#This code performes CRUD(create, return , update , delete) operations in a MongoDB.

from flask import Flask, jsonify, request, render\_template

from flask\_pymongo import PyMongo

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

app.config['MONGO\_DBNAME'] = 'cafe'

#app.config['MONGO\_URI'] = 'mongodb://username:password@hostname:port/databasename'

app.config['MONGO\_URI'] = 'mongodb://kashyap:kashyap@ds125146.mlab.com:25146/cafe'

mongo = PyMongo(app)

#---------------------------------------------------------------------------------------------------------------------

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

def add():

cafe = mongo.db.cafe

cafe.insert([

{ "number" : "1", "item": "Espresso", "price": "2.49$", "cal": 205, "size": "Small", "count" : 24 },

{ "number" : "2", "item": "Espresso", "price": "3.99$", "cal": 299, "size": "Medium", "count" : 22},

{ "number" : "3", "item": "Espresso", "price": "5.99$", "cal": 319, "size": "Large", "count" : 15},

{ "number" : "4", "item": "Cappuccino", "price": "2.99$", "cal": 215, "size": "Small", "count" : 12 },

{ "number" : "5", "item": "Cappuccino", "price": "4.49$", "cal": 335, "size": "Medium", "count" : 19},

{ "number" : "6", "item": "Cappuccino", "price": "5.49$", "cal": 465, "size": "Large", "count" : 28},

{ "number" : "7", "item": "Latte", "price": "2.49$", "cal": 205, "size": "Small", "count" : 33 },

{ "number" : "8", "item": "Latte", "price": "3.99$", "cal": 275, "size": "Medium", "count" : 25},

{ "number" : "9", "item": "Latte", "price": "4.99$", "cal": 378, "size": "Large", "count" : 28},

{ "number" : "10", "item": "Americano", "price": "2.69$", "cal": 302, "size": "Small", "count" : 35 },

{ "number" : "11", "item": "Americano", "price": "3.99$", "cal": 403, "size": "Medium", "count" : 30},

{ "number" : "12", "item": "Americano", "price": "4.49$", "cal": 515, "size": "Large", "count" : 25},

{ "number" : "13", "item": "Hot Chocolate", "price": "3$", "cal": 317, "size": "Small", "count" : 50 },

{ "number" : "14", "item": "Hot Chocolate", "price": "4$", "cal": 418, "size": "Medium", "count" : 40},

{ "number" : "15", "item": "Hot Chocolate", "price": "5$", "cal": 521, "size": "Large", "count" : 35},

{ "number" : "16", "item": "Tea", "price": "1.69$", "cal": 105, "size": "Small", "count" : 60 },

{ "number" : "17", "item": "Tea", "price": "2.49$", "cal": 195, "size": "Medium", "count" : 55},

{ "number" : "18", "item": "Tea", "price": "3.19$", "cal": 245, "size": "Large", "count" : 55},

{ "number" : "19", "item": "Black Tea", "price": "3.49$", "cal": 245, "size": "Small", "count" : 44 },

{ "number" : "20", "item": "Black Tea", "price": "4.69$", "cal": 345, "size": "Medium", "count" : 44},

{ "number" : "21", "item": "Black Tea", "price": "5.15$", "cal": 395, "size": "Large", "count" : 42},

{ "number" : "22", "item": "Iced Tea", "price": "4.49$", "cal": 445, "size": "Medium", "count" : 50 },

{ "number" : "23", "item": "Soda", "price": "2.99$", "cal": 317, "size": "Medium", "count" : 80},

])

return "Value added"

#---------------------------------------------------------------------------------------------------------------------

#Required fields for different operations

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

def operation():

# return "\n Required fields for operations: 1. Insert: all fields 2. Fetch single record: item,size 3. Delete: item,size "

return render\_template('index.html')

#---------------------------------------------------------------------------------------------------------------------

#To read all the records in the database perform "Enquire" operation

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

def get\_all\_records\_admin():

cafe = mongo.db.cafe

output = []

for q in cafe.find():

output.append({'item' : q['item'], 'price' : q['price'], 'cal' : q['cal'], 'size' : q['size'], 'number' : q['number'], 'count' : q['count']})

return jsonify({'result' : output})

#---------------------------------------------------------------------------------------------------------------------

#To read all the records in the database perform "Enquire" operation

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

def get\_all\_records\_staff():

cafe = mongo.db.cafe

output = []

for q in cafe.find():

output.append({'item' : q['item'], 'price' : q['price'], 'cal' : q['cal'], 'size' : q['size'], 'number' : q['number']})

return jsonify({'result' : output})

#---------------------------------------------------------------------------------------------------------------------

#To read only single record from database

@app.route('/enquire/admin/<number>', methods=['GET'])

def get\_one\_record\_admin(number):

cafe = mongo.db.cafe

q = cafe.find\_one({'number' : number})

if q:

output = {'item' : q['item'], 'price' : q['price'], 'cal' : q['cal'], 'size' : q['size'], 'number' : q['number'], 'count' : q['count']}

else:

output = 'No results found'

return jsonify({'result' : output})

#---------------------------------------------------------------------------------------------------------------------

#To read only single record from database

@app.route('/enquire/staff/<number>', methods=['GET'])

def get\_one\_record\_staff(number):

cafe = mongo.db.cafe

q = cafe.find\_one({'number' : number})

if q:

output = {'item' : q['item'], 'price' : q['price'], 'cal' : q['cal'], 'size' : q['size'], 'number' : q['number']}

else:

output = 'No results found'

return jsonify({'result' : output})

#---------------------------------------------------------------------------------------------------------------------

#To "Insert" a new record in the database

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

def insert\_record\_admin():

cafe = mongo.db.cafe

item = request.json['item']

price = request.json['price']

cal = request.json['cal']

size = request.json['size']

number = request.json['number']

count = request.json['count']

q = cafe.find\_one({'number' : number})

if q:

framework\_id = cafe.update\_one({'number' : number}, {'$set' : {'count' : count}})

return "Record Inserted"

else:

cafe\_id = cafe.insert({'item' : item, 'price' : price, 'cal' : cal, 'size' : size, 'number' : number, 'count' : count})

new\_cafe = cafe.find\_one({'\_id' : cafe\_id})

output = {'item' : new\_cafe['item'], 'price' : new\_cafe['price'], 'cal' : new\_cafe['cal'], 'size' : new\_cafe['size'], 'number' : new\_cafe['number'], 'count' : q['count']}

return jsonify({'result' : output})

#---------------------------------------------------------------------------------------------------------------------

#To "Insert" a new record in the database

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

def insert\_record\_user():

cafe = mongo.db.cafe

item = request.json['item']

price = request.json['price']

cal = request.json['cal']

size = request.json['size']

number = request.json['number']

q = cafe.find\_one({'number' : number})

if q:

new\_count = int(q['count']) + 1

framework\_id = cafe.update\_one({'number' : number}, {'$set' : {'count' : new\_count}})

return "Record Inserted"

else:

new\_count = 1

cafe\_id = cafe.insert({'item' : item, 'price' : price, 'cal' : cal, 'size' : size, 'number' : number, 'count' : new\_count})

new\_cafe = cafe.find\_one({'\_id' : cafe\_id})

output = {'item' : new\_cafe['item'], 'price' : new\_cafe['price'], 'cal' : new\_cafe['cal'], 'size' : new\_cafe['size'], 'number' : new\_cafe['number'], 'count' : q['count']}

return jsonify({'result' : output})

#---------------------------------------------------------------------------------------------------------------------

#To "Remove" a record from the database using "number"

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

def remove\_record():

cafe = mongo.db.cafe

number = request.json['number']

q = cafe.find\_one({'number' : number})

if q:

if int(q['count']) == 1:

result = cafe.remove(q)

else:

result = cafe.update\_one({'number' : number}, {"$inc" : {'count' : - 1}})

else:

return "No Record Found"

return "Record Deleted"

#---------------------------------------------------------------------------------------------------------------------

#To "Update" the count of an item in the database

@app.route('/update/admin/count', methods=['POST'])

def update\_record\_admin():

cafe = mongo.db.cafe

number = request.json['number']

count = request.json['count']

q = cafe.find\_one({'number' : number})

if q:

result = cafe.update\_one({'number' : number}, {"$set" : {'count' : count}})

else:

return "No Record Found"

return "Record Updated"

#---------------------------------------------------------------------------------------------------------------------

#To "Update" the price of an item in the database

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

def update\_record():

cafe = mongo.db.cafe

price = request.json['price']

number = request.json['number']

q = cafe.find\_one({'number' : number})

if q:

result = cafe.update\_one({'number' : number}, {"$set" : {'price' : price}})

else:

return "No Record Found"

return "Record Updated"

#---------------------------------------------------------------------------------------------------------------------

#To "Remove" all the records from the database

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

def remove\_all\_records():

cafe = mongo.db.cafe

result = cafe.remove()

return "All the Records are Deleted"

#---------------------------------------------------------------------------------------------------------------------

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

app.run(debug=True)