# DOC for API:-

from flask import Flask, request, jsonify

import pandas as pd

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

# Load the CSV into memory

DATA\_FILE = 'PurePlates(1).csv'

df = pd.read\_csv(DATA\_FILE)

# Convert column names to clean format

df.columns = [col.strip().replace(" ", "\_").upper() for col in df.columns]

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

def get\_all\_foods():

return jsonify(df.to\_dict(orient='records'))

@app.route('/food/<string:name>', methods=['GET'])

def get\_food(name):

food = df[df['FOOD\_PRODUCT'].str.lower() == name.lower()]

if not food.empty:

return jsonify(food.to\_dict(orient='records')[0])

return jsonify({"message": "Food not found"}), 404

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

def add\_food():

data = request.get\_json()

# STEP 3: Clean incoming data keys

data = {k.strip().replace(" ", "\_").upper(): v for k, v in data.items()}

global df

df = pd.concat([df, pd.DataFrame([data])], ignore\_index=True)

df.to\_csv(DATA\_FILE, index=False)

return jsonify({"message": "Food added successfully"}), 201

@app.route('/food/<string:name>', methods=['PUT'])

def update\_food(name):

global df

index = df[df['FOOD\_PRODUCT'].str.lower() == name.lower()].index

if not index.empty:

for key in request.json:

column\_name = key.strip().replace(" ", "\_").upper()

if column\_name in df.columns:

df.loc[index[0], column\_name] = request.json[key]

else:

return jsonify({"message": f"Invalid field: {column\_name}"}), 400

df.to\_csv(DATA\_FILE, index=False)

return jsonify({"message": "Food updated successfully"})

return jsonify({"message": "Food not found"}), 404

@app.route('/food/<string:name>', methods=['DELETE'])

def delete\_food(name):

global df

index = df[df['FOOD\_PRODUCT'].str.lower() == name.lower()].index

if not index.empty:

df = df.drop(index).reset\_index(drop=True)

df.to\_csv(DATA\_FILE, index=False)

return jsonify({"message": "Food deleted successfully"})

return jsonify({"message": "Food not found"}), 404

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

app.run(debug=True)