## Rishabh Raman

## Submitted on June 5,2025

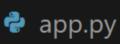
All of the pictures and steps are shown below with code:

- 1. The first photo sets the dataset and the model training
- 2. The second file shows the procfile file.
- 3. The third one shows the requirements file.
- 4. The fourth one trains the model with the data we want and sets the criteria. It was Python code to train a binary classifier to detect Setosa flowers. The model was saved as 'model.pkl'.
- 5. The fifth one shows the heroku website account working.
- 6. The sixth one shows the product actually in github.

SCROLL DOWN!

١

```
арр.ру
🦆 app.py > ...
     from flask import Flask, request, jsonify, render_template
     import joblib
     import numpy as np
     app = Flask(__name__)
     model = joblib.load('model.pkl')
     @app.route('/')
     def home():
         return render_template('index.html')
     @app.route('/predict', methods=['POST'])
     def predict():
         features = [float(x) for x in request.form.values()]
         prediction = model.predict([features])[0]
         result = "Setosa 🌸" if prediction == 1 else "Not Setosa 🜿"
         return render_template('index.html', prediction_text=f"Prediction: {result}")
     @app.route('/predict_api', methods=['POST'])
     def predict_api():
         data = request.get_json(force=True)
         values = np.array(list(data.values())).reshape(1, -1)
         prediction = model.predict(values)
         return jsonify({'prediction': int(prediction[0])})
     if __name__ == "__main__":
         app.run(debug=True)
```



H Procfile



<mark>ዘ</mark> Procfile

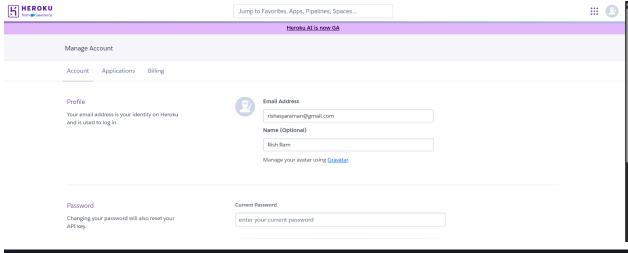
1 web: gunicorn app:app

```
app.py = requirements.txt X

Frequirements.txt

1  flask
2  scikit-learn
3  joblib
4  numpy
5  gunicorn
```

```
train_model.py > ...
    from sklearn.datasets import load_iris
    from sklearn.linear_model import LogisticRegression
    import joblib
    # Load dataset
    iris = load iris()
    X = iris.data
    y = (iris.target == 0).astype(int) # Binary: Setosa or not
L0
    # Train model
    model = LogisticRegression()
11
12
    model.fit(X, y)
L3
L4
    # Save model
    joblib.dump(model, 'model.pkl')
     print("Model saved as model.pkl")
L6
```



Rxr2664 Add files via upload		0f5bfdf · 2 hours ago	1 Commit
Procfile	Add files via upload		2 hours ago
app.py	Add files via upload		2 hours ago
index.html	Add files via upload		2 hours ago
model.pkl	Add files via upload		2 hours ago
requirements.txt	Add files via upload		2 hours ago
🖺 train_model.py	Add files via upload		2 hours ago