Week – 4: Deployment on Flask

Name: Ishwarya Rajaiya

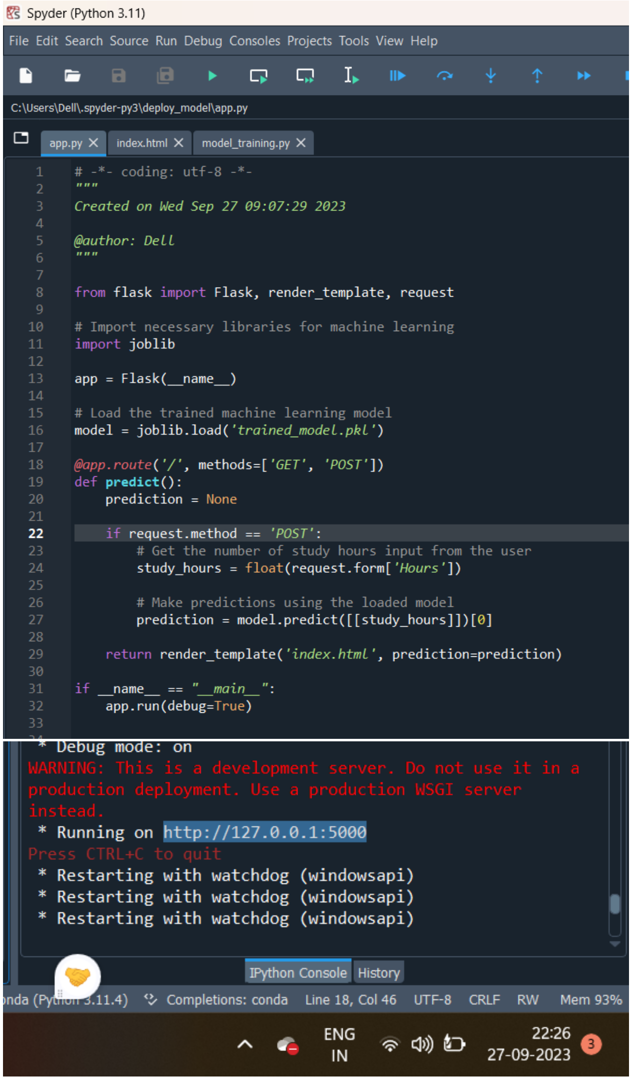
Batch code: LISUM25

Submission date: 27-09-2023

Submitted to: Data Glacier Dashboard

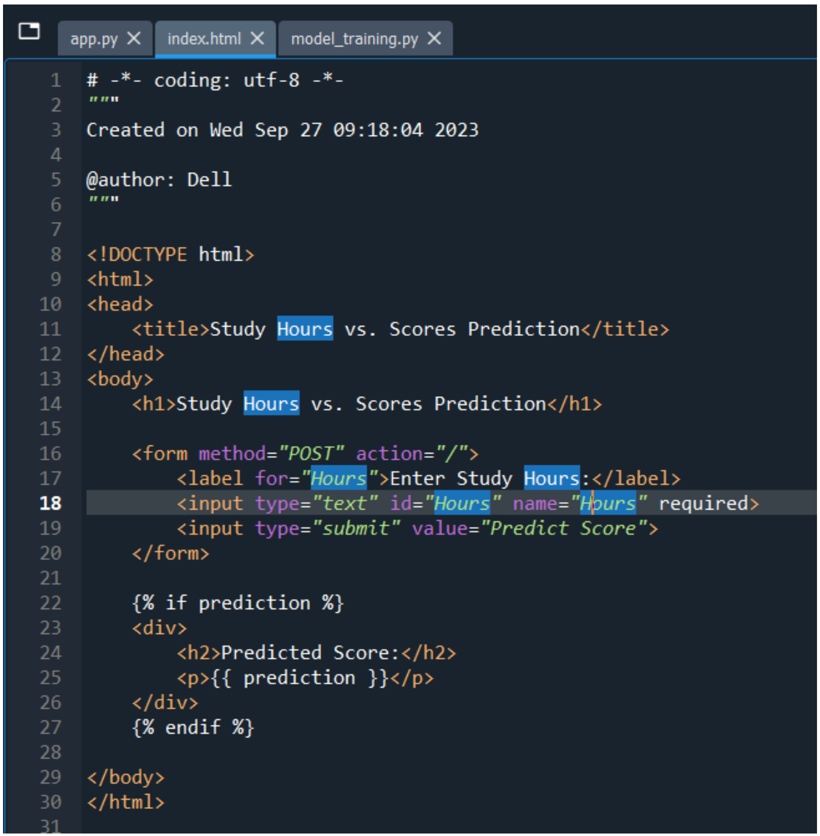
SCREENSHOTS OF MODEL DEPLOYMENT

1. **Code of training the model, Flask application and HTML index.**

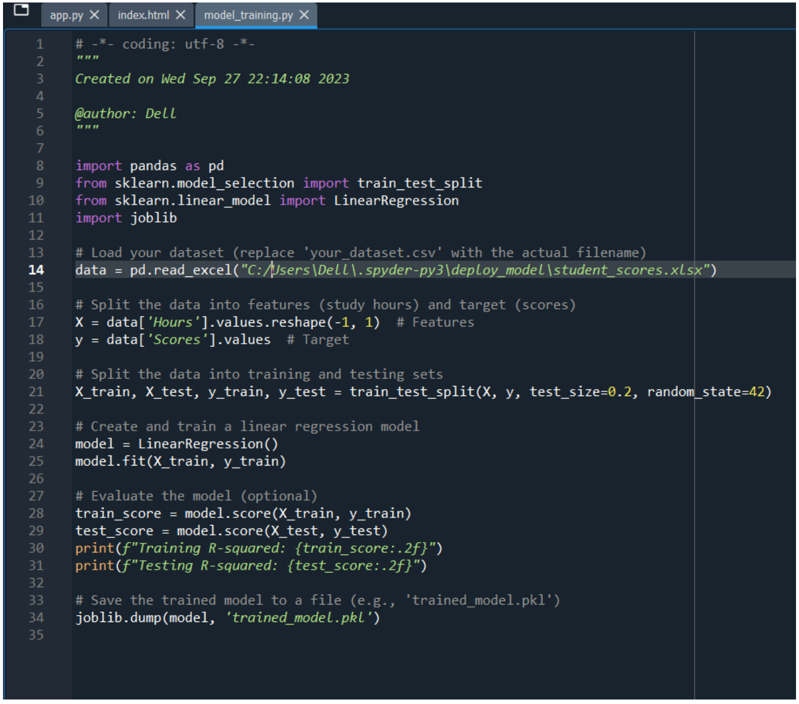


```python
# -*- coding: utf-8 -*-
"""
Created on Wed Sep 27 09:07:29 2023

@author: Dell
"""

from flask import Flask, render_template, request

# Import necessary libraries for machine learning
import joblib

app = Flask(__name__)

# Load the trained machine learning model
model = joblib.load('trained_model.pkl')

@app.route('/', methods=['GET', 'POST'])
def predict():
    prediction = None

    if request.method == 'POST':
        # Get the number of study hours input from the user
        study_hours = float(request.form['Hours'])

        # Make predictions using the loaded model
        prediction = model.predict([[study_hours]])[0]

    return render_template('index.html', prediction=prediction)

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

```
 * 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 watchdog (windowsapi)
 * Restarting with watchdog (windowsapi)
 * Restarting with watchdog (windowsapi)
```

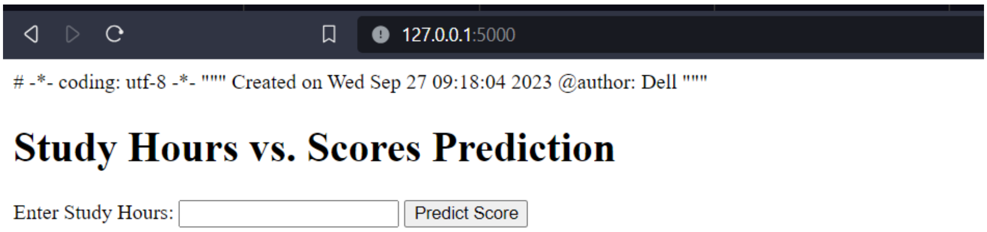


```html
# -*- coding: utf-8 -*-
"""
Created on Wed Sep 27 09:18:04 2023

@author: Dell
"""

<!DOCTYPE html>
<html>
<head>
    <title>Study Hours vs. Scores Prediction</title>
</head>
<body>
    <h1>Study Hours vs. Scores Prediction</h1>

    <form method="POST" action="/">
        <label for="Hours">Enter Study Hours:</label>
        <input type="text" id="Hours" name="Hours" required>
        <input type="submit" value="Predict Score">
    </form>

    {% if prediction %}
    <div>
        <h2>Predicted Score:</h2>
        <p>{{ prediction }}</p>
    </div>
    {% endif %}

</body>
</html>
```



```python
# -*- coding: utf-8 -*-
"""
Created on Wed Sep 27 22:14:08 2023

@author: Dell
"""

import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
import joblib

# Load your dataset (replace 'your_dataset.csv' with the actual filename)
data = pd.read_excel("C:/Users\Dell\.spyder-py3\deploy_model\student_scores.xlsx")

# Split the data into features (study hours) and target (scores)
X = data['Hours'].values.reshape(-1, 1)  # Features
y = data['Scores'].values  # Target

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create and train a linear regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Evaluate the model (optional)
train_score = model.score(X_train, y_train)
test_score = model.score(X_test, y_test)
print(f"Training R-squared: {train_score:.2f}")
print(f"Testing R-squared: {test_score:.2f}")

# Save the trained model to a file (e.g., 'trained_model.pkl')
joblib.dump(model, 'trained_model.pkl')
```

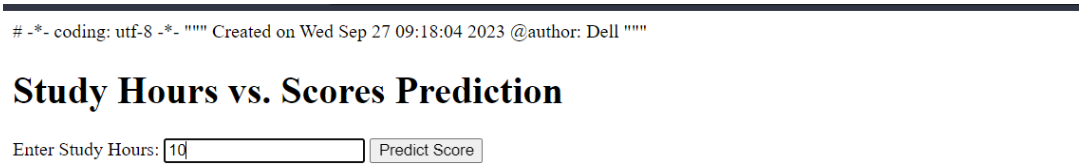
2. Model deployment with predictions:



-*- coding: utf-8 -*- """ Created on Wed Sep 27 09:18:04 2023 @author: Dell """

Study Hours vs. Scores Prediction

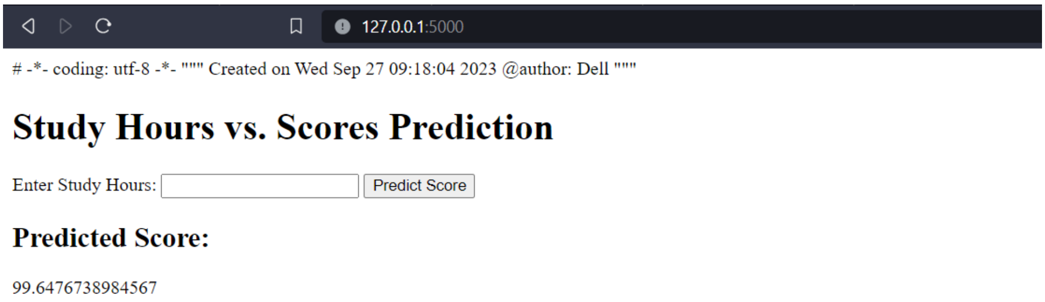
Enter Study Hours:



-*- coding: utf-8 -*- """ Created on Wed Sep 27 09:18:04 2023 @author: Dell """

Study Hours vs. Scores Prediction

Enter Study Hours: 10



-*- coding: utf-8 -*- """ Created on Wed Sep 27 09:18:04 2023 @author: Dell """

Study Hours vs. Scores Prediction

Enter Study Hours:

Predicted Score:

99.6476738984567