

Name: Hiten Chadha  
Batch Code: LISUM10: 30  
Submission Date: 28/06/2022  
Submitted to: DataGlacier

Step 1: Serialization and Deserialization of ML Model to create .pkl file

```
pickle.dump(classifier, open('toy_data_wineclassification.pkl', 'wb'))  
print("Pickling Done")  
  
model = pickle.load(open('toy_data_wineclassification.pkl', 'rb'))  
print("Unpickling Done")
```

Step 2: Creating index.html file for home page (with style.css file for styling)

```
</head>  
  
<body>  
  <div class="login">  
    <h1>Predict Wine Class</h1>  
  
    <!-- Main Input For Receiving Query to our ML -->  
    <form action="{{ url_for('predict')}}"method="post">  
      <input type="text" name="Alcohol" placeholder="Alcohol" required="required" />  
      <input type="text" name="Malic acid" placeholder="Malic acid" required="required" />  
      <input type="text" name="Ash" placeholder="Ash" required="required" />  
      <input type="text" name="Alcalinity of Ash" placeholder="Alcalinity of Ash" required="required" />  
      <input type="text" name="Magnesium" placeholder="Magnesium" required="required" />  
      <input type="text" name="Total phenols" placeholder="Total phenols" required="required" />  
      <input type="text" name="Flavanoids" placeholder="Flavanoids" required="required" />  
      <input type="text" name="Nonflavanoids phenols" placeholder="Nonflavanoids phenols" required="required" />  
      <input type="text" name="Proanthocyanins" placeholder="Proanthocyanins" required="required" />  
      <input type="text" name="Color Intensity" placeholder="Color Intensity" required="required" />  
      <input type="text" name="Hue" placeholder="Hue" required="required" />  
      <input type="text" name="OD280/OD315 of diluted wines" placeholder="OD280/OD315 of diluted wines" required="required" />  
      <input type="text" name="Proline" placeholder="Proline" required="required" />  
  
      <button type="submit" class="btn btn-primary btn-block btn-large">Predict</button>  
    </form>  
  
    <br>  
    <br>  
    {{ prediction_text }}  
  </div>  
  
</body>  
</html>
```

Step 3: Creating the web app using flask\_wineapp.py file and extracting the features from the client and render the result back to the html page

```

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

app = Flask(__name__)

model = pickle.load(open('toy_data_wineclassification.pkl', 'rb'))
print(model)

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

@app.route('/predict', methods = ['POST'])
def predict():
    int_features = [float(x) for x in request.form.values()]
    final_features = [np.array(int_features)]
    print(final_features)
    prediction = model.predict(final_features)

    output = prediction
    #print(output)
    return render_template('index.html', prediction_text='Classified wine is of class {}'.format(output))

if __name__ == '__main__':
    app.run(port=5000, debug=True)

```

Step 4: Run the flask\_wineapp.py file and run the address on the web browser as shown:

## Predict Wine Class

Classified wine is of class [1]