import json  
import time  
from random import random  
from flask import Flask, request, jsonify, abort, make\_response  
from flask\_cors import CORS  
from user import User  
# export FLASK\_ENV=development  
# flask run  
app = Flask(\_\_name\_\_) # יצירת מופע מהשרת  
CORS(app, supports\_credentials=True) # supports\_credentials=True- מאפשר עוגיות  
users=['']  
  
# יצירת עוגיה למשתמש חדש ל10 דקות  
# @app.route('/set\_cookie', methods=['POST'])  
# def set\_cookie\_func():  
# obj = request.json  
# response = make\_response("Cookie set!")  
# # שם המפתח בעוגייה, הערך, אורך החיים שלה, מאופשרת רק דרך פרוטוקול HTTP או HTTS, לא דורשת HTTPS ורק בקשות מאותו אתר  
# response.set\_cookie("name", obj['user\_name'], max\_age=10, httponly=True, secure=False, samesite='None')  
# return response  
  
@app.route('/say\_hello/<name>', methods=['GET'])  
def hello\_func(name):  
 # מתודה שמקבלת פרמטר דרך הניתוב ומחזירה שלום ל...  
  
 return f"Hello {name}!"  
  
@app.route('/getWord',methods=['POST'])  
def getWord():  
 # פונקציה שמקבלת מספר  
 obj = request.json  
 number=obj['number']  
 content = ''  
 with open('Word.txt', 'r', encoding='utf-8') as file:  
 content = file.readlines()  
 cleaned\_words = [word.strip() for word in content]  
 print(cleaned\_words)  
 # random.shuffle(cleaned\_words) # ערבוב רנדומלי של המילים בטקסט  
 length = len(cleaned\_words) # שמירת מספר המילים הקיימות במשתנה  
 number = (number - 1) % length  
 theWord = cleaned\_words[number]  
 return theWord  
  
@app.route('/login', methods=['POST'])  
def login():  
 # פונקציה של התחברות  
 data = request.json  
 name = data.get('name')  
 password = data.get('password')  
 u = getAllUsers()  
 # for player in u:  
 # if player['name'] == name and player['password'] == password:  
 # newTime=time.time()  
 # newTime = (newTime // 60) % 60  
 # userTime=player['timeU']  
 # userTime=(userTime//60)%60  
 # # אם עברו יותר מ-10 דקות, נעדכן את זמן התחברות המשתמש  
 # if newTime - userTime < 10:  
 # return "no"  
 # return "You can play!!!😉😃😍"  
 # אם המשתמש לא נמצא, נבצע רישום  
 new\_user = User(name,password,len(u)+1,time.time())  
 # כדי להוסיף את השחקן החדש נמיר אותו לאובייקט שמוכל בגייסון  
 u.append(new\_user.to\_dict())  
 with open('./data.json', 'w') as file:  
 json.dump(u, file)  
 # כעת ניצור עוגיה למשתמש החדש  
 response = make\_response("Welcome!!!")  
 response.set\_cookie("user",name,max\_age=600,httponly=True,secure=False,samesite='None')  
 return response  
# פונקציה שמעדכנת לפי קוד את השעה החדשה  
# @app.route('/cookies/<password>')  
# def cookies(password):  
# u = getAllUsers()  
# # נחפש את השחקן הנוכחי  
# for player in u:  
# if player['password'] == password:  
# # נוסיף עוד ניצחון  
# player['timeU'] =time.time()  
# # נעדכן בקובץ הגיסון  
# with open('data.json', 'w') as file:  
# json.dump(u, file)  
@app.route('/win',methods=['POST'])  
def win():  
 # פונקציה שמוסיפה ניצחון  
 data = request.json  
 word = data.get('word')  
 password = data.get('password')  
 u = getAllUsers()  
 # נחפש את השחקן הנוכחי  
 for player in u:  
 if player['password'] == password:  
 #נוסיף עוד ניצחון  
 player['numWin']=player['numWin']+1  
 # נוסיף את המילה למערך המילים  
 w=player['word']  
 w.append(word)  
 # נעדכן בקובץ הגיסון  
 with open('data.json', 'w') as file:  
 json.dump(u, file)  
def getAllUsers():  
 # פונקציה ששולפת את הנתונים מהגיסון ומחזירה אותם  
 # למנוע קוד כפול  
 try:  
 with open('./data.json', 'r') as file:  
 u = json.load(file)  
 except FileNotFoundError:  
 u = []  
 except json.JSONDecodeError:  
 u = []  
 return u  
  
  
@app.route('/set\_cookie', methods=['POST'])  
def set\_cookie\_func():  
 obj = request.json  
 response = make\_response("Cookie set!")  
 # שם המפתח בעוגייה, הערך, אורך החיים שלה, מאופשרת רק דרך פרוטוקול HTTP או HTTS, לא דורשת HTTPS ורק בקשות מאותו אתר  
 response.set\_cookie("user", obj['password'], max\_age=5, httponly=True, secure=False, samesite='None')  
 return response  
  
  
@app.route('/get\_cookie', methods=['GET'])  
def get\_cookie\_func():  
 user\_name = request.cookies.get('user')  
 if user\_name:  
 return make\_response("Cookie found!")  
 return make\_response("Cookie not found")  
  
@app.route('/doCookies',methods=['POST'])  
def c():  
 obj = request.json  
 response = make\_response(f"Hello {obj['name']}! you have another cookies")  
 response.set\_cookie('timeU', str(time.time())) # עדכון זמן התחברות בעוגיה  
 return response  
  
#  
# def write():  
# with open('./data.json', 'w') as file:  
# json.dump(u, file)  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 app.run(debug=True)