**Docker file:**

FROM python:3.9-slim

RUN pip install --no-cache-dir flask joblib scikit-learn

# Copy model and inference code

COPY model /opt/ml/model/

COPY inference.py /opt/ml/code/inference.py

EXPOSE 8080

ENTRYPOINT ["python", "/opt/ml/code/inference.py"]

**Inference.py**

import os

import joblib

from flask import Flask, request, jsonify

app = Flask(\_\_name\_\_)

model = None

def load\_model():

    global model

    model\_path = os.path.join('/opt/ml/model', 'model.pkl')

    model = joblib.load(model\_path)

@app.route('/ping', methods=['GET'])

def ping():

    return '', 200 if model else 404

@app.route('/invocations', methods=['POST'])

def invoke():

    data = request.get\_json()

    inputs = data['inputs']

    predictions = model.predict(inputs)

    return jsonify({'predictions': predictions.tolist()})

if \_\_name\_\_ == '\_\_main\_\_':

    load\_model()

    app.run(host='0.0.0.0', port=8080)

**Model\_train.py**

from sklearn.datasets import load\_iris

from sklearn.ensemble import RandomForestClassifier

import joblib

import os

# Load dataset

iris = load\_iris()

X, y = iris.data, iris.target

# Train model

clf = RandomForestClassifier()

clf.fit(X, y)

# Save model

os.makedirs("model", exist\_ok=True)

joblib.dump(clf, "model/model.pkl")

print("Model trained and saved to model/model.pkl")