

Salary prediction MachineLearning project

HTML and CSS code to design my webpage for salary Prediction

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>💼 Predict Your Salary</title>
  <style>
    body {
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
      background-color: #f4f6f8;
      margin: 0;
      padding: 0;
    }

    .container {
      max-width: 500px;
      background-color: #ffffff;
      margin: 60px auto;
      padding: 40px 30px;
      border-radius: 12px;
      box-shadow: 0 8px 24px rgba(0, 0, 0, 0.1);
    }

    h2 {
      text-align: center;
      margin-bottom: 30px;
      color: #333;
    }

    label {
      font-weight: 600;
```

```

        display: block;
        margin-bottom: 5px;
        margin-top: 15px;
    }

    input[type="number"],
    select {
        width: 100%;
        padding: 10px;
        margin-top: 3px;
        border-radius: 6px;
        border: 1px solid #ccc;
        box-sizing: border-box;
    }

    input[type="submit"] {
        margin-top: 25px;
        width: 100%;
        background-color: #007bff;
        color: white;
        padding: 12px;
        border: none;
        border-radius: 6px;
        font-size: 16px;
        font-weight: bold;
        cursor: pointer;
        transition: background-color 0.3s ease;
    }

    input[type="submit"]:hover {
        background-color: #0056b3;
    }

    .footer {
        text-align: center;
        margin-top: 20px;
        font-size: 13px;
        color: #777;
    }
</style>
</head>
<body>
<div class="container">
    <h2>👨‍💻 Predict Your Salary</h2>
    <form action="/predict" method="POST">

        <label for="age">Age:</label>
        <input type="number" name="age" required>

```

```

<label for="gender">Gender:</label>
<select name="gender" required>
  <option value="0">Female</option>
  <option value="1">Male</option>
</select>

<label for="education_level">Education Level:</label>
<select name="education_level" required>
  <option value="0">High School</option>
  <option value="1">Bachelor's</option>
  <option value="2">Master's</option>
  <option value="3">PhD</option>
</select>

<label for="job_title">Job Title:</label>
<select name="job_title" required>
  <option value="0">Data Analyst</option>
  <option value="1">Software Engineer</option>
  <option value="2">Data Scientist</option>
  <option value="3">HR Specialist</option>
  <option value="4">Project Manager</option>
  <option value="5">Other</option>
</select>

<label for="years_of_experience">Years of Experience:</label>
<input type="number" step="0.1" name="years_of_experience" required>

<input type="submit" value="Predict Salary">
</form>
<div class="footer">
  Powered by Machine Learning 💡
</div>
</div>
</body>
</html>

```

PYTHON CODE USING FLASK TO INTEGRATE MY TRAINED MODEL AND WEBPAGE

```

from flask import Flask, render_template, request
import numpy as np

```

```

import joblib

app = Flask(__name__)

# Load model and scaler
model = joblib.load("salary_model.pkl")
scaler = joblib.load("scaler.pkl")

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

@app.route('/predict', methods=['POST'])
def predict():
    try:
        age = float(request.form['age'])
        gender = float(request.form['gender'])
        education_level = float(request.form['education_level'])
        job_title = float(request.form['job_title'])
        years_of_experience = float(request.form['years_of_experience'])

        features = np.array([[age, gender, education_level, job_title, years_of_experience]])
        features_scaled = scaler.transform(features)
        predicted_salary = model.predict(features_scaled)[0]

        return f"<h3> 💰 Predicted Salary: ₹{predicted_salary:,.2f}</h3>"

    except Exception as e:
        return f"<h3 style='color:red;'>Error: {e}</h3>"

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

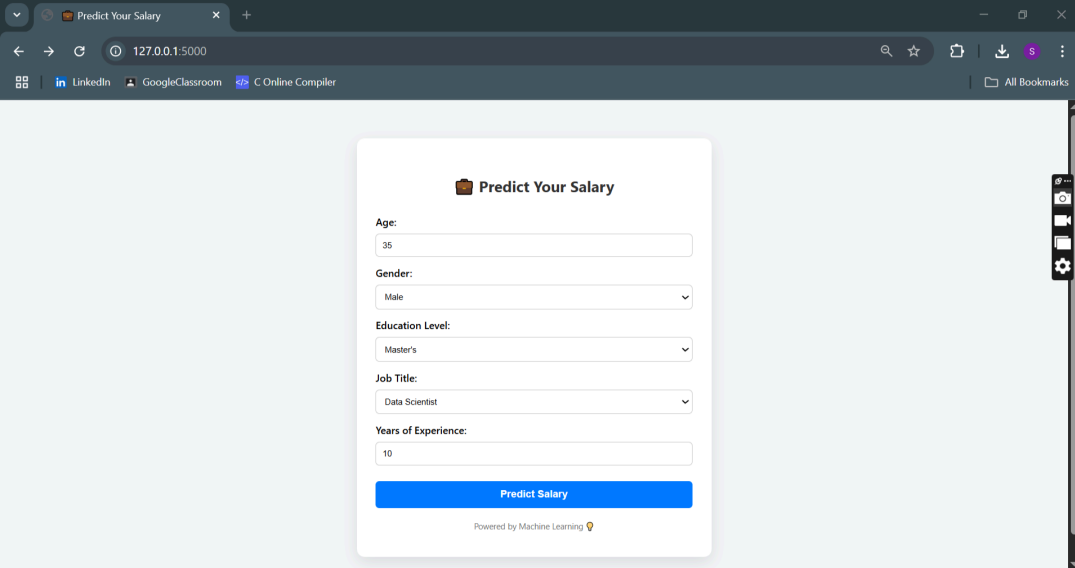
```

COMMAND PROMPT COMMAND TO OPEN MY WEBPAGE

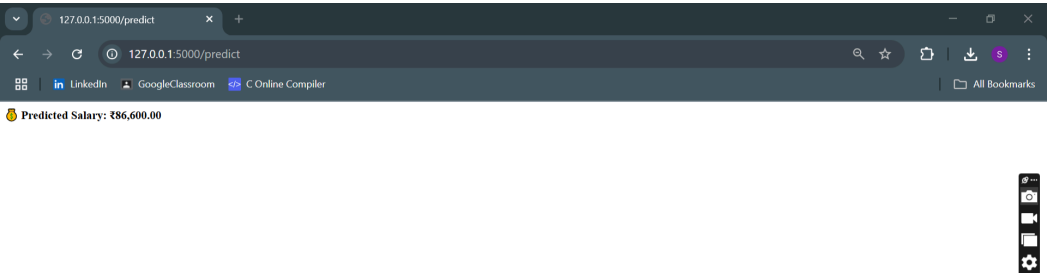
```
C:\Windows\System32\cmd.exe - python salaryapp.py
Microsoft Windows [Version 10.0.19045.5796]
(c) Microsoft Corporation. All rights reserved.

D:\ML models\salary_prediction>python salaryapp.py
* Serving Flask app 'salaryapp'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 640-721-693
127.0.0.1 - - [24/Apr/2025 19:41:26] "GET / HTTP/1.1" 200 -
C:\Users\Dell\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\utils\validation.py:2739: UserWarning: X does not have valid feature names, but StandardScaler was fitted with feature names
  warnings.warn(
127.0.0.1 - - [24/Apr/2025 19:42:01] "POST /predict HTTP/1.1" 200 -
```

SAMPLE 1: web page with input and predicted salary output



The screenshot shows a web browser window with the address bar displaying '127.0.0.1:5000'. The page title is 'Predict Your Salary'. The main content area features a white card with the title 'Predict Your Salary' and a briefcase icon. Below the title are five input fields: 'Age' (text input with value '35'), 'Gender' (dropdown menu with value 'Male'), 'Education Level' (dropdown menu with value 'Master's'), 'Job Title' (dropdown menu with value 'Data Scientist'), and 'Years of Experience' (text input with value '10'). A blue button labeled 'Predict Salary' is positioned below the input fields. At the bottom of the card, it says 'Powered by Machine Learning' with a small location pin icon.



The screenshot shows the same web browser window, but the address bar now displays '127.0.0.1:5000/predict'. The page content is minimal, showing a single line of text: 'Predicted Salary: ₹86,600.00'.

SAMPLE 2: web page with input and predicted salary output

