Machine Learning Approach For Employee Performance Prediction

1. INTRODUCTION

1.1 Overview

In this project we are going to analyse and predict the performance of employees in an organization on the basis of various factors, including, but not limited to, individual and domain specific characteristics, nature and level of schooling, socioeconomic status and different psychological factors.

1.2 Purpose

The purpose of this project is to predict the performance of employee.

2. PROJECT INITIALIZATION AND PLANNING PHASE

2.1 Define Problem Statement

Please reffer to our Define Problem Statement Document for a comprehensive overview of our project's problem statements.

2.2 Project proposal (Proposed Solution)

Please reffer to our Project Proposal (Proposed Solution) Document for a comprehensive overview of our Project Proposal.

2.3 Initial Project Planning Report

Please reffer to our Initial Project Planning Report Document for a comprehensive overview of our Initial Project Planning.

3. DATA COLLECTION AND PREPROCESSING PHASE

3.1 Data Collection Plan & Raw Data Sources Identification Report

Please reffer to our Data Collection Plan & Raw Data Sources Identification Report Document for a comprehensive overview of our Data Collection Plan & Raw Data Sources Identification..

3.2 Data Quality Report

Please reffer to our Data Quality Report Document for a comprehensive overview of our Data Quality.

3.3 Data Exploration & Preprocessing Report

Please reffer to Preprocessing Report Document for a comprehensive overview of our Data Exploration And Preprocessing.

4. MODEL DEVELOPMENT PHASE

4.1 Feature Selection Report

Please reffer to Feature Selection Report Document for a comprehensive overview of our Feature Selection.

4.2 Model Selection Report

Please reffer to Model Selection Report Document for a comprehensive overview of our Model Selection.

4.3 Initial Model Training Code,Model Validation And Evaluation Report

Please reffer Initial Model Training Code,Model Validation And Evaluation Report Document for a comprehensive overview of our Initial Model Training Code,Model Validation And Evaluation. Link:

5.MODEL OPTIMIZATION AND TUNING PHASE

5.1 Hyperparameter Tuning documentation

Please reffer Model Optimization and Tuning Report Document for a comprehensive overview of our Hyperparameter Tuning.

5.2 Performance Metrics Classification Report

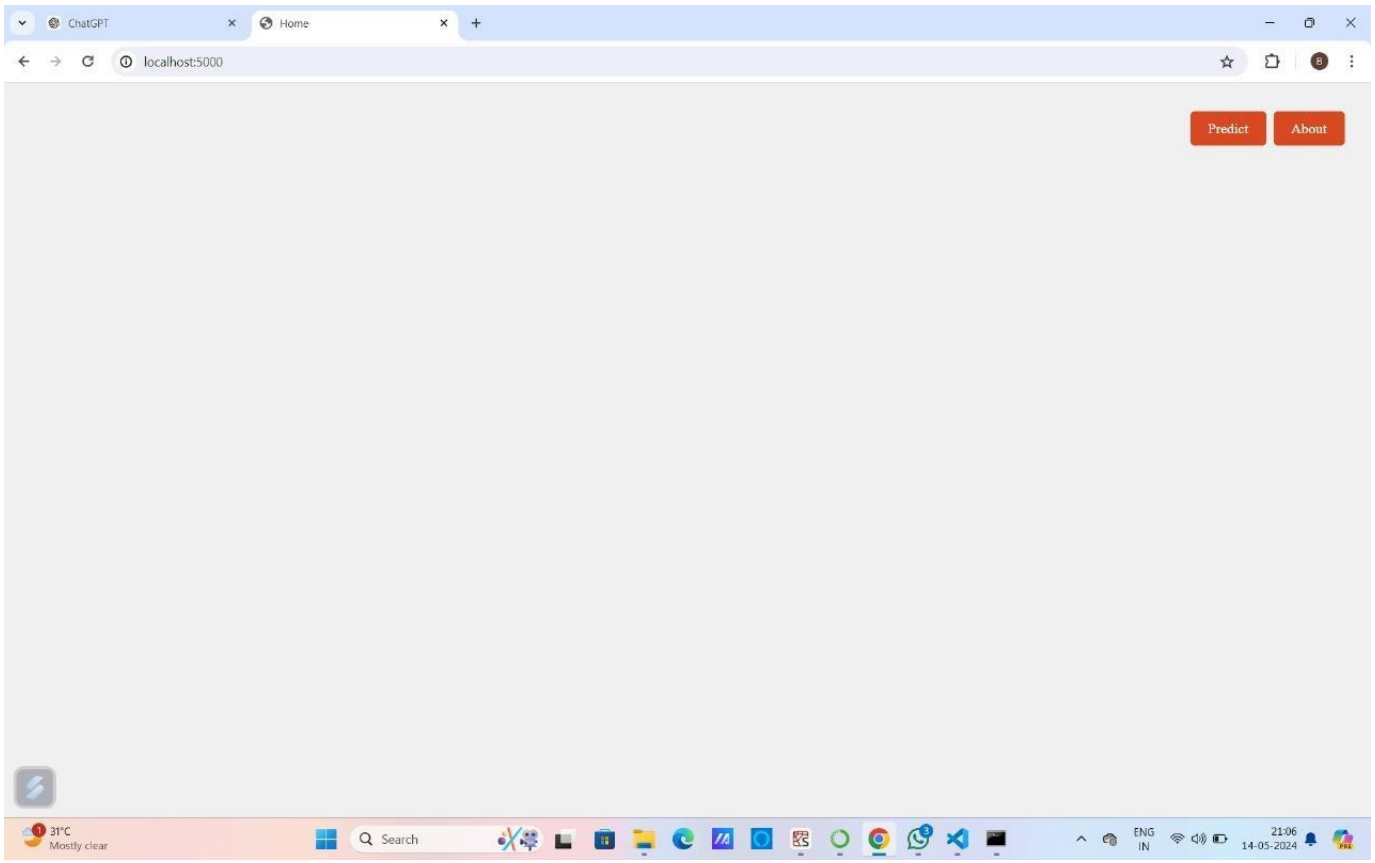
Please reffer Performance Metrics Classification Report Document for a comprehensive overview of our Performance Metrics Classification.

5.3 Final Model Selection Justification

Please reffer Final Model Selection Justification Report Document for a comprehensive overview of our Final Model Selection Justification.

6. RESULT

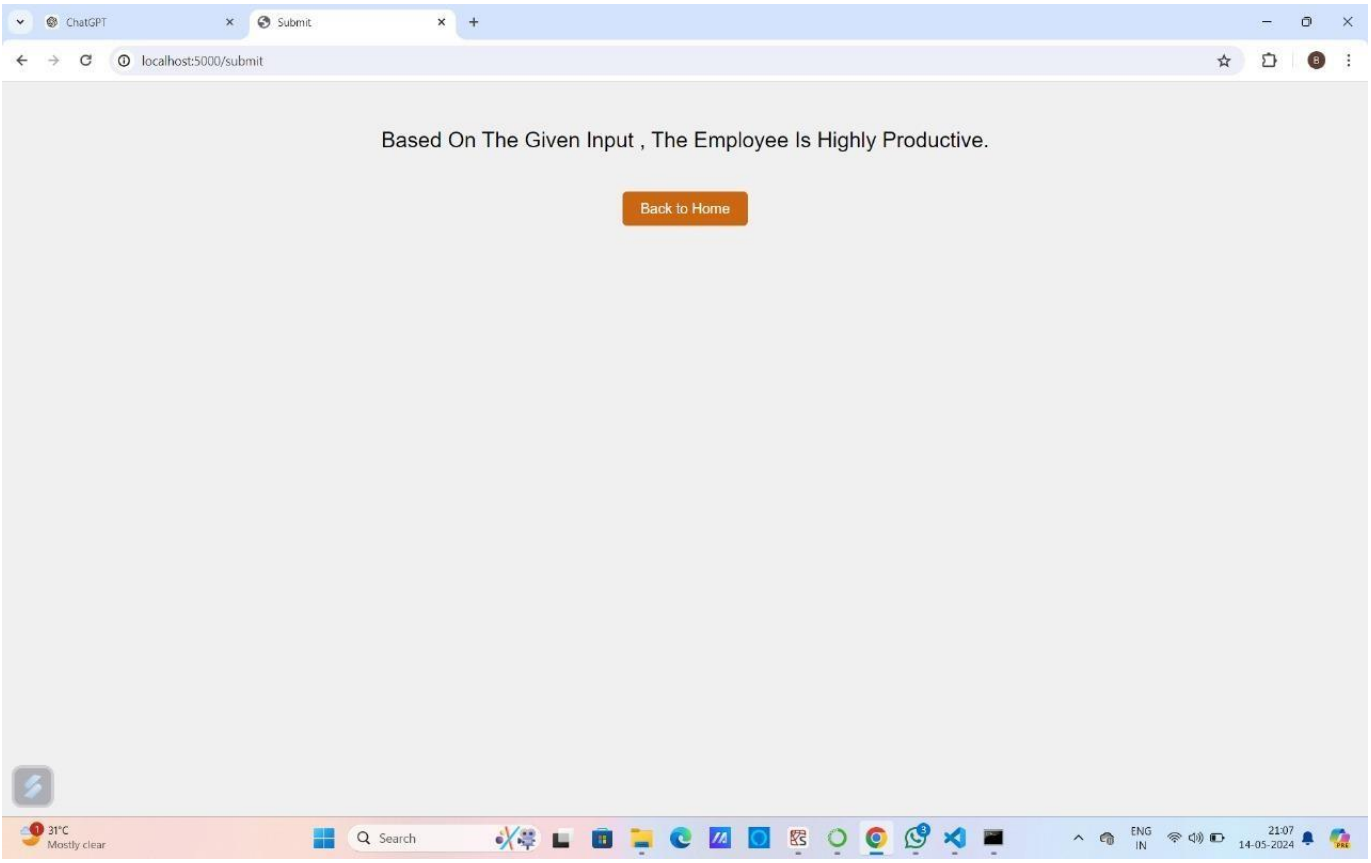
5.1 Output Screenshots Home Page:



❼When we click on the “Predict” button which is on the top right of my web page it will redirects to the another page where we can give inputs to our model.

❼When we click on “About” button which is on the top right of my web page it will redirects to the another page where we find some details about my web page.

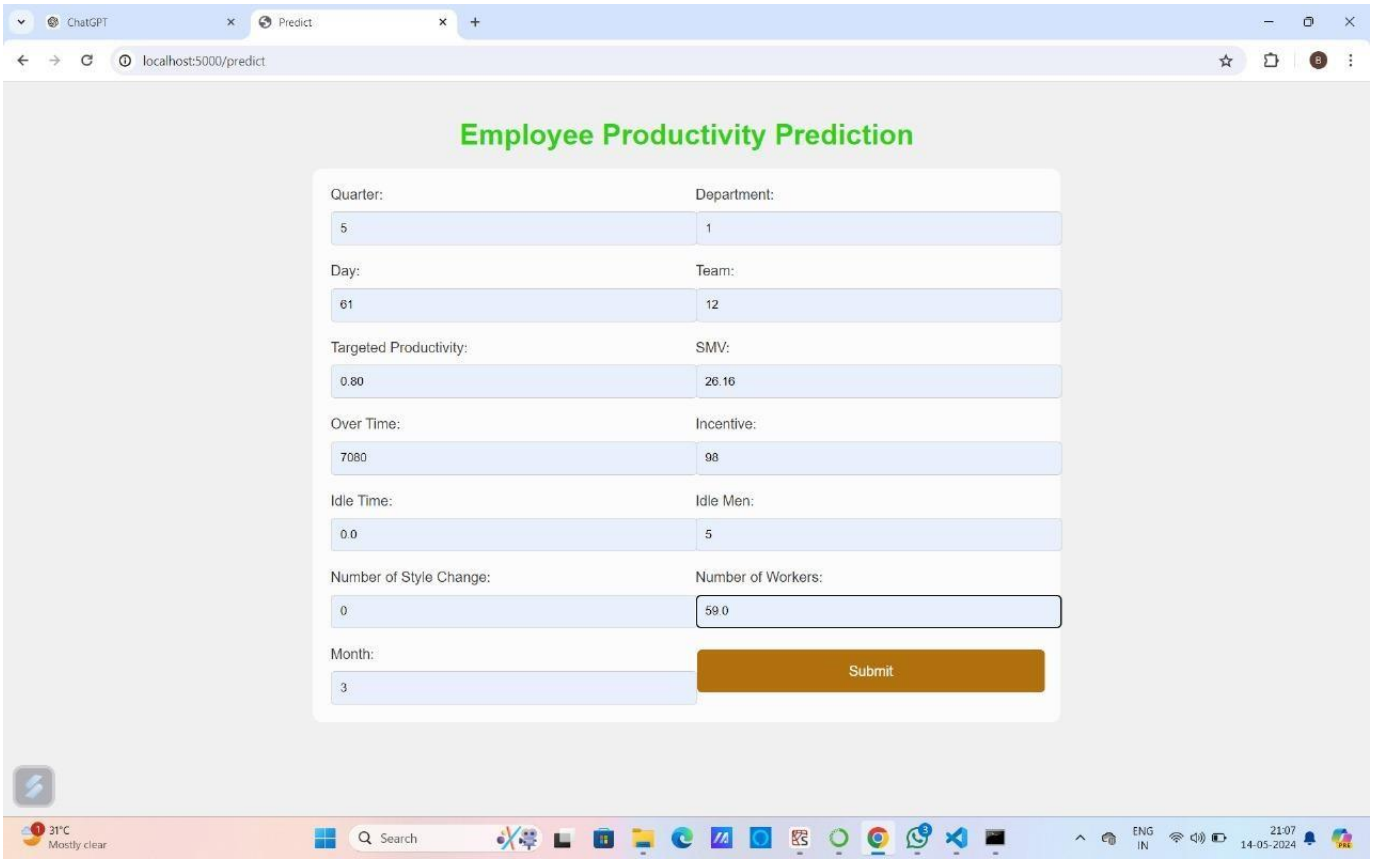
About page:



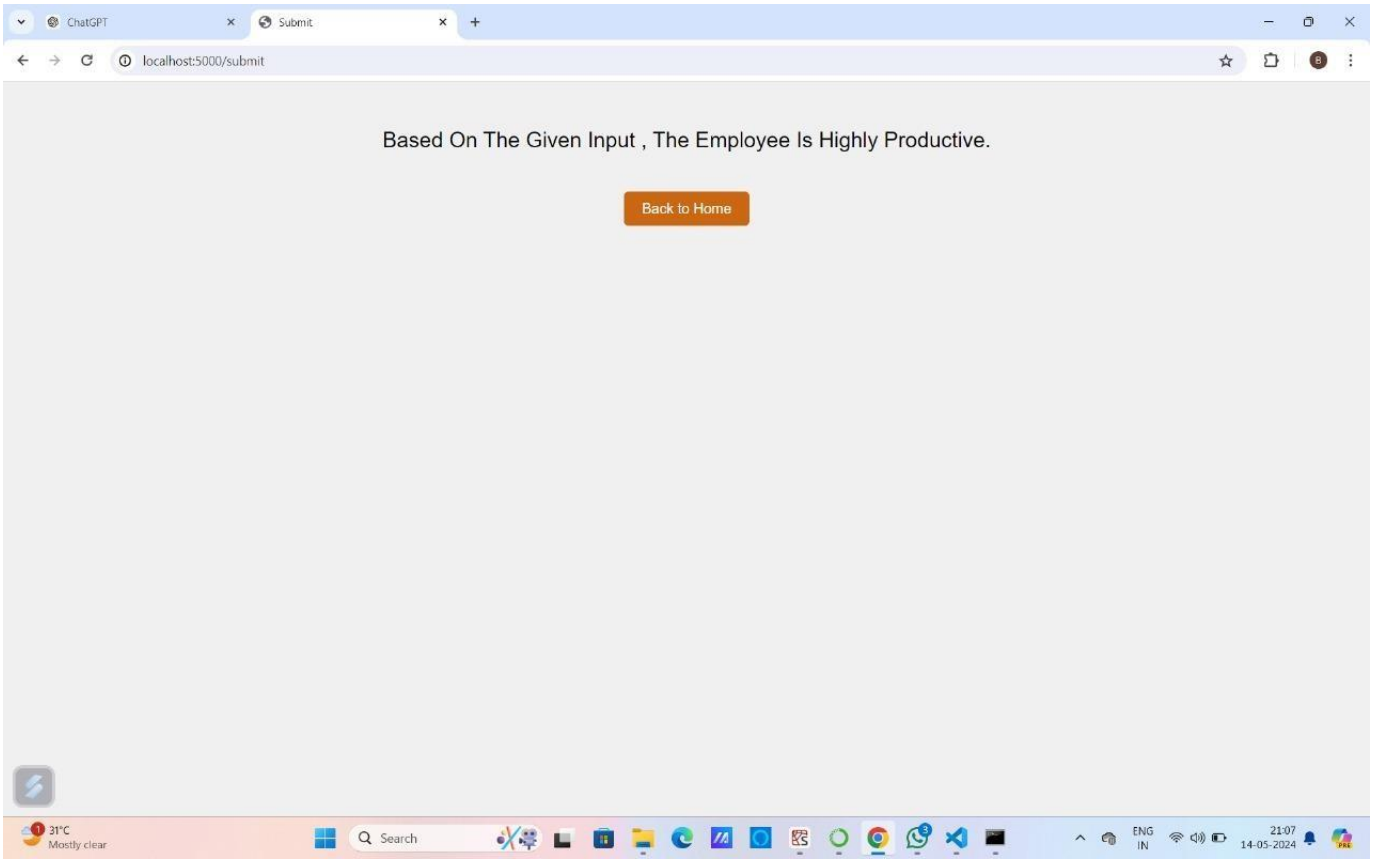
❼When we click on “Back to Home” button which is on the bottom of the content of my web page it will redirects to the home page again.

❼When we click on the “Predict” button which is on the top right of home page of my web page it will redirects to the another page where we can give inputs to our model.

Input 1:


ChatGPT
Predict
localhost:5000/predict
Employee Productivity Prediction
Quarter:
5
Department:
1
Day:
61
Team:
12
Targeted Productivity:
0.80
SMV:
26.16
Over Time:
7080
Incentive:
98
Idle Time:
0.0
Idle Men:
5
Number of Style Change:
0
Number of Workers:
59.0
Month:
3
Submit
31°C
Mostly clear
Search
ENG
IN
21:07
14-05-2024

Output 1:



❼When we click on “Back to Home” button which is on the bottom of the result of my web page it will redirects to the home page again.

8. CONCLUSION

This project analyse and predict the performance of employees in an organization on the basis of various factors, including, but not limited to, individual and domain specific characteristics, nature and level of schooling, socioeconomic status and different psychological factors. The performance is evaluated successfully.

9. FUTURE SCOPE

Provide employees with a better understanding of their role and responsibilities. Increase confidence through recognizing strengths while identifying training needs to improve weaknesses.

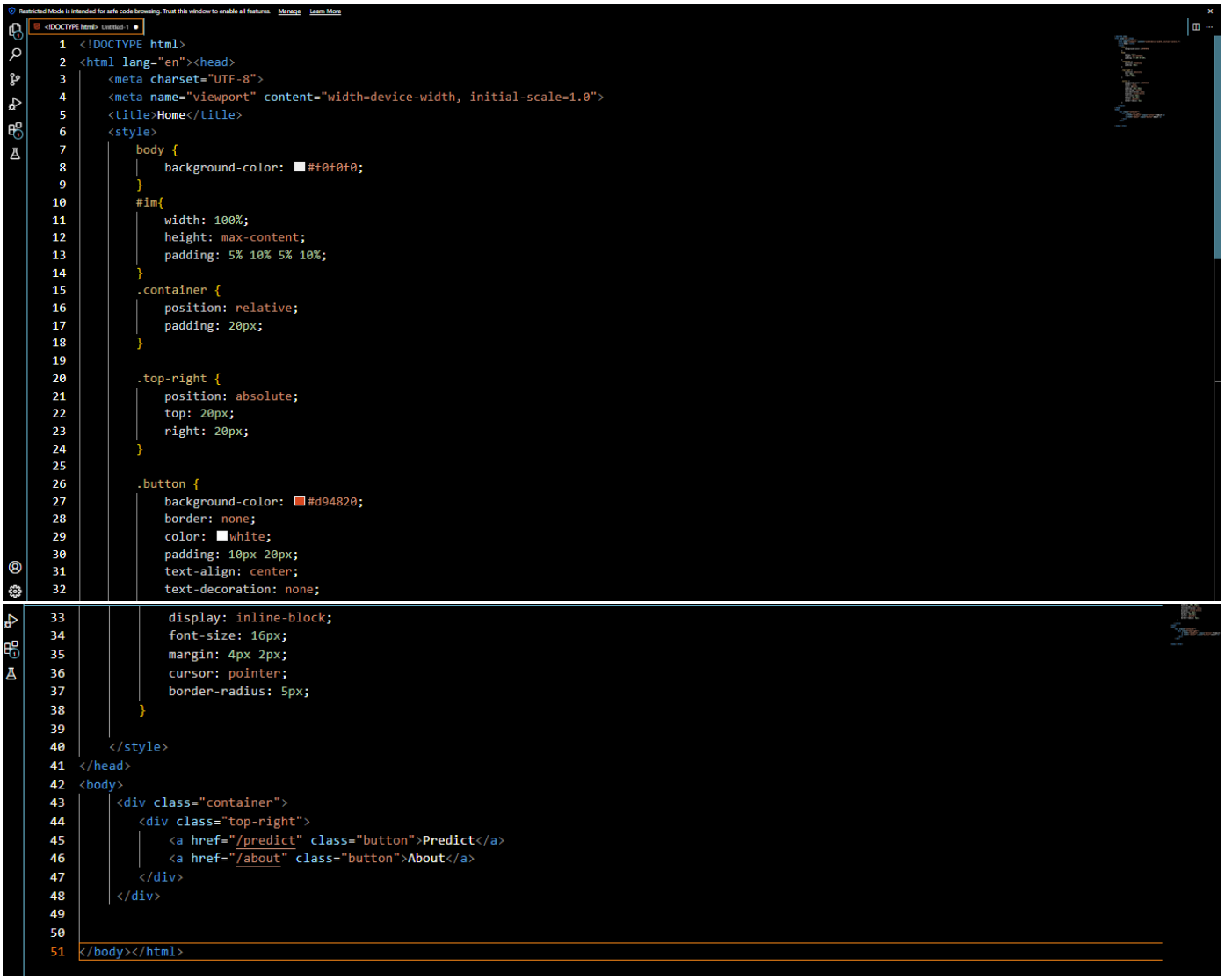
10. APPENDIX 10.1

Source Code home.html

```html
<!DOCTYPE html><html lang="en"><head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Submit</title>
    <style>
        body {
            background-color: #f0f0f0;
            color: white;
            font-family: Arial, sans-serif;
        }

        .container {
            padding: 20px;
            text-align: center;
        }

        h1 {
            margin-top: 50px;
        }

        p {
            font-size: 24px;
            color: black;
        }

        button {
            background-color: #cb6613;
            border: none;
            color: white;
            padding: 10px 20px;
            text-align: center;
            text-decoration: none;
```

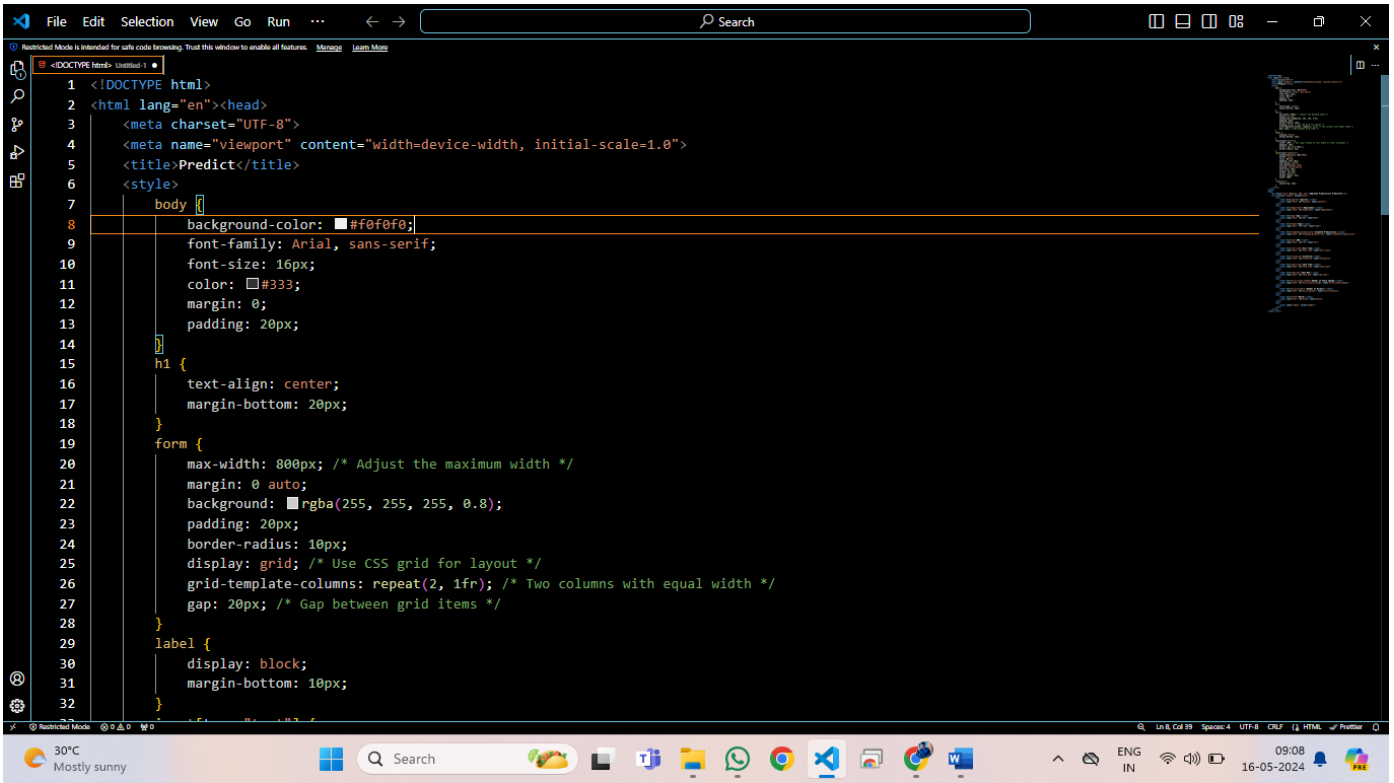
```
            text-decoration: none;
            display: inline-block;
            font-size: 16px;
            margin-top: 20px;
            cursor: pointer;
            border-radius: 5px;
        }
    </style>
</head>
<body>
    <div class="container">
        <p>Based On The Given Input , The Employee Is {{ productivity_level }}.</p>
        <button onclick="window.location.href='/'">Back to Home</button>
    </div>


</body></html>
```

about.html

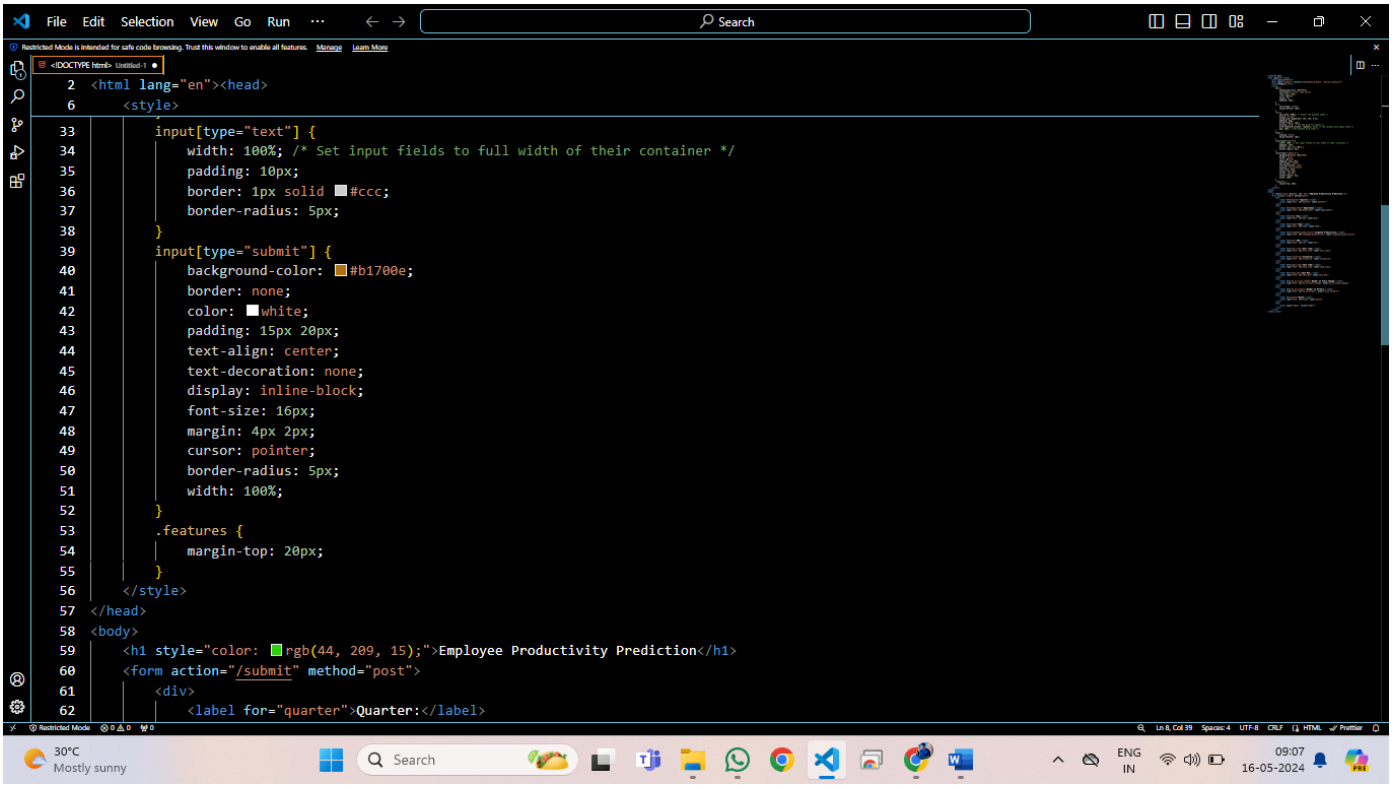


```html
<!DOCTYPE html>
<html lang="en"><head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Home</title>
    <style>
        body {
            background-color: #f0f0f0;
        }
        #im{
            width: 100%;
            height: max-content;
            padding: 5% 10% 5% 10%;
        }
        .container {
            position: relative;
            padding: 20px;
        }

        .top-right {
            position: absolute;
            top: 20px;
            right: 20px;
        }

        .button {
            background-color: #d94820;
            border: none;
            color: white;
            padding: 10px 20px;
            text-align: center;
            text-decoration: none;
            display: inline-block;
            font-size: 16px;
            margin: 4px 2px;
            cursor: pointer;
            border-radius: 5px;
        }

    </style>
</head>
<body>
    <div class="container">
        <div class="top-right">
            <a href="/predict" class="button">Predict</a>
            <a href="/about" class="button">About</a>
        </div>
    </div>


</body></html>
```

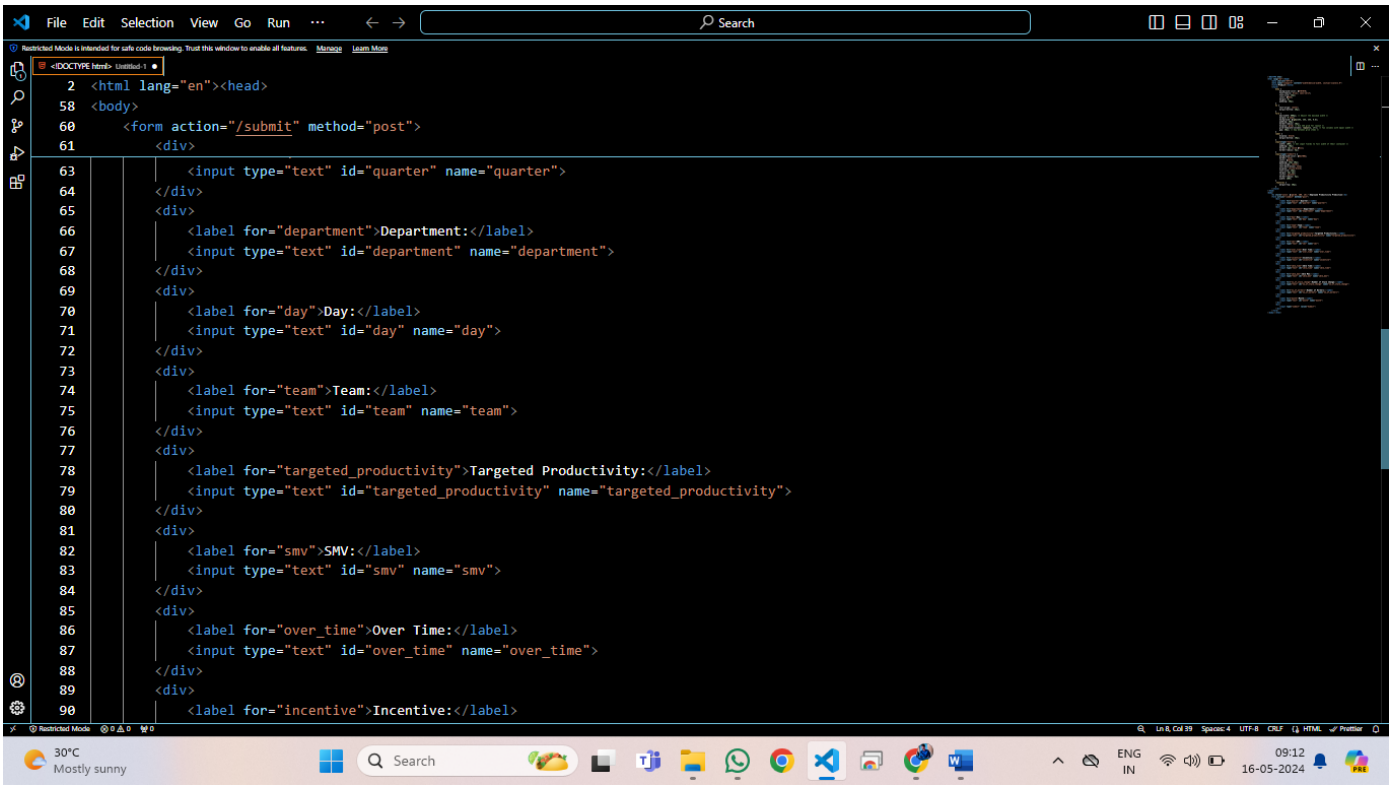
Predict.html



```html
<!DOCTYPE html>
<html lang="en"><head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Predict</title>
    <style>
        body {
            background-color: #f0f0f0;
            font-family: Arial, sans-serif;
            font-size: 16px;
            color: #333;
            margin: 0;
            padding: 20px;
        }
        h1 {
            text-align: center;
            margin-bottom: 20px;
        }
        form {
            max-width: 800px; /* Adjust the maximum width */
            margin: 0 auto;
            background: rgba(255, 255, 255, 0.8);
            padding: 20px;
            border-radius: 10px;
            display: grid; /* Use CSS grid for layout */
            grid-template-columns: repeat(2, 1fr); /* Two columns with equal width */
            gap: 20px; /* Gap between grid items */
        }
        label {
            display: block;
            margin-bottom: 10px;
        }
```



```
<!DOCTYPE html> Untitled-1
<html lang="en"><head>
    <style>
        input[type="text"] {
            width: 100%; /* Set input fields to full width of their container */
            padding: 10px;
            border: 1px solid #ccc;
            border-radius: 5px;
        }
        input[type="submit"] {
            background-color: #b1700e;
            border: none;
            color: white;
            padding: 15px 20px;
            text-align: center;
            text-decoration: none;
            display: inline-block;
            font-size: 16px;
            margin: 4px 2px;
            cursor: pointer;
            border-radius: 5px;
            width: 100%;
        }
        .features {
            margin-top: 20px;
        }
    </style>
</head>
<body>
    <h1 style="color: rgb(44, 209, 15);">Employee Productivity Prediction</h1>
    <form action="/submit" method="post">
        <div>
            <label for="quarter">Quarter:</label>
```



```
<!DOCTYPE html> Untitled-1
<html lang="en"><head>
<body>
    <form action="/submit" method="post">
        <div>
            <input type="text" id="quarter" name="quarter">
        </div>
        <div>
            <label for="department">Department:</label>
            <input type="text" id="department" name="department">
        </div>
        <div>
            <label for="day">Day:</label>
            <input type="text" id="day" name="day">
        </div>
        <div>
            <label for="team">Team:</label>
            <input type="text" id="team" name="team">
        </div>
        <div>
            <label for="targeted_productivity">Targeted Productivity:</label>
            <input type="text" id="targeted_productivity" name="targeted_productivity">
        </div>
        <div>
            <label for="smv">SMV:</label>
            <input type="text" id="smv" name="smv">
        </div>
        <div>
            <label for="over_time">Over Time:</label>
            <input type="text" id="over_time" name="over_time">
        </div>
        <div>
            <label for="incentive">Incentive:</label>
```

Submit.html

```
            <input type="text" id="incentive" name="incentive">
        </div>
        <div>
            <label for="idle_time">Idle Time:</label>
            <input type="text" id="idle_time" name="idle_time">
        </div>
        <div>
            <label for="idle_men">Idle Men:</label>
            <input type="text" id="idle_men" name="idle_men">
        </div>
        <div>
            <label for="no_of_style_change">Number of Style Change:</label>
            <input type="text" id="no_of_style_change" name="no_of_style_change">
        </div>
        <div>
            <label for="no_of_workers">Number of Workers:</label>
            <input type="text" id="no_of_workers" name="no_of_workers">
        </div>
        <div>
            <label for="month">Month:</label>
            <input type="text" id="month" name="month">
        </div>
        <div>
            <input type="submit" value="Submit">
        </div>
    </form>
</body></html>
```

Flask Code for Application Building.py ...

Import the necessary librariesfrom flask import Flask, render_template, requestimport joblibimport numpy as npimport os# Load the saved modelmodel_path = os.path.join("C:/Users/akash/OneDrive/Documents/MchineLearning/my_Externship_files", "garments_model_Final.pkl")final_model = joblib.load(model_path)# Function to predict productivity leveldef predict_productivity(data): prediction = final_model.predict(data) if prediction >= 0.7: result = "Highly Productive" elif prediction >= 0.5: result = "Medium Productive" else: result = "Low Productive" return result# Initialize the Flask applicationapp = Flask(_name, template_folder="templates_testing")# Define routes to render HTML pages@app.route('/')def home(): return render_template('home_t.html')@app.route('/predict')def predict(): return render_template('predict_t.html')@app.route('/submit', methods=['POST'])def pred(): if request.method == 'POST': # Retrieve the values entered by the user features = [float(x) for x in request.form.values()] # Reshape the data to match the model's input shape data = np.array(features).reshape(1, -1) # Get productivity prediction using the predictive system productivity_level = predict_productivity(data) # Render the submit.html page with the predicted productivity level return render_template('submit_t.html', productivity_level=productivity_level) # Define route for about page@app.route('/about')def about(): return render_template('about_t.html')if __name_ == '_main_': # Run the Flask application app.run(debug=True