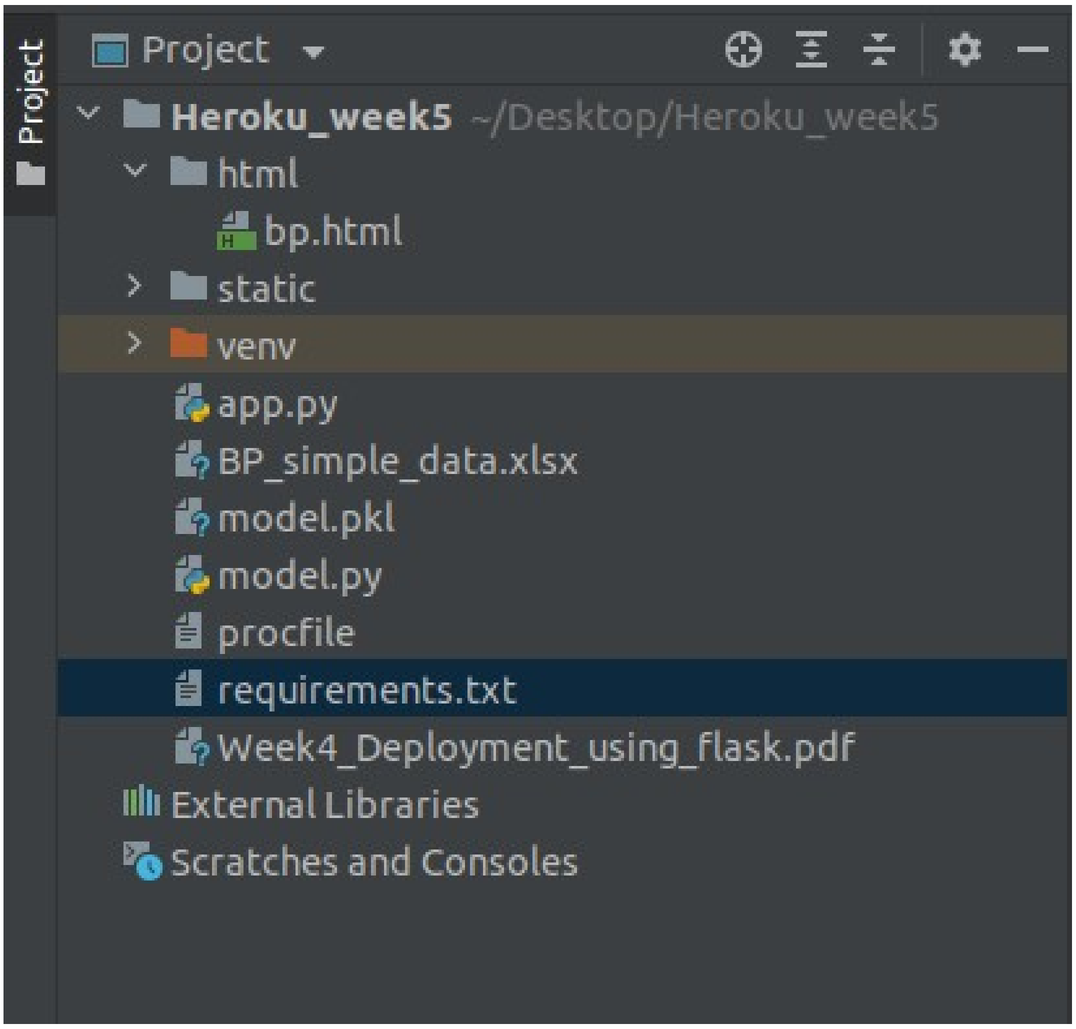
Week5: Cloud API Deployment

Name: Ilyas Nayle
Batch Code: LISUM03
Submission Date: 19/09/2021
Submitted to: Data Glacier

For the Week5- Cloud API Deployment the following steps were applied.

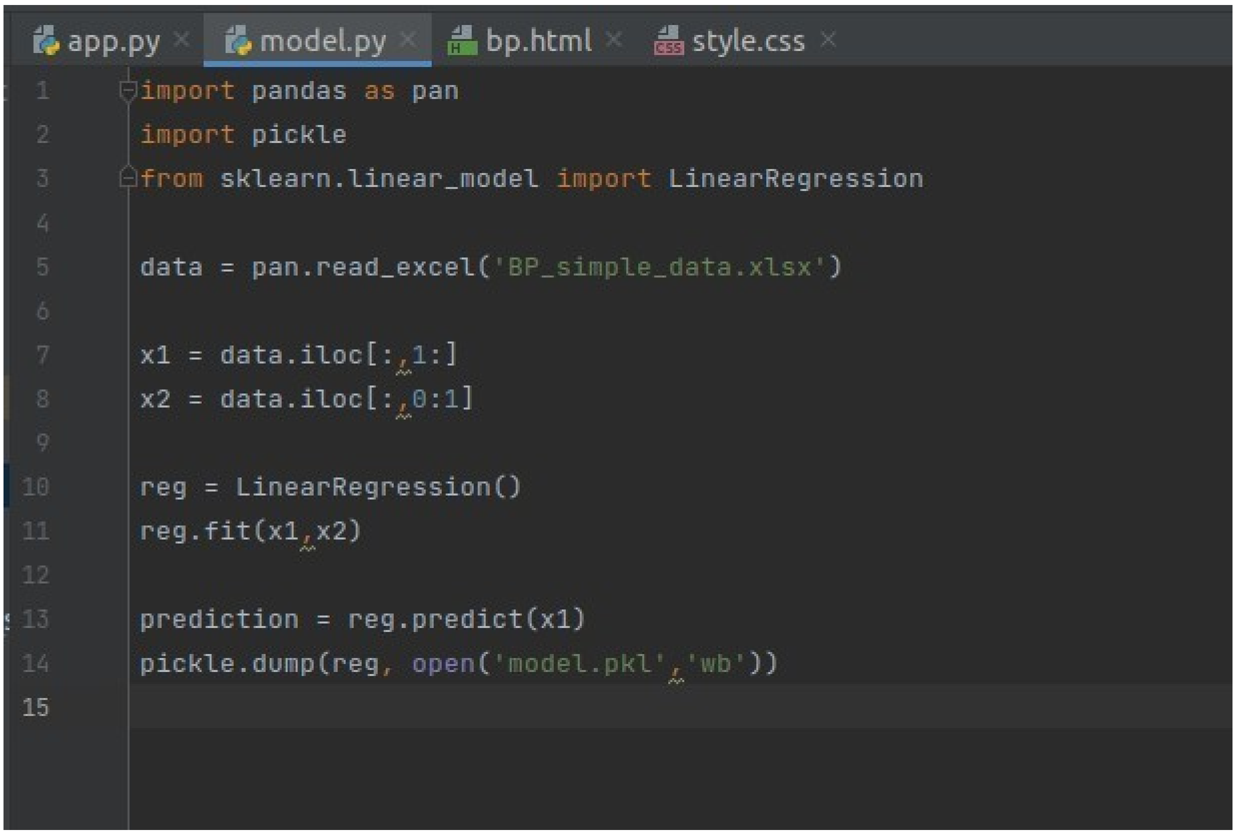
Directory:



```
Heroku_week5 ~/Desktop/Heroku_week5
  html
    bp.html
  static
  venv
  app.py
  BP_simple_data.xlsx
  model.pkl
  model.py
  procfile
  requirements.txt
  Week4_Deployment_using_flask.pdf
External Libraries
Scratches and Consoles
```

Step 1:

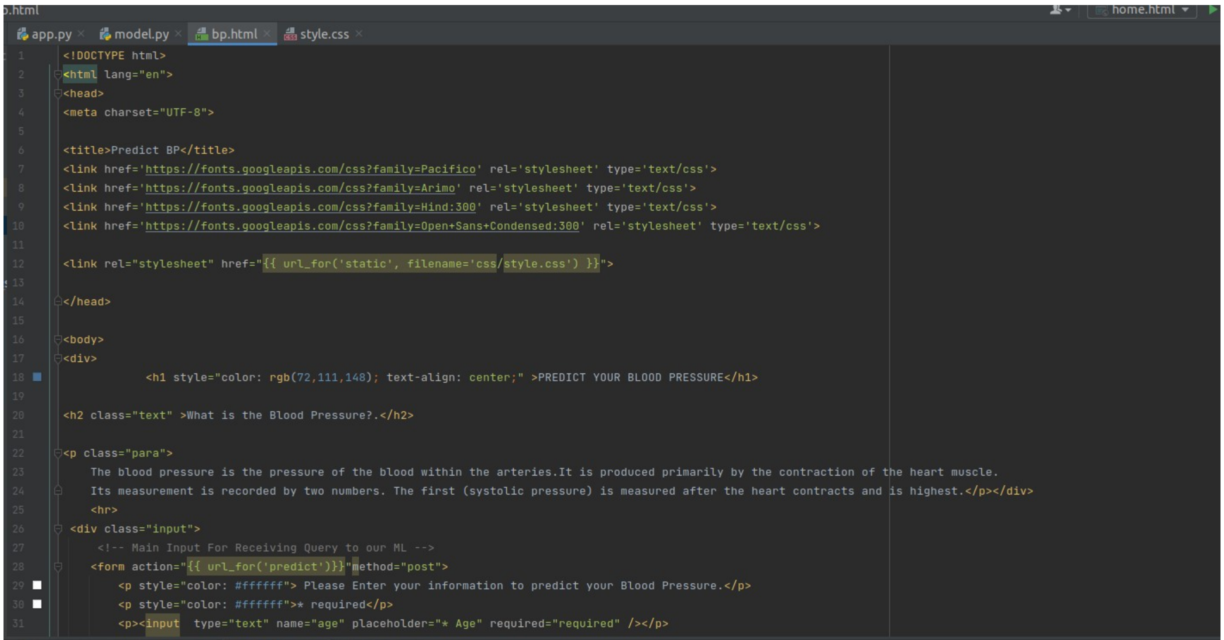
- Found a sample Blood pressure Data
- Converted the .xlsx file format to .pkl using python code.
- Saved the file in model.pkl in the directory



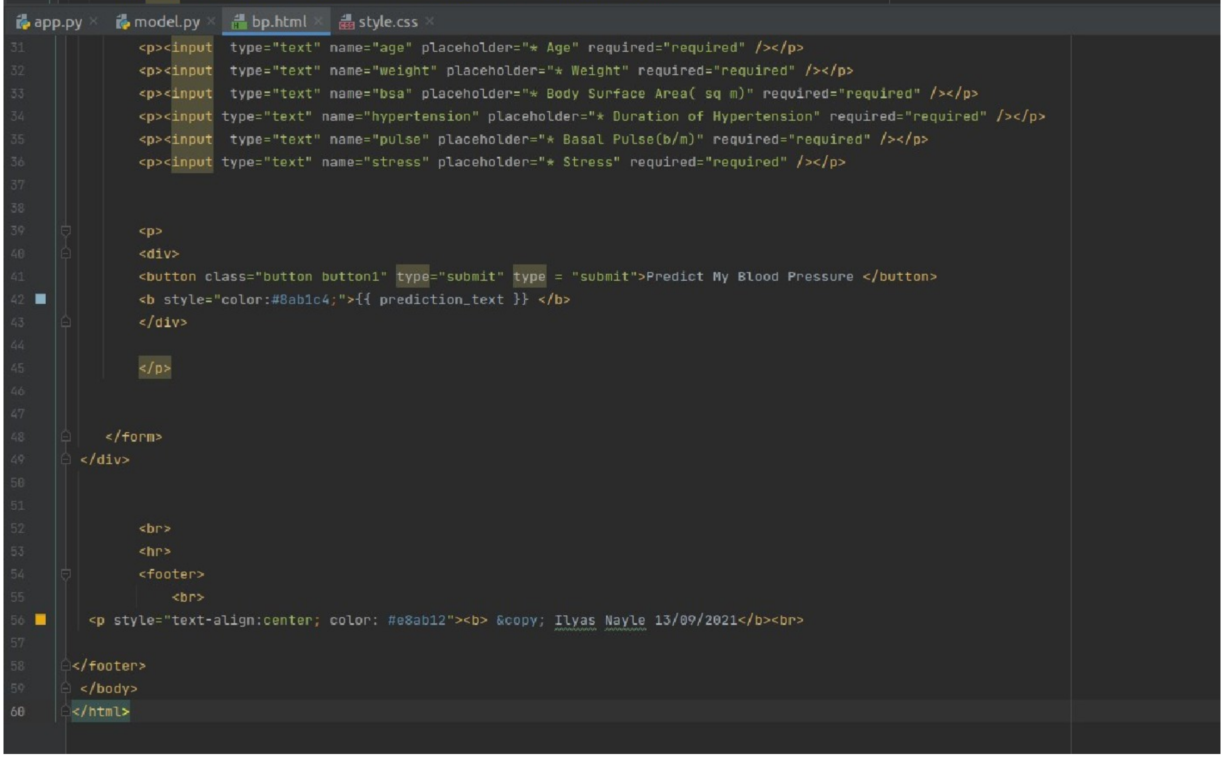
```python
import pandas as pan
import pickle
from sklearn.linear_model import LinearRegression

data = pan.read_excel('BP_simple_data.xlsx')

x1 = data.iloc[:,1:]
x2 = data.iloc[:,0:1]

reg = LinearRegression()
reg.fit(x1,x2)

prediction = reg.predict(x1)
pickle.dump(reg, open('model.pkl','wb'))
```

Step 2:

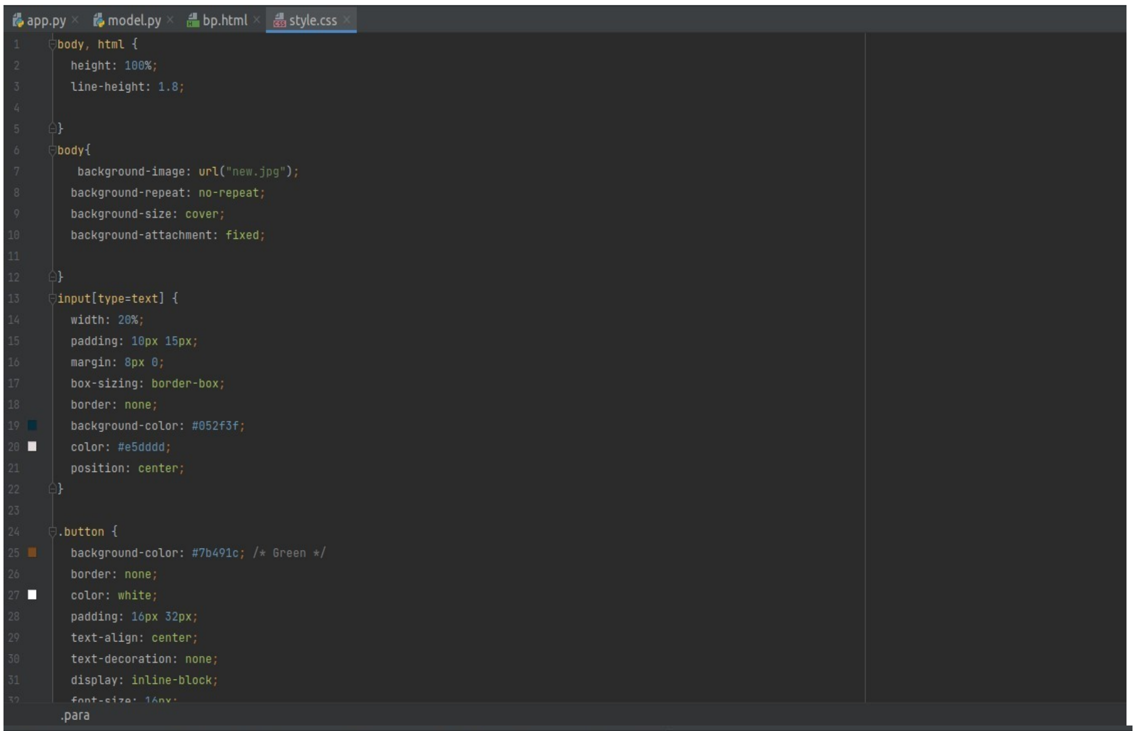
- Creating of the HTML (bp.html) and CSS (style.css) file



```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">

<title>Predict BP</title>
<link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet' type='text/css'>

<link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">

</head>

<body>
<div>
            <h1 style="color: rgb(72,111,148); text-align: center;" >PREDICT YOUR BLOOD PRESSURE</h1>

<h2 class="text" >What is the Blood Pressure?.</h2>

<p class="para">
    The blood pressure is the pressure of the blood within the arteries.It is produced primarily by the contraction of the heart muscle.
    Its measurement is recorded by two numbers. The first (systolic pressure) is measured after the heart contracts and is highest.</p></div>
    <hr>
 <div class="input">
     <!-- Main Input For Receiving Query to our ML -->
    <form action="{{ url_for('predict')}}"method="post">
        <p style="color: #ffffff"> Please Enter your information to predict your Blood Pressure.</p>
        <p style="color: #ffffff">* required</p>
        <p><input  type="text" name="age" placeholder="* Age" required="required" /></p>
```



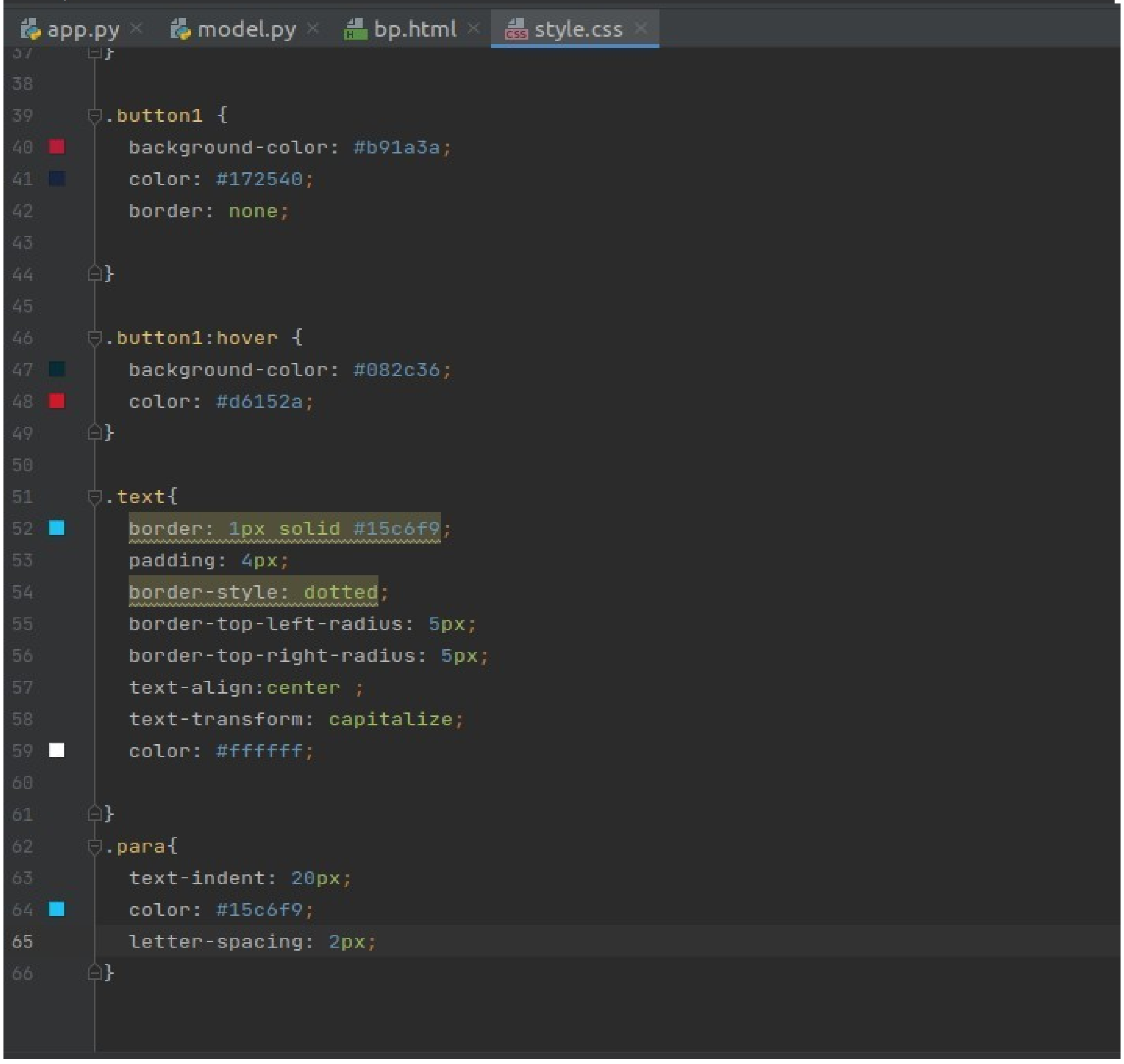
```html
        <p><input  type="text" name="age" placeholder="* Age" required="required" /></p>
        <p><input  type="text" name="weight" placeholder="* Weight" required="required" /></p>
        <p><input  type="text" name="bsa" placeholder="* Body Surface Area( sq m)" required="required" /></p>
        <p><input type="text" name="hypertension" placeholder="* Duration of Hypertension" required="required" /></p>
        <p><input  type="text" name="pulse" placeholder="* Basal Pulse(b/m)" required="required" /></p>
        <p><input type="text" name="stress" placeholder="* Stress" required="required" /></p>



        <p>
        <div>
        <button class="button button1" type="submit" type = "submit">Predict My Blood Pressure </button>
        <b style="color:#8ab1c4;">{{ prediction_text }} </b>
        </div>

        </p>


    </form>
 </div>



        <br>
        <hr>
        <footer>
            <br>
  <p style="text-align:center; color: #e8ab12"><b> &copy; Ilyas Nayle 13/09/2021</b><br>

</footer>
 </body>
</html>
```



app.py × model.py × bp.html × style.css ×

```css
body, html {
  height: 100%;
  line-height: 1.8;

}
body{
   background-image: url("new.jpg");
  background-repeat: no-repeat;
  background-size: cover;
  background-attachment: fixed;

}
input[type=text] {
  width: 20%;
  padding: 10px 15px;
  margin: 8px 0;
  box-sizing: border-box;
  border: none;
  background-color: #052f3f;
  color: #e5dddd;
  position: center;
}

.button {
  background-color: #7b491c; /* Green */
  border: none;
  color: white;
  padding: 16px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 16px;
```

.para

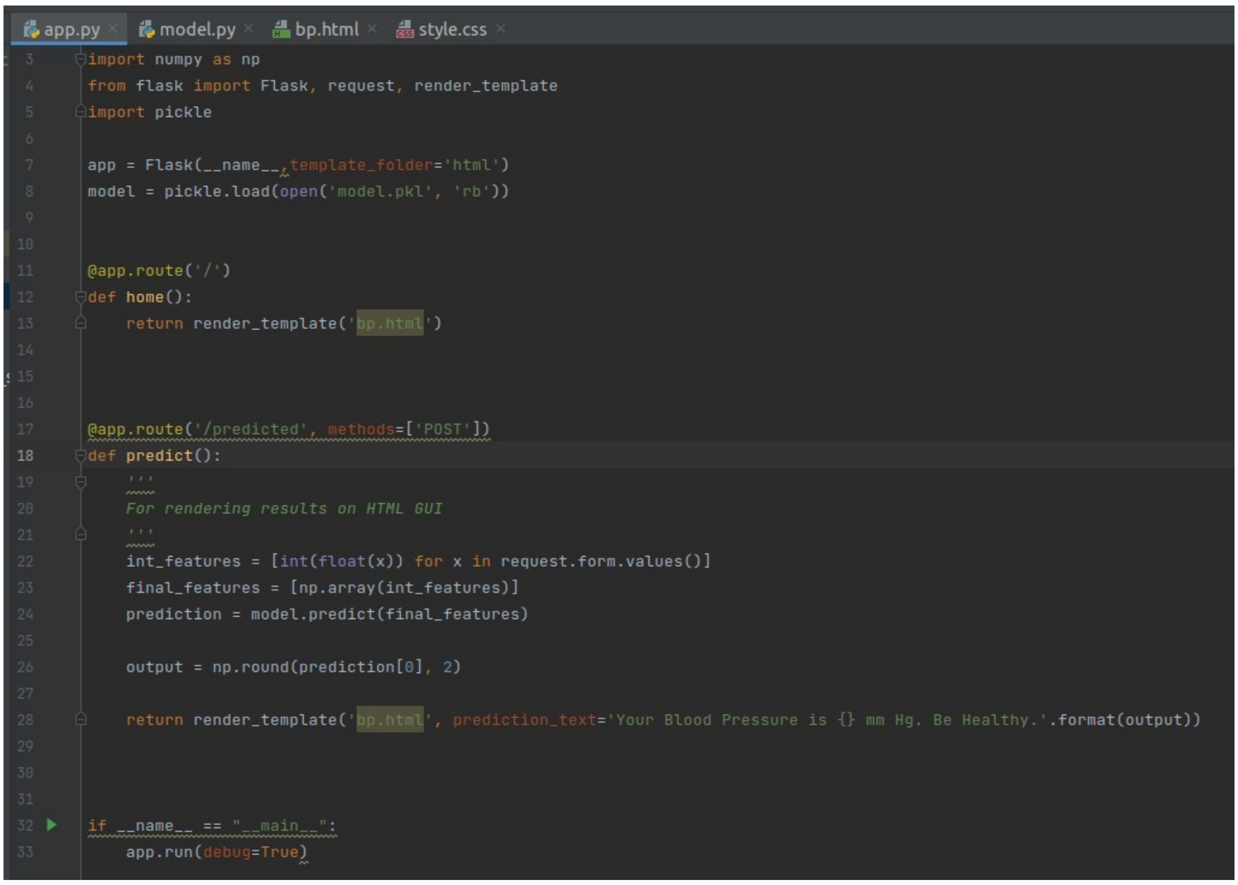


app.py × model.py × bp.html × style.css ×

```css
}

.button1 {
  background-color: #b91a3a;
  color: #172540;
  border: none;

}

.button1:hover {
  background-color: #082c36;
  color: #d6152a;
}

.text{
  border: 1px solid #15c6f9;
  padding: 4px;
  border-style: dotted;
  border-top-left-radius: 5px;
  border-top-right-radius: 5px;
  text-align:center ;
  text-transform: capitalize;
  color: #ffffff;

}
.para{
  text-indent: 20px;
  color: #15c6f9;
  letter-spacing: 2px;
}
```

Step 3:

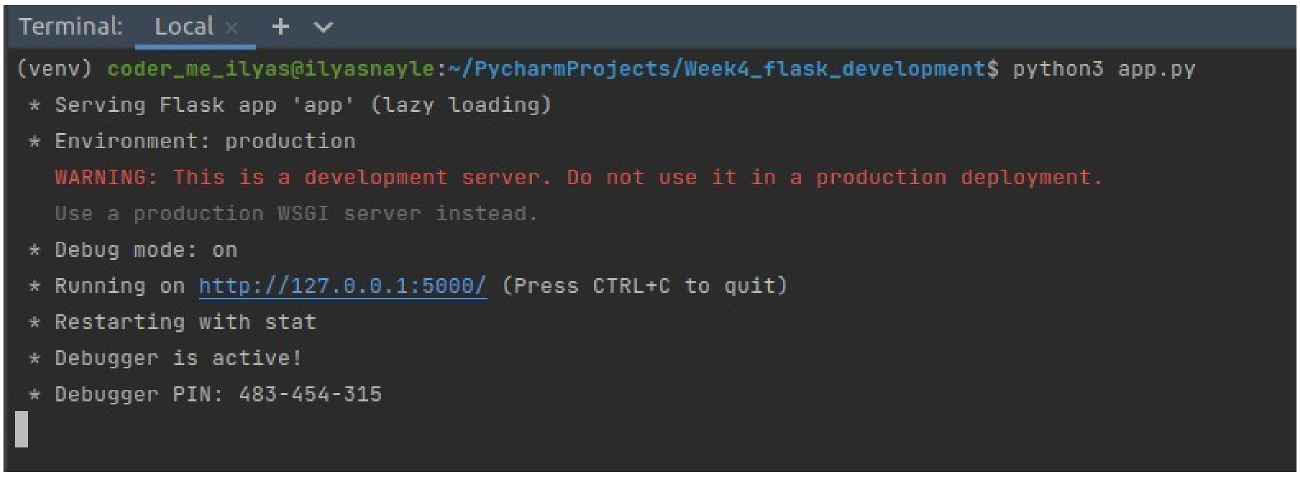
- Creating the Web Application app.py



```python
import numpy as np
from flask import Flask, request, render_template
import pickle

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


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



@app.route('/predicted', methods=['POST'])
def predict():
    '''
    For rendering results on HTML GUI
    '''
    int_features = [int(float(x)) for x in request.form.values()]
    final_features = [np.array(int_features)]
    prediction = model.predict(final_features)

    output = np.round(prediction[0], 2)

    return render_template('bp.html', prediction_text='Your Blood Pressure is {} mm Hg. Be Healthy.'.format(output))



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

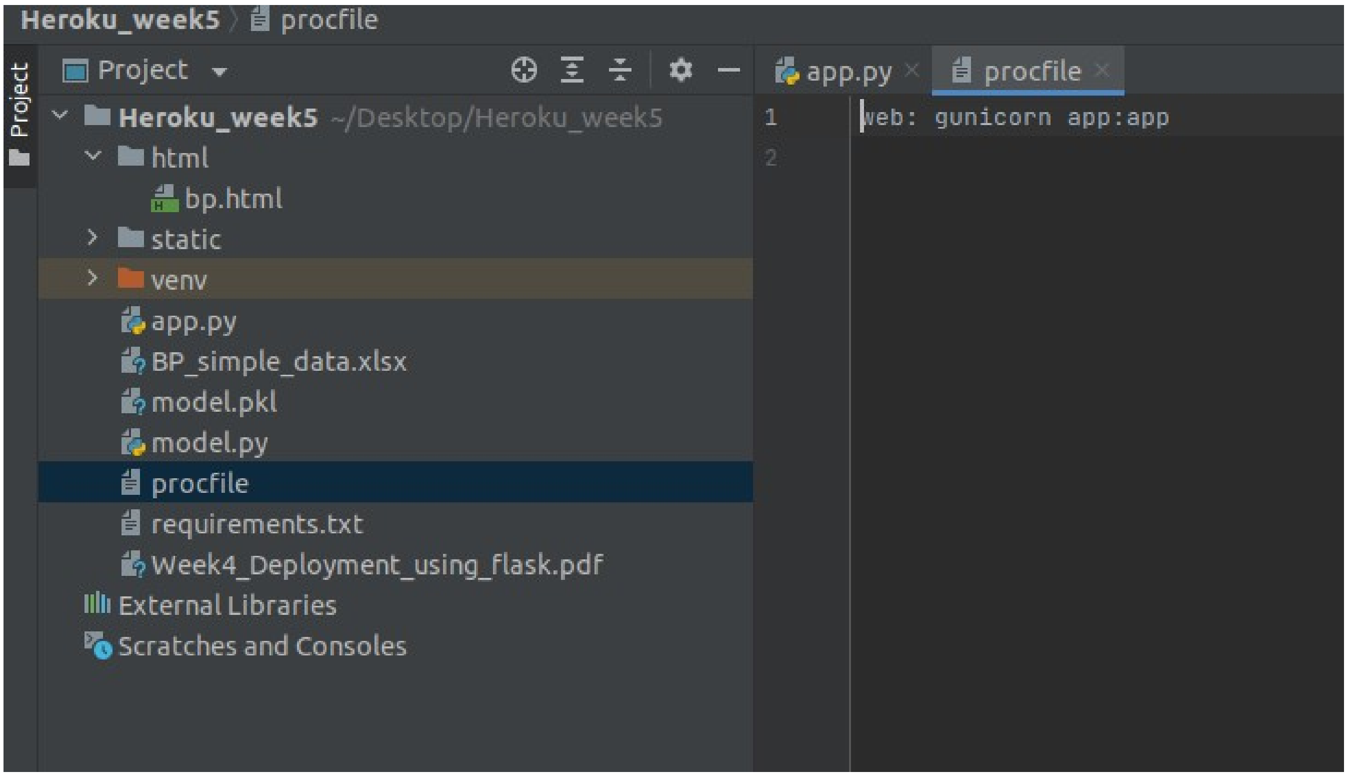
Step 4: Testing and Deployment of the model using command prompt to check if our app runs properly

- We open the command prompt
- Navigate to the location of the folder
- Run the file as: python3 app.py



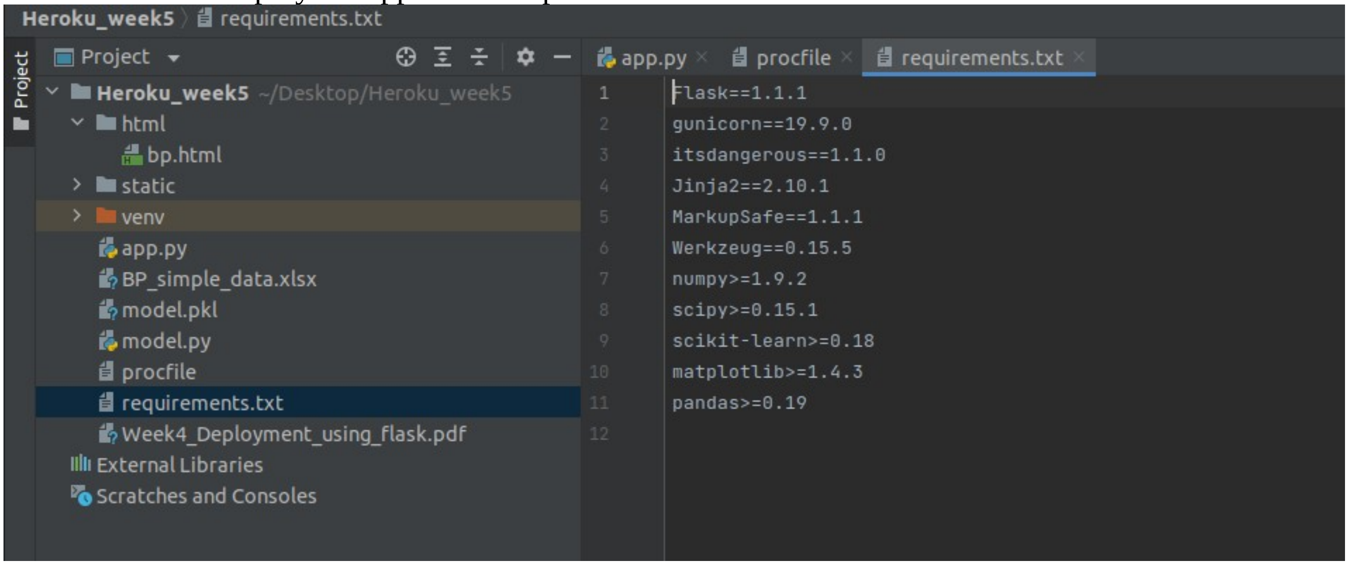
```
(venv) coder_me_ilyas@ilyasnayle:~/PycharmProjects/Week4_flask_development$ python3 app.py
 * Serving Flask app 'app' (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
 * Restarting with stat
 * Debugger is active!
 * Debugger PIN: 483-454-315
```

Step 5: Creating the Procfile(Without giving it any extension) which tells the heroku about the process beforehand.



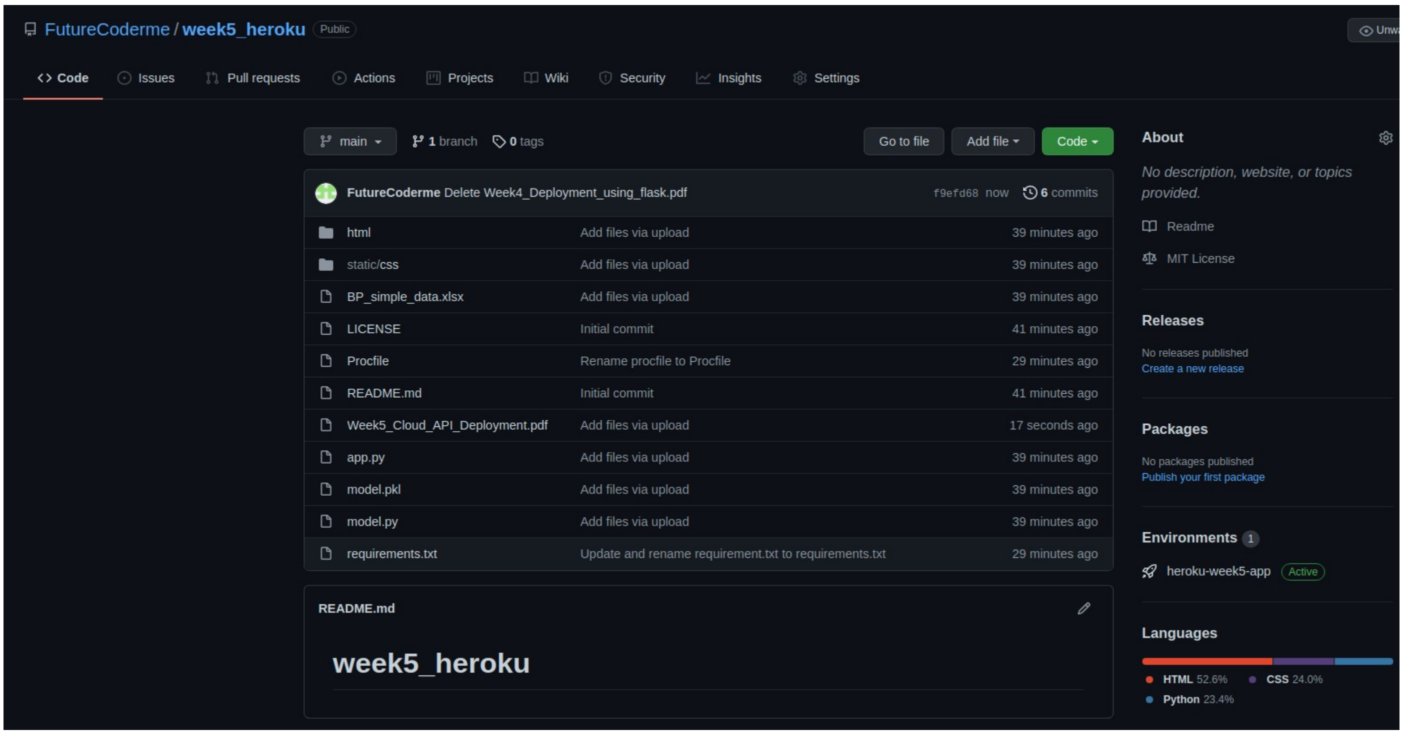
```
web: gunicorn app:app
```

Step 6: Creating the requirements.txt file which is for letting Heroku to set the application libraries and install them and Deploy the app after completion.



```
Flask==1.1.1
gunicorn==19.9.0
itsdangerous==1.1.0
Jinja2==2.10.1
MarkupSafe==1.1.1
Werkzeug==0.15.5
numpy>=1.9.2
scipy>=0.15.1
scikit-learn>=0.18
matplotlib>=1.4.3
pandas>=0.19
```

Step 7: Adding and pushing all the files in GitHub repository(in my case I created a new repository)



FutureCoderme / week5_heroku Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

FutureCoderme Delete Week4_Deployment_using_flask.pdf f9efd68 now 6 commits

html	Add files via upload	39 minutes ago
static/css	Add files via upload	39 minutes ago
BP_simple_data.xlsx	Add files via upload	39 minutes ago
LICENSE	Initial commit	41 minutes ago
Procfile	Rename procfile to Procfile	29 minutes ago
README.md	Initial commit	41 minutes ago
Week5_Cloud_API_Deployment.pdf	Add files via upload	17 seconds ago
app.py	Add files via upload	39 minutes ago
model.pkl	Add files via upload	39 minutes ago
model.py	Add files via upload	39 minutes ago
requirements.txt	Update and rename requirement.txt to requirements.txt	29 minutes ago

README.md

week5_heroku

About

No description, website, or topics provided.

Readme

MIT License

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

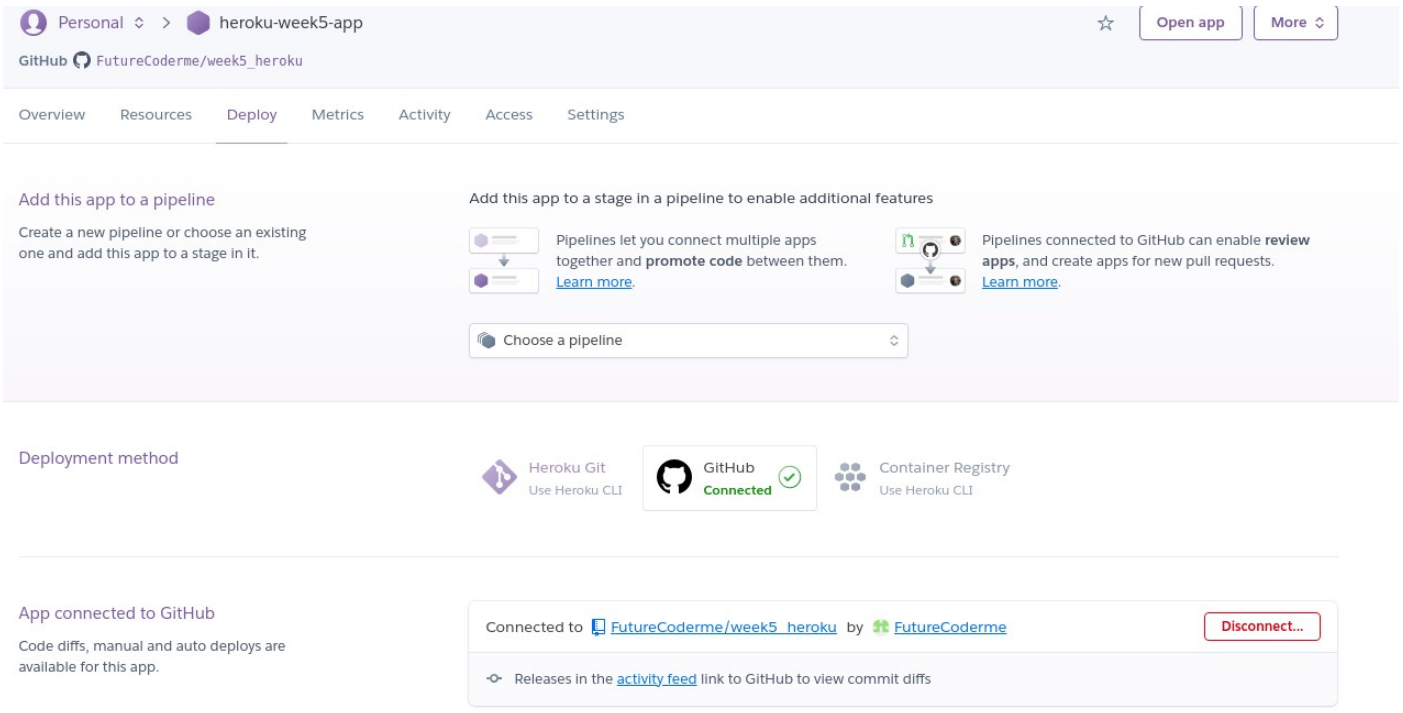
Environments 1

heroku-week5-app Active

Languages

HTML 52.6% CSS 24.0% Python 23.4%

Step 8: Creating Heroku account and linking GitHub with the Heroku account.



Personal > heroku-week5-app Open app More

GitHub FutureCoderme/week5_heroku

Overview Resources Deploy Metrics Activity Access Settings

Add this app to a pipeline

Create a new pipeline or choose an existing one and add this app to a stage in it.

Add this app to a stage in a pipeline to enable additional features

Pipelines let you connect multiple apps together and **promote code** between them. Learn more.

Pipelines connected to GitHub can enable **review apps**, and create apps for new pull requests. Learn more.

Choose a pipeline

Deployment method

Heroku Git Use Heroku CLI | GitHub Connected | Container Registry Use Heroku CLI

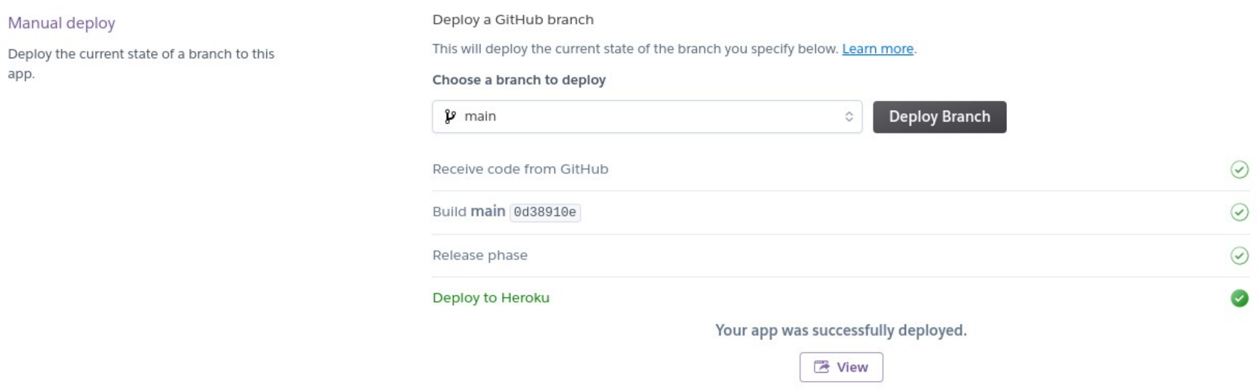
App connected to GitHub

Code diffs, manual and auto deploys are available for this app.

Connected to FutureCoderme/week5_heroku by FutureCoderme Disconnect...

Releases in the activity feed link to GitHub to view commit diffs

Step 9: Deploying the launch.



Manual deploy

Deploy the current state of a branch to this app.

Deploy a GitHub branch

This will deploy the current state of the branch you specify below. Learn more.

Choose a branch to deploy

main

Deploy Branch

Receive code from GitHub ✓

Build main `0d38910e` ✓

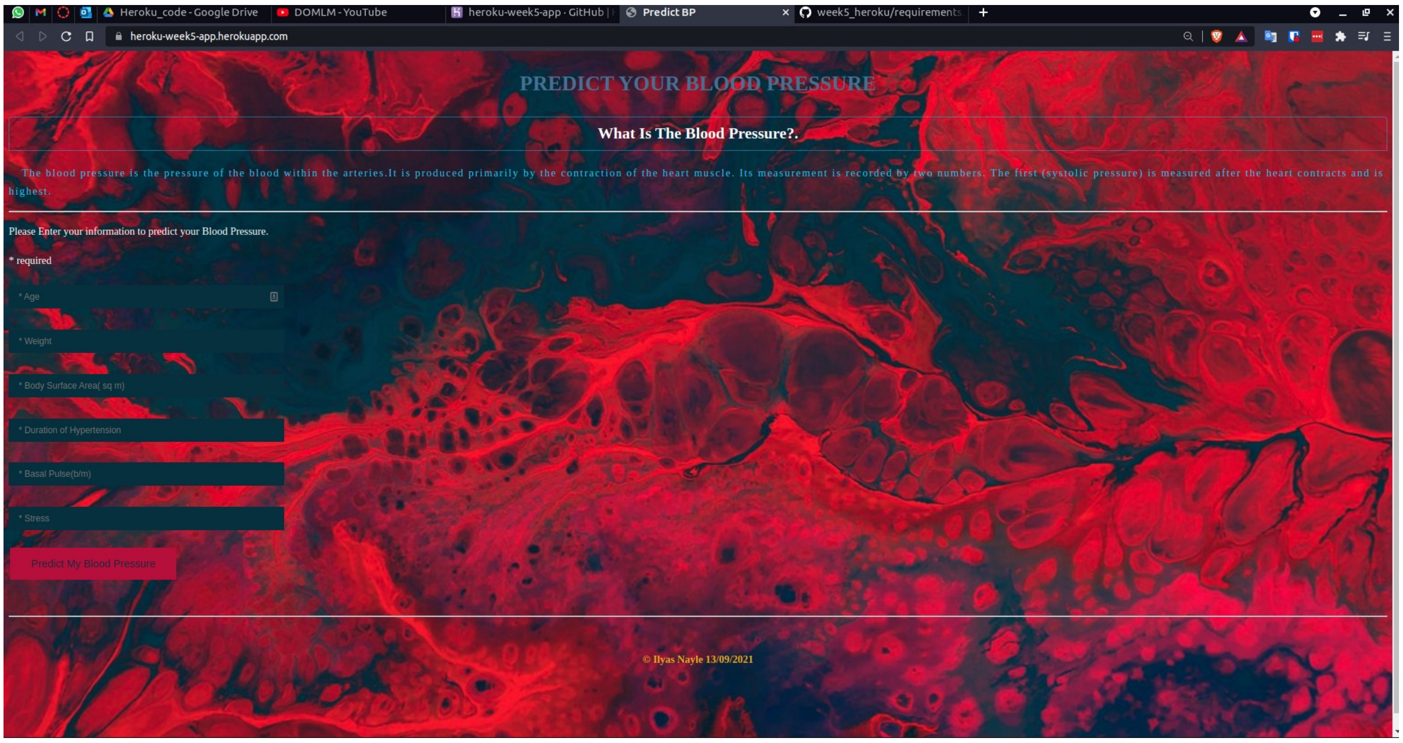
Release phase ✓

Deploy to Heroku ✓

Your app was successfully deployed.

View

After the successful deployment we will get the link for launching our app and the result is as below.



Heroku_code - Google Drive | DOMLM - YouTube | heroku-week5-app · GitHub | Predict BP | week5_heroku/requirements

heroku-week5-app.herokuapp.com

PREDICT YOUR BLOOD PRESSURE

What Is The Blood Pressure?.

The blood pressure is the pressure of the blood within the arteries.It is produced primarily by the contraction of the heart muscle. Its measurement is recorded by two numbers. The first (systolic pressure) is measured after the heart contracts and is highest.

Please Enter your information to predict your Blood Pressure.

* required

* Age

* Weight

* Body Surface Area(sq m)

* Duration of Hypertension

* Basal Pulse(b/m)

* Stress

Predict My Blood Pressure

© Ilyas Nayle 13/09/2021