="Self-employed">Self-employed</option>
        <option value="Govt_job">Government Job</option>
        <option value="Children">Children</option>
        <option value="Never_worked">Never Worked</option>
      </select>
```


File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py \ 1
app.py C:\...\deployment
model.py 1
X_test_encoded
templates > stroke.html > html > body > div.container > form > select#ever_married
<html lang="en">
<body>

<form method="POST" action="/predict">
</select>
<!-- Residence Type -->
<label for="Residence_type">Residence Type:</label>
<select id="Residence_type" name="Residence_type" required>
<option value="">Select</option>
<option value="Urban">Urban</option>
<option value="Rural">Rural</option>
</select>
<!-- Average Glucose Level -->
<label for="avg_glucose_level">Average Glucose Level:</label>
<input type="number" id="avg_glucose_level" name="avg_glucose_level" step="0.1" required>
<!-- BMI -->
<label for="bmi">BMI:</label>
<input type="number" id="bmi" name="bmi" step="0.1" required>
<!-- Smoking Status -->
<label for="smoking_status">Smoking Status:</label>
<select id="smoking_status" name="smoking_status" required>
<option value="">Select</option>
<option value="Never smoked">Never Smoked</option>
<option value="Formerly smoked">Formerly Smoked</option>
<option value="Smokes">Smokes</option>
</select>
Type here to search
17°C Partly cloudy
00:41
27/11/2024

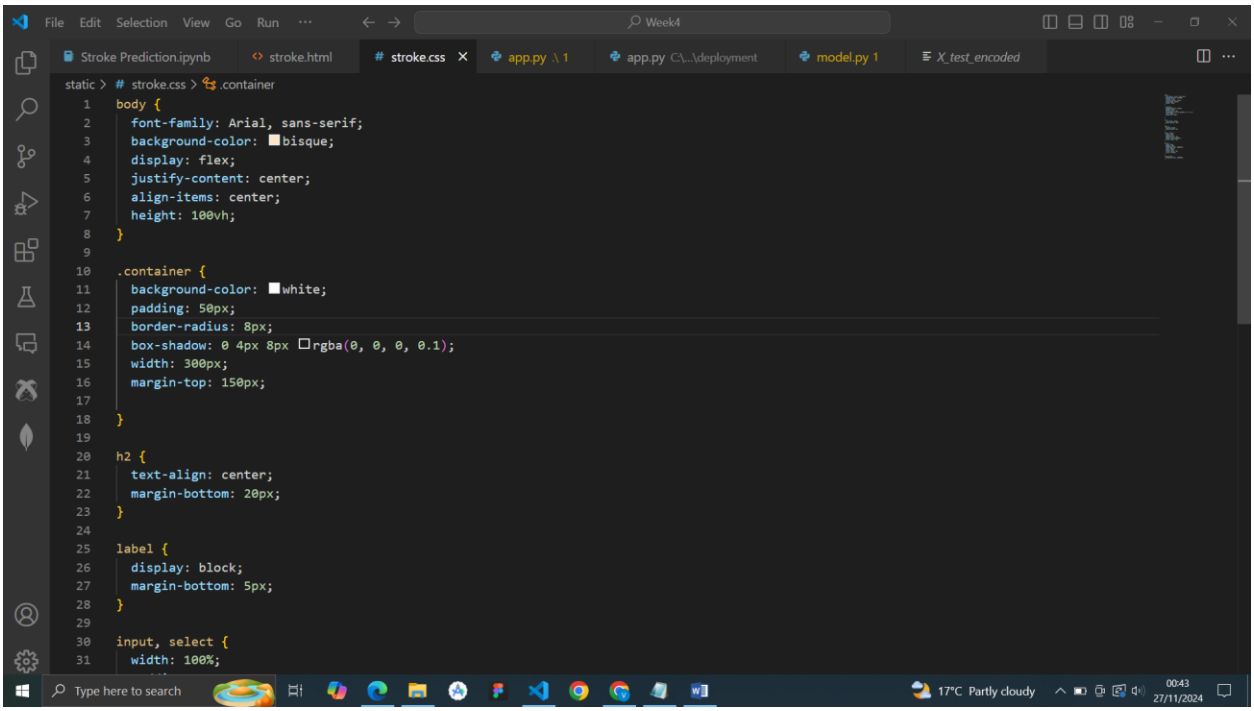

File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
app.py \ 1
app.py C:\...\deployment
model.py 1
X_test_encoded
templates > stroke.html > html > body > div.container > form > select#smoking_status
<html lang="en">
<body>

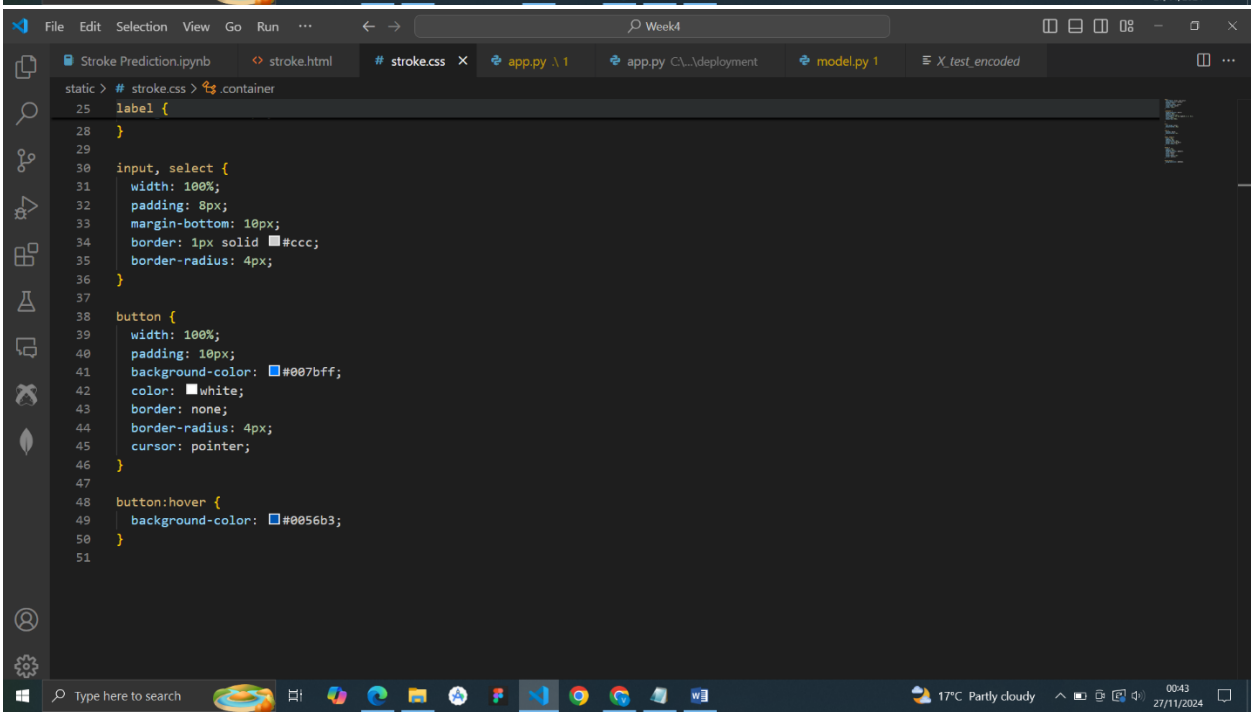
<form method="POST" action="/predict">
<select id="smoking_status" name="smoking_status" required>
<option value="Never smoked">Never Smoked</option>
<option value="Formerly smoked">Formerly Smoked</option>
<option value="Smokes">Smokes</option>
</select>
<!-- Submit Button -->
<button type="submit">Submit</button>
</form>
<!-- Area to display prediction result -->

{% if prediction %}
<h3>Prediction Result: {{ prediction }}</h3>
{% endif %}

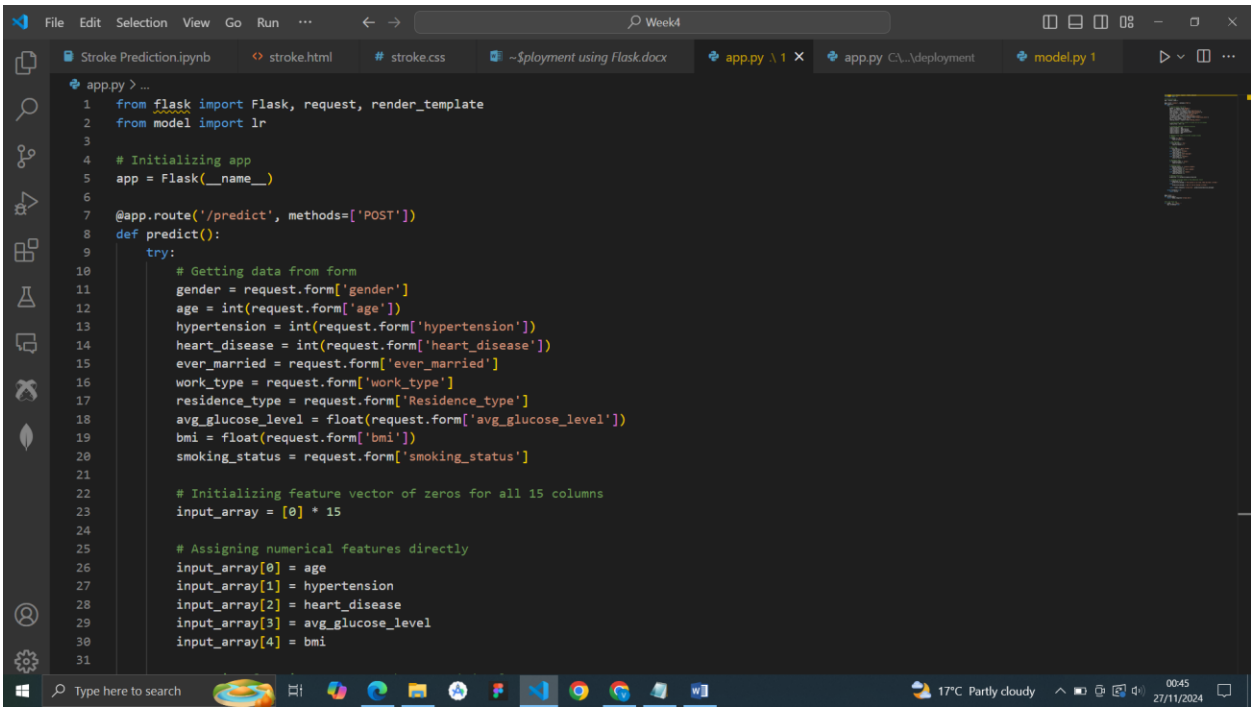
</body>
</html>
Type here to search
17°C Partly cloudy
00:41
27/11/2024

ii) Css File

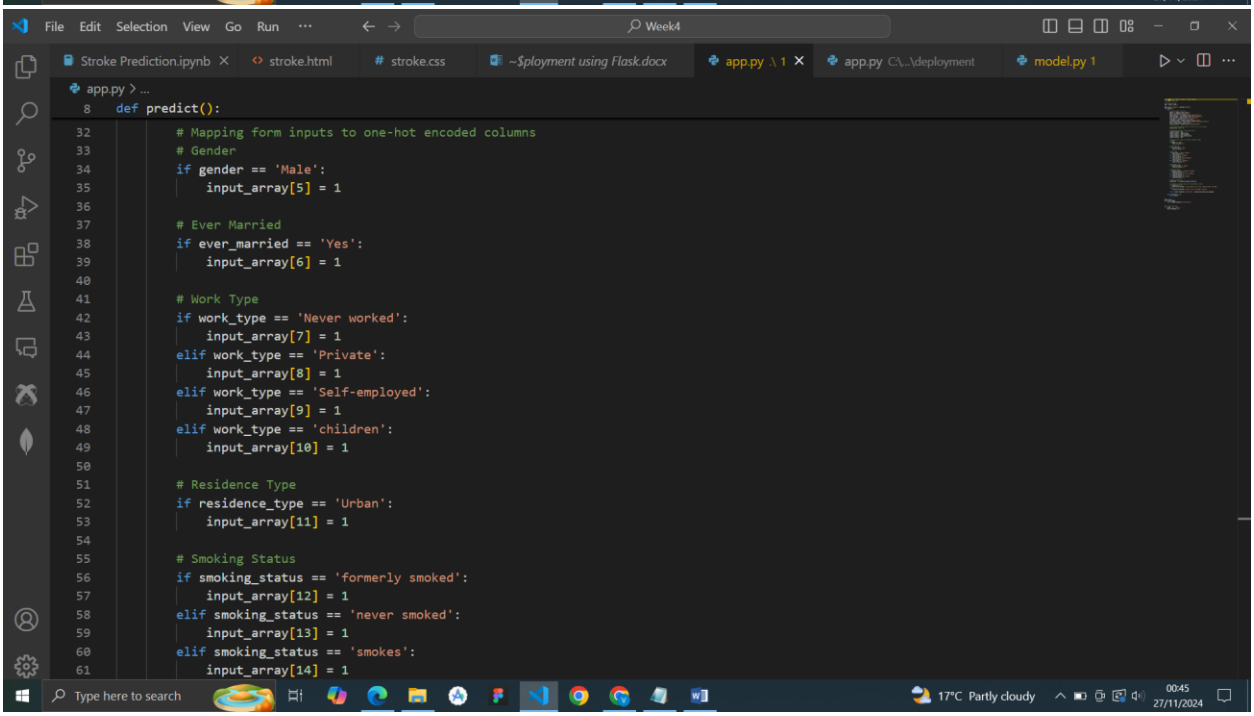




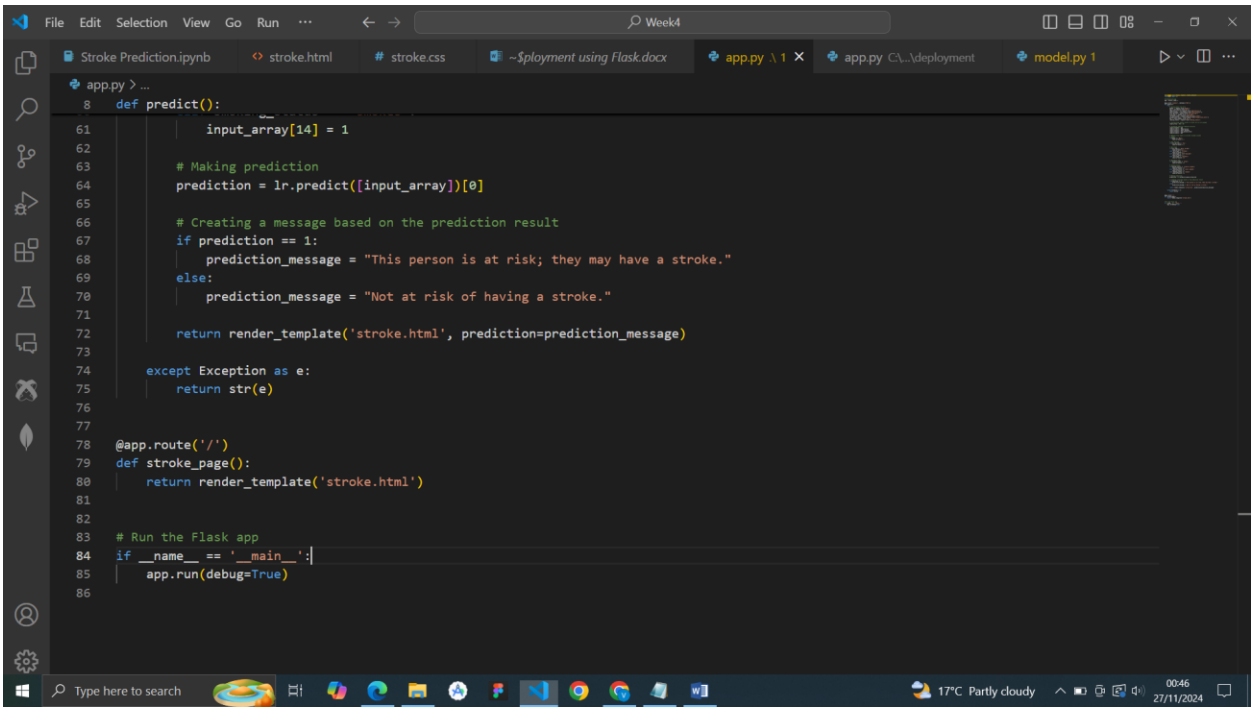
iii) App.py File (Flask)



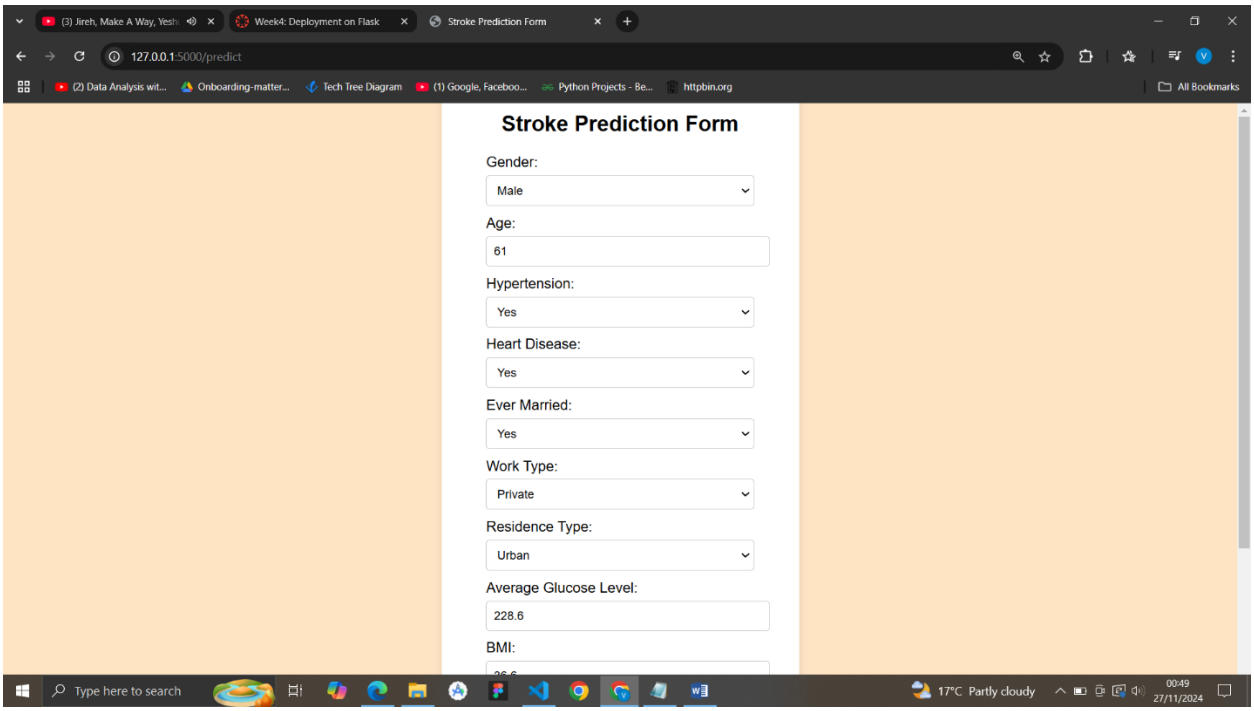
```python
from flask import Flask, request, render_template
from model import lr

# Initializing app
app = Flask(__name__)

@app.route('/predict', methods=['POST'])
def predict():
    try:
        # Getting data from form
        gender = request.form['gender']
        age = int(request.form['age'])
        hypertension = int(request.form['hypertension'])
        heart_disease = int(request.form['heart_disease'])
        ever_married = request.form['ever_married']
        work_type = request.form['work_type']
        residence_type = request.form['Residence_type']
        avg_glucose_level = float(request.form['avg_glucose_level'])
        bmi = float(request.form['bmi'])
        smoking_status = request.form['smoking_status']

        # Initializing feature vector of zeros for all 15 columns
        input_array = [0] * 15

        # Assigning numerical features directly
        input_array[0] = age
        input_array[1] = hypertension
        input_array[2] = heart_disease
        input_array[3] = avg_glucose_level
        input_array[4] = bmi
```

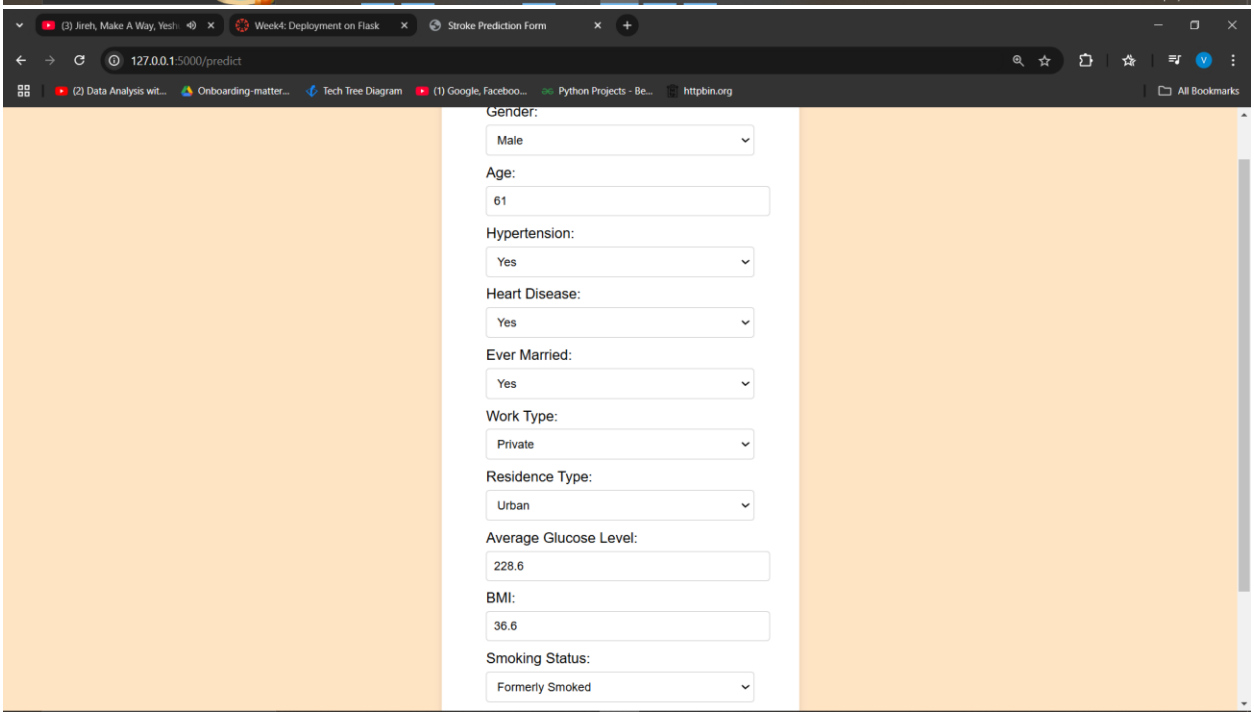


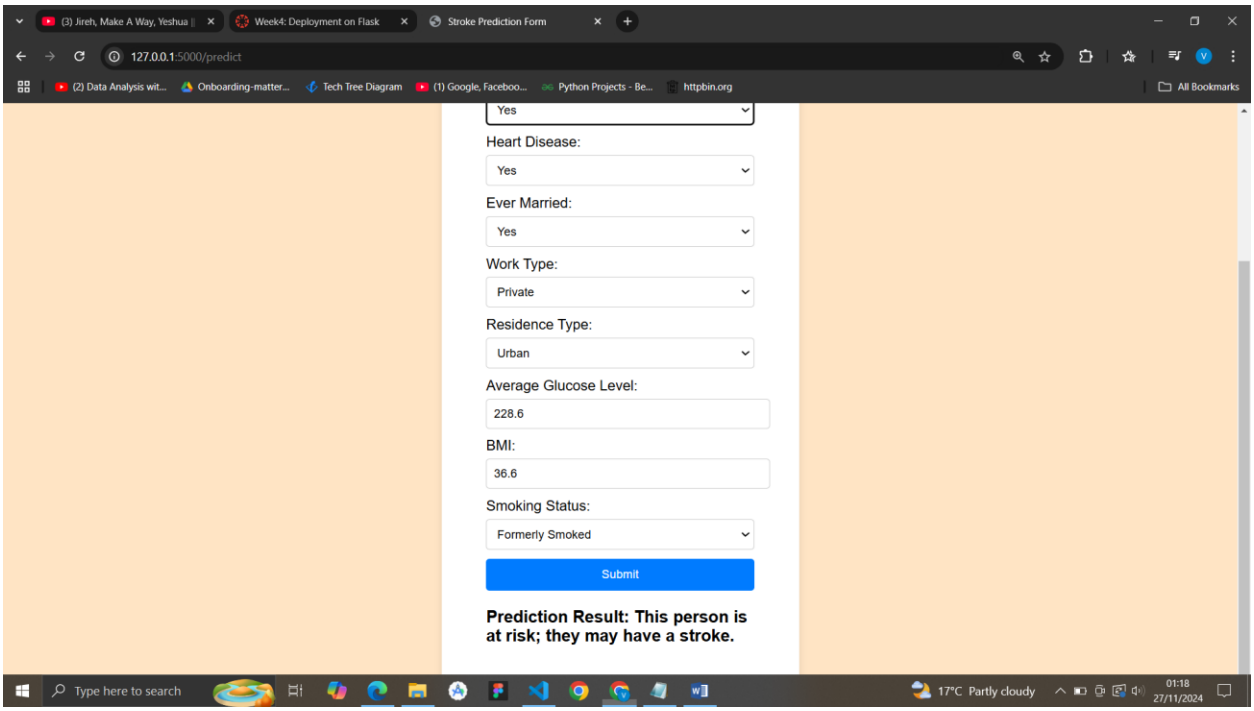
```python
def predict():
        # Mapping form inputs to one-hot encoded columns
        # Gender
        if gender == 'Male':
            input_array[5] = 1

        # Ever Married
        if ever_married == 'Yes':
            input_array[6] = 1

        # Work Type
        if work_type == 'Never worked':
            input_array[7] = 1
        elif work_type == 'Private':
            input_array[8] = 1
        elif work_type == 'Self-employed':
            input_array[9] = 1
        elif work_type == 'children':
            input_array[10] = 1

        # Residence Type
        if residence_type == 'Urban':
            input_array[11] = 1

        # Smoking Status
        if smoking_status == 'formerly smoked':
            input_array[12] = 1
        elif smoking_status == 'never smoked':
            input_array[13] = 1
        elif smoking_status == 'smokes':
            input_array[14] = 1
```


File Edit Selection View Go Run
Week4
Stroke Prediction.ipynb
stroke.html
stroke.css
~$ployment using Flask.docx
app.py
app.py C:\...\deployment
model.py 1
app.py > ...
def predict():
input_array[14] = 1
# Making prediction
prediction = lr.predict([input_array])[0]
# Creating a message based on the prediction result
if prediction == 1:
prediction_message = "This person is at risk; they may have a stroke."
else:
prediction_message = "Not at risk of having a stroke."
return render_template('stroke.html', prediction=prediction_message)
except Exception as e:
return str(e)
@app.route('/')
def stroke_page():
return render_template('stroke.html')
# Run the Flask app
if __name__ == '__main__':
app.run(debug=True)
Type here to search
17°C Partly cloudy
00:46
27/11/2024

OUTPUT


(3) Jireh, Make A Way, Yesh
Week4: Deployment on Flask
Stroke Prediction Form
127.0.0.1:5000/predict
(2) Data Analysis wit...
Onboarding-matter...
Tech Tree Diagram
(1) Google, Faceboo...
Python Projects - Be...
httpbin.org
All Bookmarks
Stroke Prediction Form
Gender:
Male
Age:
61
Hypertension:
Yes
Heart Disease:
Yes
Ever Married:
Yes
Work Type:
Private
Residence Type:
Urban
Average Glucose Level:
228.6
BMI:
Type here to search
17°C Partly cloudy
00:49
27/11/2024


(3) Jireh, Make A Way, Yesh
Week4: Deployment on Flask
Stroke Prediction Form
127.0.0.1:5000/predict
(2) Data Analysis wit...
Onboarding-matter...
Tech Tree Diagram
(1) Google, Faceboo...
Python Projects - Be...
httpbin.org
All Bookmarks
Gender:
Male
Age:
61
Hypertension:
Yes
Heart Disease:
Yes
Ever Married:
Yes
Work Type:
Private
Residence Type:
Urban
Average Glucose Level:
228.6
BMI:
36.6
Smoking Status:
Formerly Smoked


(3) Jireh, Make A Way, Yeshua
Week4: Deployment on Flask
Stroke Prediction Form
127.0.0.1:5000/predict
(2) Data Analysis wit...
Onboarding-matter...
Tech Tree Diagram
(1) Google, Faceboo...
Python Projects - Be...
httpbin.org
All Bookmarks
Yes
Heart Disease:
Yes
Ever Married:
Yes
Work Type:
Private
Residence Type:
Urban
Average Glucose Level:
228.6
BMI:
36.6
Smoking Status:
Formerly Smoked
Submit
Prediction Result: This person is at risk; they may have a stroke.
Type here to search
17°C Partly cloudy
01:18
27/11/2024