

Data Glacier

Project Report

Internship Batch: LISUM05

Reference: <https://www.kaggle.com/yersever/500-person-gender-height-weight-bodymassindex>

Project: <https://github.com/karanamsuryachandra/BMI-Calculator/tree/master>

Project Abstract

The BMI is defined as the body mass divided by the square of the body height and is expressed in units of kg/m^2 . The height is entered in centimeters and weight in kilograms. A dataset is considered which contains height, weight, and BMIs of five hundred people. The machine learning model uses logistic regression to predict the output.

Required Data

The dataset contains information about gender, height, weight, and BMI index of individuals

Height: Integer (cm)

Weight: Integer (Kg)

Output after predicting BMI using ML

0 - Extremely Weak

1 - Weak

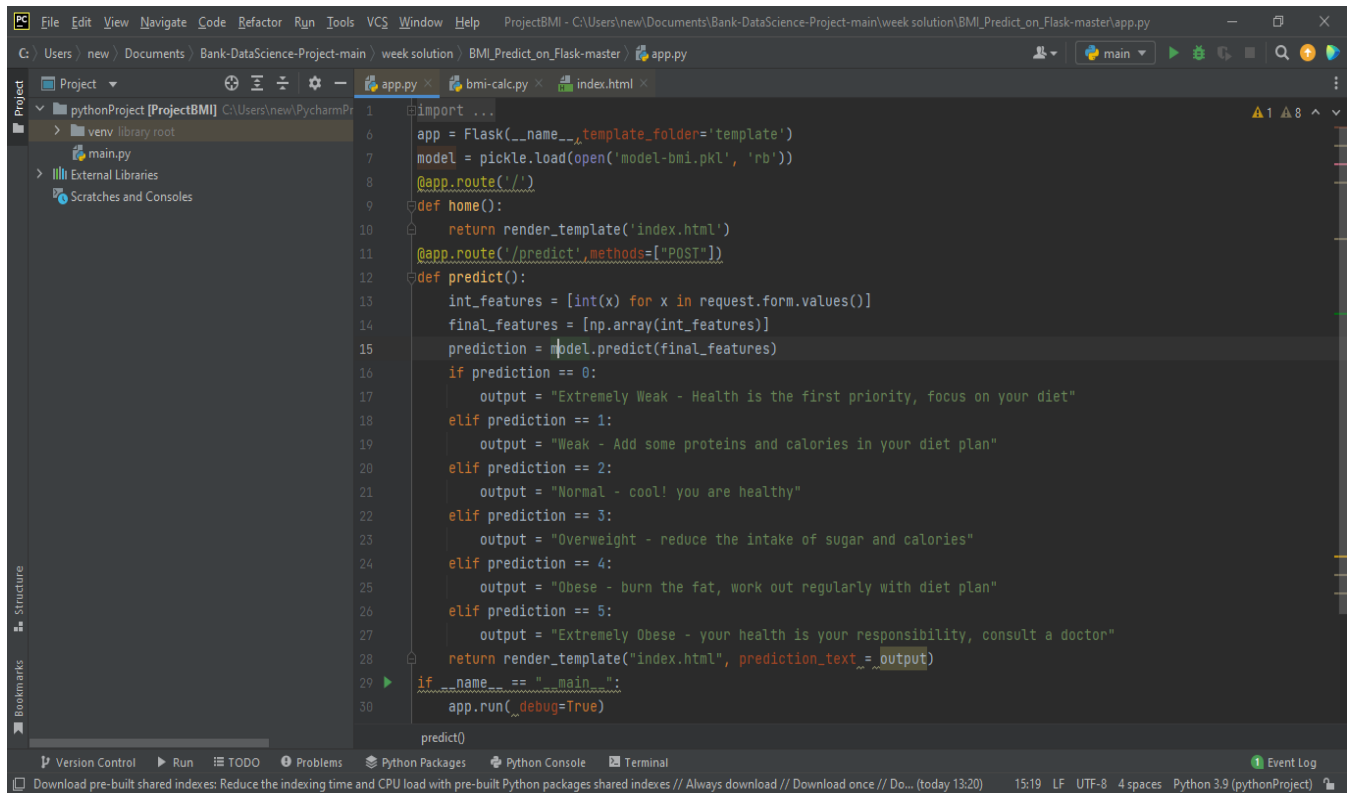
2 - Normal

3 - Overweight

4 - Obesity

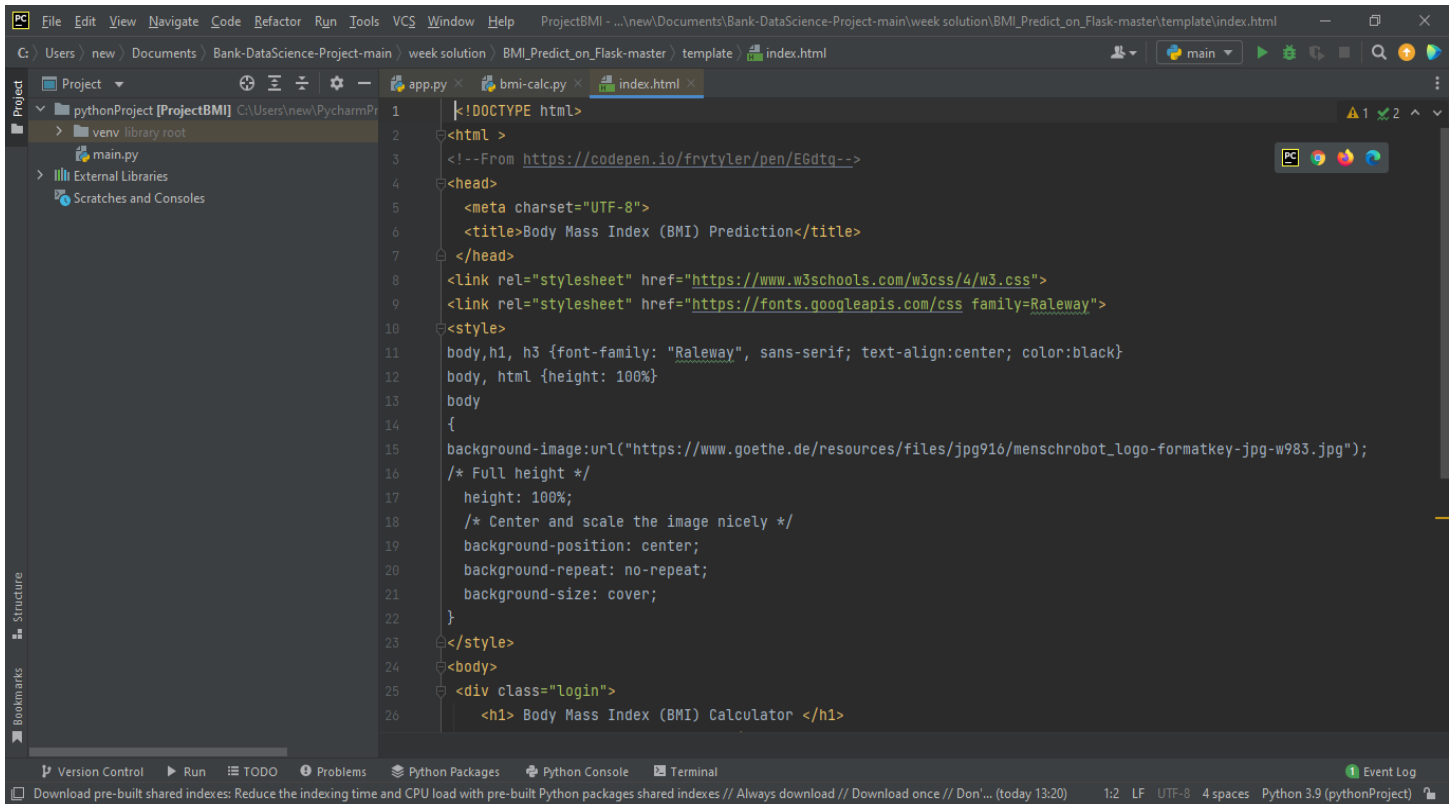
5 - Extreme Obesity

1. Prediction, Pickling and Depickling - “app.py”

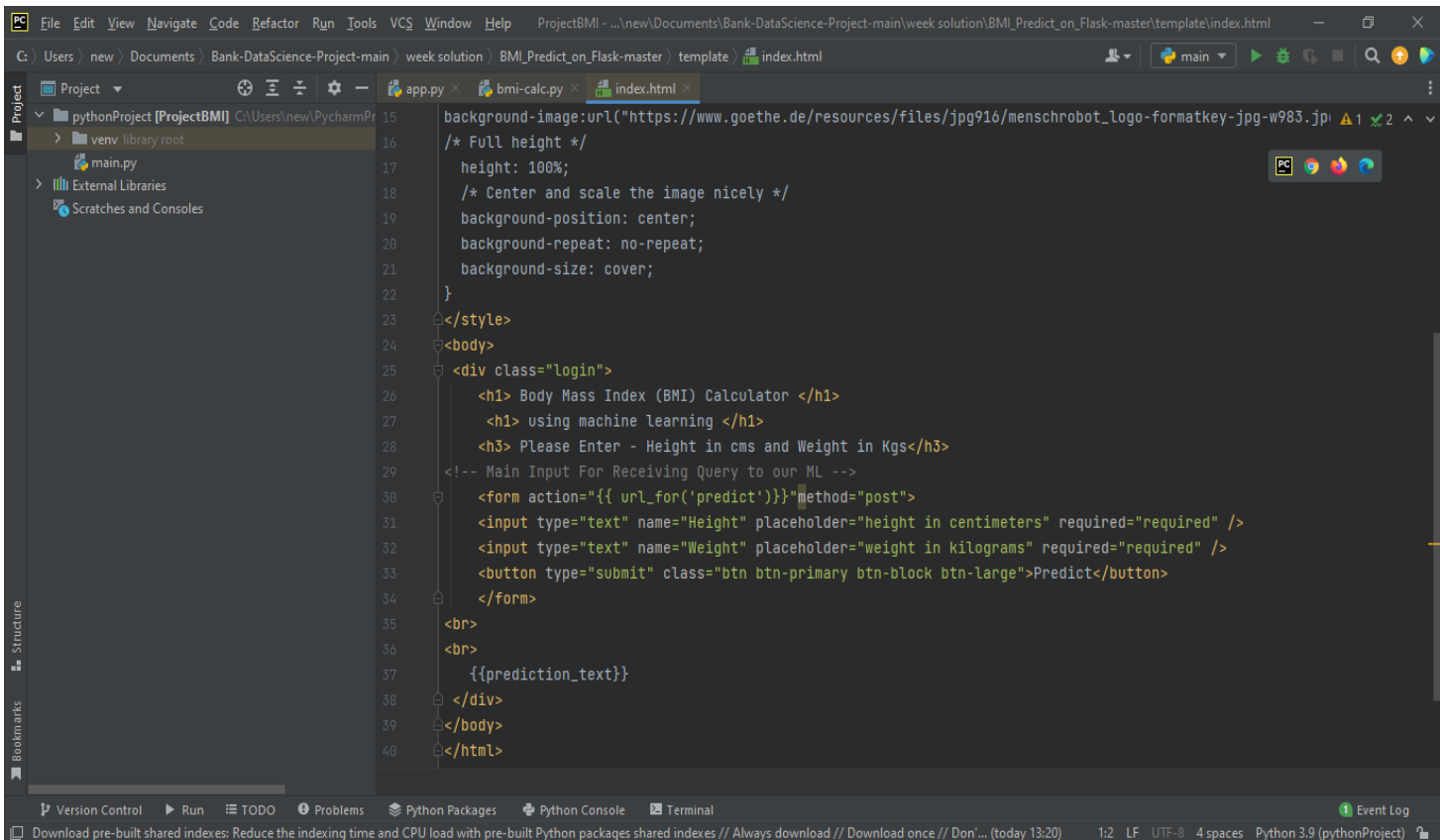


```
1 import ...
2
3 app = Flask(__name__, template_folder='template')
4 model = pickle.load(open('model-bmi.pkl', 'rb'))
5
6 @app.route('/')
7 def home():
8     return render_template('index.html')
9
10 @app.route('/predict', methods=["POST"])
11 def predict():
12     int_features = [int(x) for x in request.form.values()]
13     final_features = [np.array(int_features)]
14     prediction = model.predict(final_features)
15
16     if prediction == 0:
17         output = "Extremely Weak - Health is the first priority, focus on your diet"
18     elif prediction == 1:
19         output = "Weak - Add some proteins and calories in your diet plan"
20     elif prediction == 2:
21         output = "Normal - cool! you are healthy"
22     elif prediction == 3:
23         output = "Overweight - reduce the intake of sugar and calories"
24     elif prediction == 4:
25         output = "Obese - burn the fat, work out regularly with diet plan"
26     elif prediction == 5:
27         output = "Extremely Obese - your health is your responsibility, consult a doctor"
28     return render_template("index.html", prediction_text = output)
29
30 if __name__ == "__main__":
31     app.run(debug=True)
```

2. HTML code (index.html)



```
1 <!DOCTYPE html>
2 <html>
3   <!-- From https://codepen.io/frtytyler/pen/E6dtq -->
4   <head>
5     <meta charset="UTF-8">
6     <title>Body Mass Index (BMI) Prediction</title>
7   </head>
8   <link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
9   <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Raleway">
10  <style>
11    body, h1, h3 {font-family: "Raleway", sans-serif; text-align:center; color:black}
12    body, html {height: 100%}
13    body
14    {
15      background-image:url("https://www.goethe.de/resources/files/jpg916/menschrobot_logo-formatkey-jpg-w983.jpg");
16      /* Full height */
17      height: 100%;
18      /* Center and scale the image nicely */
19      background-position: center;
20      background-repeat: no-repeat;
21      background-size: cover;
22    }
23  </style>
24  <body>
25    <div class="login">
26      <h1> Body Mass Index (BMI) Calculator </h1>
```



```
15    background-image:url("https://www.goethe.de/resources/files/jpg916/menschrobot_logo-formatkey-jpg-w983.jpg");
16    /* Full height */
17    height: 100%;
18    /* Center and scale the image nicely */
19    background-position: center;
20    background-repeat: no-repeat;
21    background-size: cover;
22  }
23 </style>
24 <body>
25   <div class="login">
26     <h1> Body Mass Index (BMI) Calculator </h1>
27     <h1> using machine learning </h1>
28     <h3> Please Enter - Height in cms and Weight in Kgs</h3>
29     <!-- Main Input For Receiving Query to our ML -->
30     <form action="{ url_for('predict')}" method="post">
31       <input type="text" name="Height" placeholder="height in centimeters" required="required" />
32       <input type="text" name="Weight" placeholder="weight in kilograms" required="required" />
33       <button type="submit" class="btn btn-primary btn-block btn-large">Predict</button>
34     </form>
35     <br>
36     <br>
37     {{prediction_text}}
38   </div>
39 </body>
40 </html>
```

3.Executing the app.py file anaconda navigator

```
Anaconda Prompt (anaconda3) - python app.py

(base) C:\Users\new>cd documents

(base) C:\Users\new\Documents>cd bank-datascience-project-main

(base) C:\Users\new\Documents\Bank-DataScience-Project-main>cd week solution

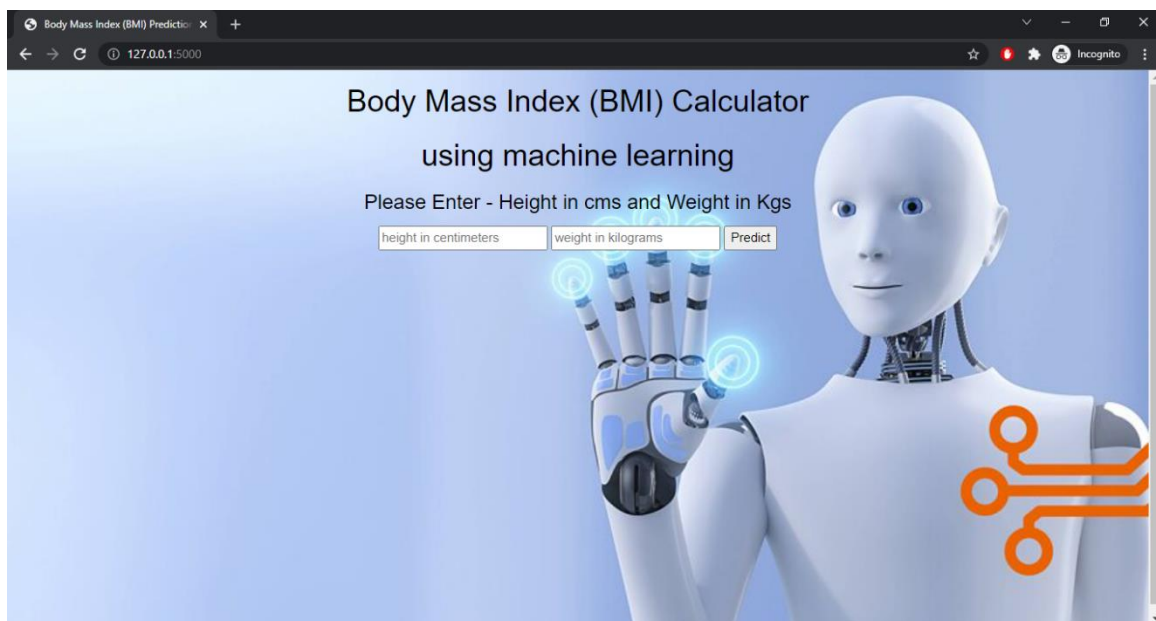
(base) C:\Users\new\Documents\Bank-DataScience-Project-main\week solution>cd bmi_predict_on_flask-master

(base) C:\Users\new\Documents\Bank-DataScience-Project-main\week solution\BMI_Predict_on_Flask-master>python add.py
python: can't open file 'C:\Users\new\Documents\Bank-DataScience-Project-main\week solution\BMI_Predict_on_Flask-master\
add.py': [Errno 2] No such file or directory

(base) C:\Users\new\Documents\Bank-DataScience-Project-main\week solution\BMI_Predict_on_Flask-master>python 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
* Restarting with watchdog (windowsapi)
* Debugger is active!
* Debugger PIN: 924-713-182
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

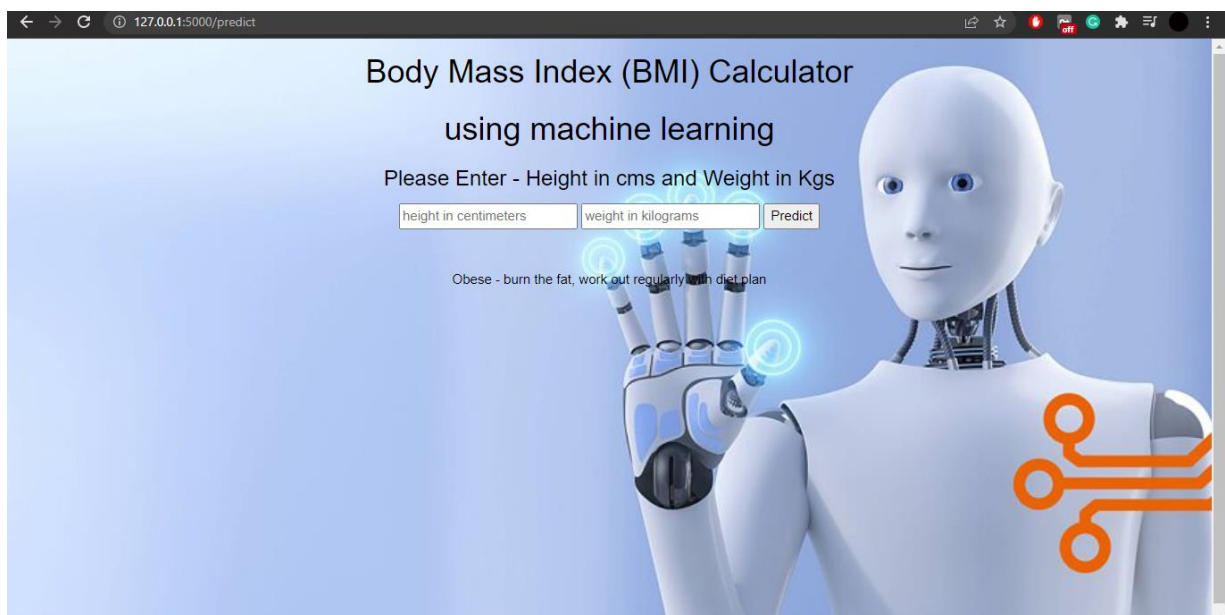
Output - <http://127.0.0.1:5000/>

3. BMI Calculation using Machine learning model

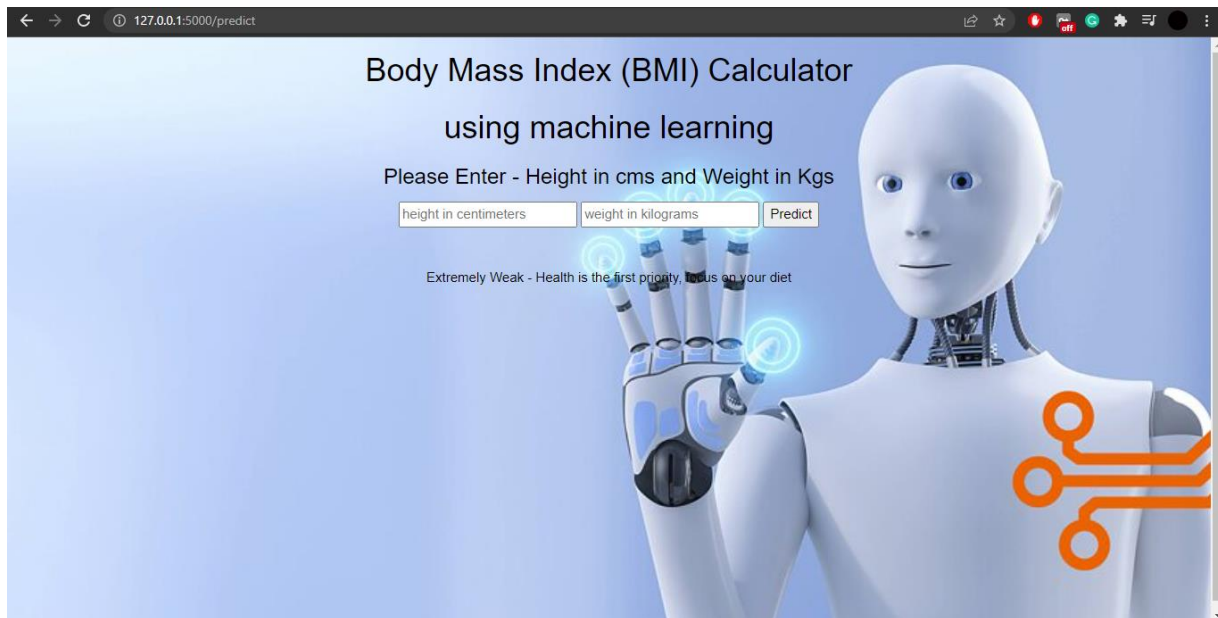


Predictions by the model:

- 1. Extremely Weak - Health is the first priority, focus on your diet**
- 2. Weak - Add some proteins and calories in your diet plan**
- 3. Normal - cool! you are healthy**
- 4. Overweight - reduce the intake of sugar and calories**
- 5. Obese - burn the fat, work out regularly with diet plan**
- 6. Extremely Obese - your health is your responsibility, consult a doctor**

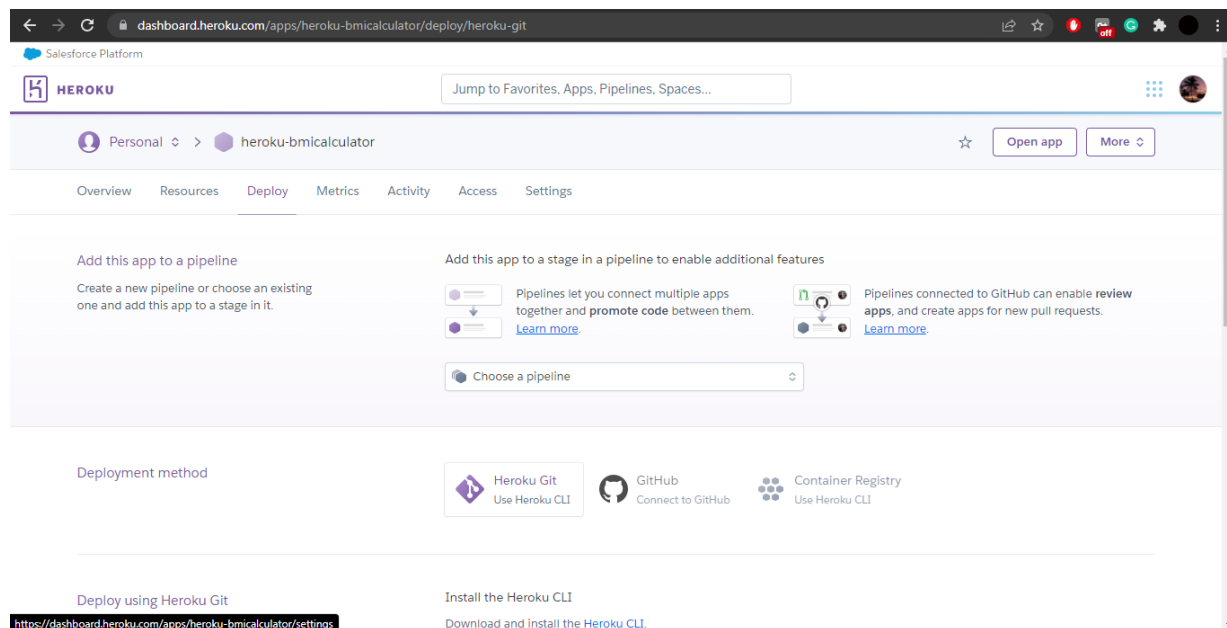


Extremely weak prediction

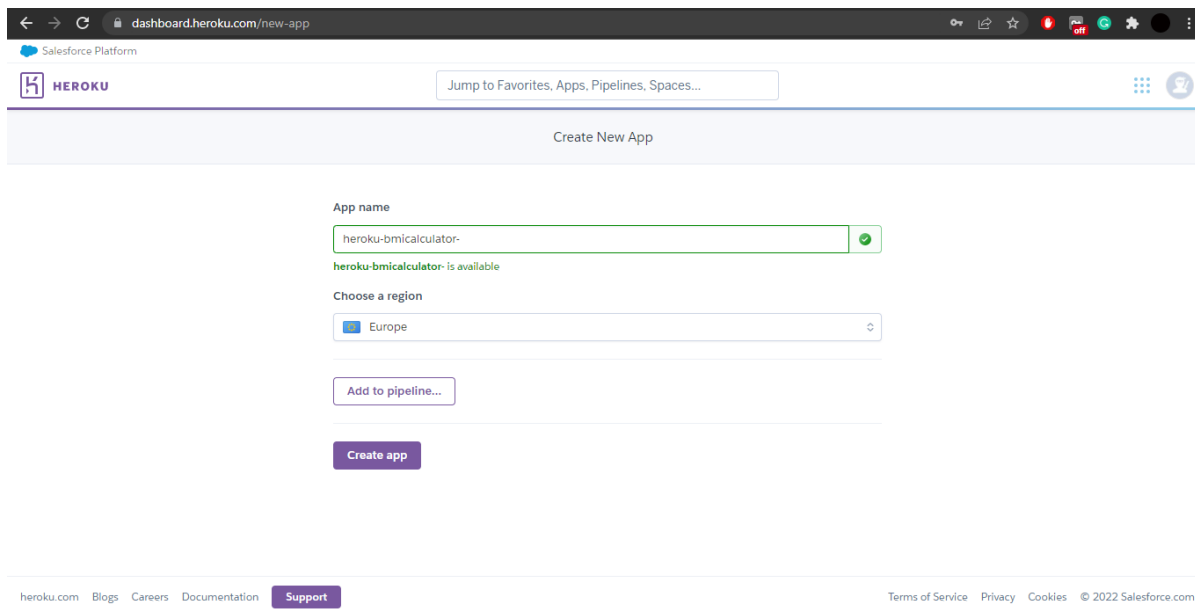


Deployment using Heroku app

1. Heroku dashboard

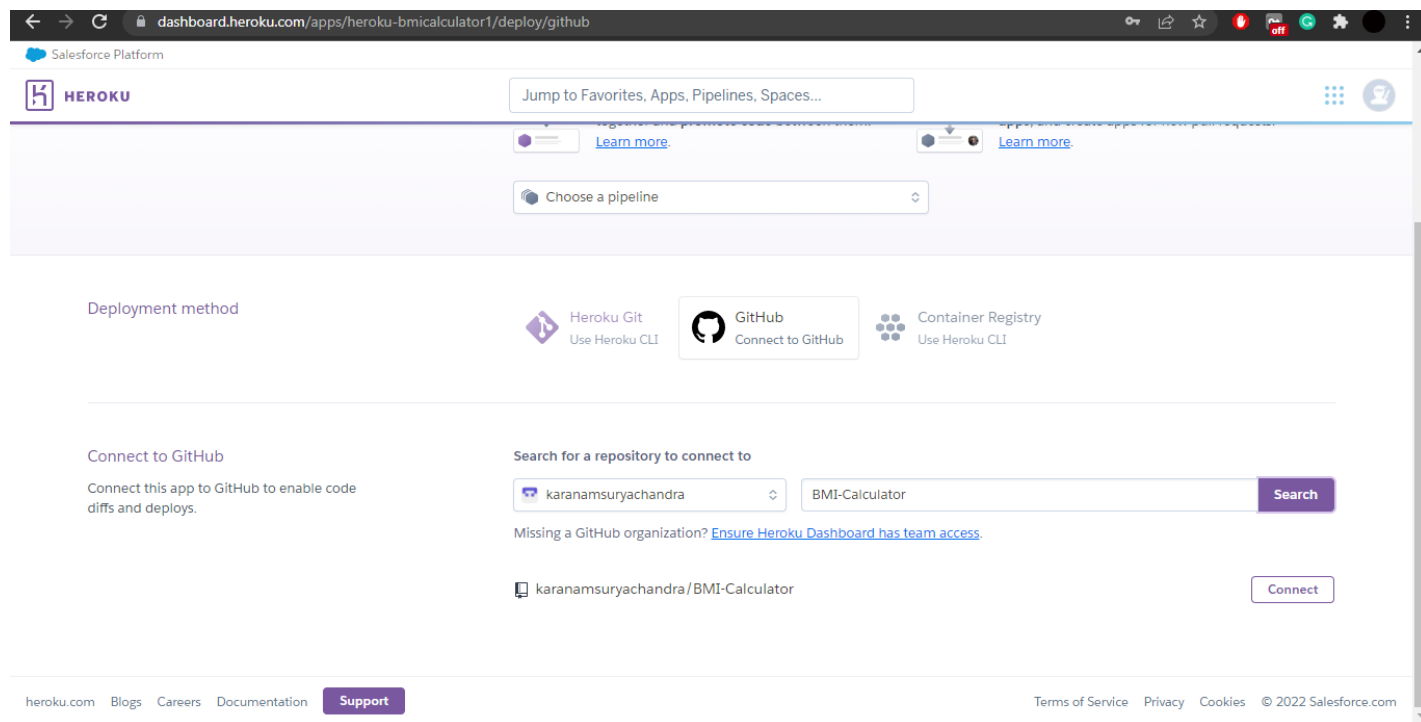


2. Create a new app with unique name



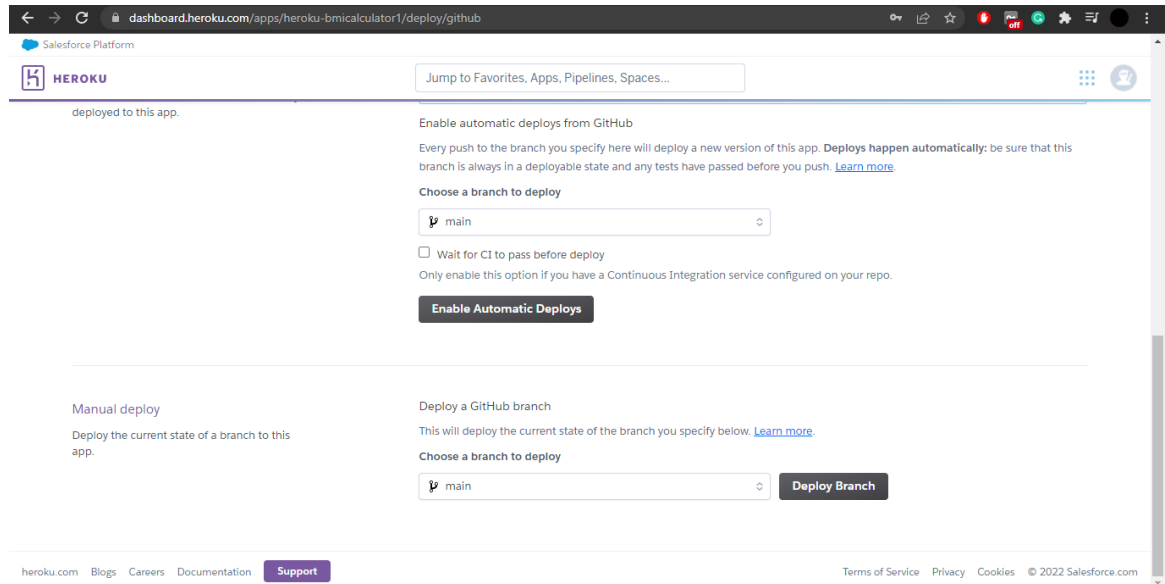
The screenshot shows the Heroku dashboard at `dashboard.heroku.com/new-app`. The page has a header with the Heroku logo and a search bar. Below the header is a 'Create New App' section. It includes a text input for 'App name' with the value 'heroku-bmicalculator-' and a green checkmark icon. Below the input, a message states 'heroku-bmicalculator- is available'. There is a 'Choose a region' dropdown menu with 'Europe' selected. Below the dropdown are two buttons: 'Add to pipeline...' and 'Create app'.

3. Connect the GitHub repository to Heroku

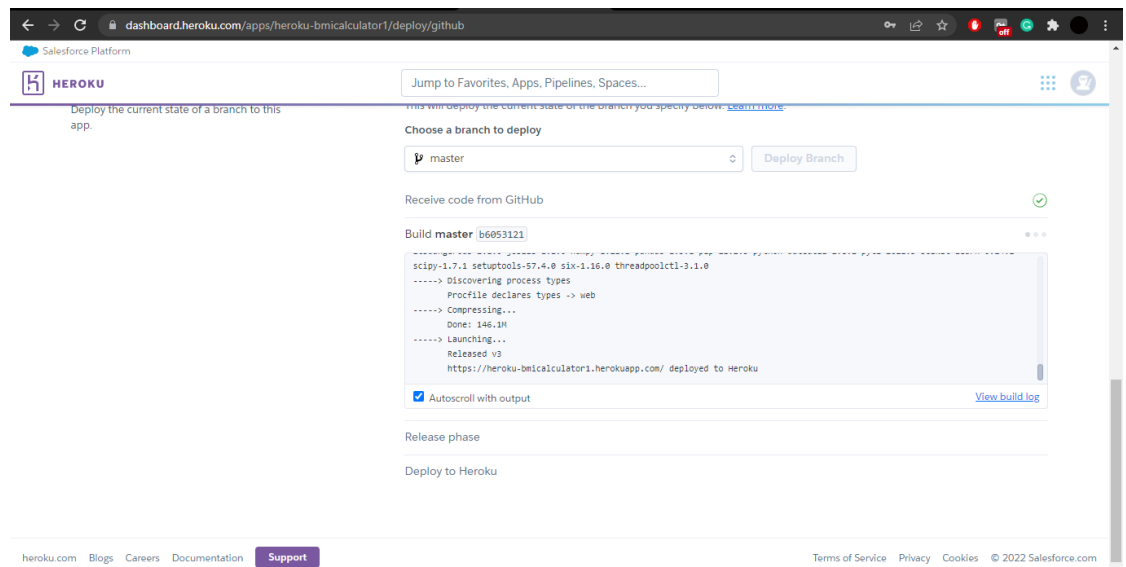


The screenshot shows the Heroku dashboard at `dashboard.heroku.com/apps/heroku-bmicalculator1/deploy/github`. The page has a header with the Heroku logo and a search bar. Below the header is a 'Deployment method' section. It includes a 'Choose a pipeline' dropdown menu. Below the dropdown are three options: 'Heroku Git Use Heroku CLI', 'GitHub Connect to GitHub', and 'Container Registry Use Heroku CLI'. Below the 'Connect to GitHub' section, there is a 'Search for a repository to connect to' section. It includes a text input with the value 'karanamsuryachandra' and a dropdown menu with the value 'BMI-Calculator'. Below the input is a 'Search' button. Below the search results, there is a message 'Missing a GitHub organization? Ensure Heroku Dashboard has team access.' and a list of repositories with the value 'karanamsuryachandra/BMI-Calculator' and a 'Connect' button.

4. Choose the required branch to deploy and select manual deploy



5. If required, add requirements.txt file and import Heroku/ python library in the settings option. Select “Deploy Branch”.



6. The ML model is deployed using Heroku and a sample of three outputs are given below.



