|  |
| --- |
| ROBERT SAMUEL NDIVE |
| **FINAL PROJECT: FLATFILE DATABASES** |

DESCRIPTION  
  
YOU HAVE BEEN ASSIGNED TO BUILD A SIMPLE COMMAND LINE REGISTRATION TOOL THAT WORKS WITH A REGULAR FLATFILE DATABASE. YOU ARE EXPECTED TO SHOW UNDERSTANDING USING PYTHON FUNCTIONS, FILE HANDLING, DICTIONARIES, IMPORTS/MODULES IN PYTHON INCLUDING JSON. IN THIS PROJECT, YOU ARE TO SIMULATE THE PROCESS OF A SCHOOL NEEDING A SAMPLE DATABASE TO HOLD RECORD FOR THEIR STUDENTS. WHEN THE PROGRAM IS OPENED THE FIRST TIME, THE ARE REQUESTED TO CREATE A SCHOOL ACCOUNT.   
AFTER LOGIN IN, THEY NOW HAVE ACCESS TO CREATE STUDENTS. AND THEY CAN ADD STUDENTS, DELETE STUDENTS OR EDIT STUDENTS. ALL THE DATA WILL BE STORED IN A JSON FILE. THIS JSON FILE WILL ACT AS YOUR FLATFILE DATABASE.  
  
  
INSTRUCTIONS  
  
  
1. EVERYTHING WILL BE DONE INSIDE OF ONE PYTHON FILE CALLED database\_system.py  
  
2. YOU ARE ALLOWED TO USE ONLY TWO PYTHON MODULES NAMELY import os and import json  
  
3. YOUR PYTHON PROGRAM CAN HAVE AS MANY FUNCTIONS AS POSSIBLE BUT THERE MUST EXIST A MAIN ENTRY FUNCTION CALLED   
  
   
  
  
4. SPLIT KEY FEATURES IN SEPARATE FUNCTIONS, LIKE ADDING DATA TO A DATABASE, EDITING, EXISTING, OR DELETING DATA.  
  
5. FORMAT YOUR PRINT STATEMENTS SO OUTPUT MESSAGES LOOK CLEAN AND READABLE. MAKE USE OF COMMENTS AS YOU WILL REFER BACK TO THIS CODE LATER  
  
6. IN THE EVENT YOU WANT TO USE A DIFFERENT PYTHON MODULE THAN THE ONES INDICATED,   
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
CONTACT MR TAMBE AS SOON AS POSSIBLE  
  
7. YOU WILL WRITE AND READ DATA FROM TWO MAIN FILES   
 accounts.json AND student\_[school\_name].json.   
THE accounts.json FILE WILL STORE THE ACCOUNT DETAILS OF THE SCHOOL ONLY.   
THE students.json FILE WILL STORE THE STUDENT DETAILS OF A PARTICULAR SCHOOL  
  
8. USERS WILL HAVE 3 TIMES MAX TO TRY TO LOGIN. IF THEY FAIL, YOU ASK THE USER TO ENTER A KEY TO EXIT THE PROGRAM.  
  
9. YOU CAN USE NUMBERS OR /COMMANDS TO KNOW WHICH ACTION THE USER WANTS TO CARRY OUT. YOU MUST PROVIDE AN OPTION FOR THE USER TO QUIT THE PROGRAM.  
  
10. USE WELL-FORMATTED AND HELPFUL MESSAGES TO THE USER. ACTIONS LIKE CREATING/MODIFYING/DELETING A STUDENT RECORD SHOULD SHOW MESSAGES IF THEY WERE CARRIED OUT SUCCESSFULLY  
  
FURTHER INSTRUCTIONS  
  
1. ENSURE TO DOCUMENT YOUR RESEARCH PROCESS INSIDE OF YOUR FINAL VENOMRAIDERS   
 SUMMARY REPORT. ALL THE VIDEOS/LINKS/ARTICLES/BOOKS YOU USED IF ANY. THIS STEP IS   
 MANDATORY  
  
2. YOU WILL CREATE A FOLDER AND NAME IT 9\_SATRA. THAT IS WHERE YOUR DATABASE FILES   
 AND PYTHON FILE WILL BE  
  
3. DO NOT COPY CODE FROM THE INTERNET TO SUBMIT FOR THIS PROJECT  
  
4. YOU WORK ON THIS PROJECT INDEPENDENTLY. IF YOU HAVE QUESTIONS, YOU CONTACT MR   
 TAMBE.  
  
5. INCOMPLETE WORK WILL LEAD TO YOU FAILING THE FINAL PROJECT OF VENOMRAIDERS  
  
7. LATE SUBMISSIONS WILL NOT BE ACCEPTED  
  
8. ALL QUESTIONS REGARDING THIS PROJECT SHOULD BE POSTED IN THE TELEGRAM GROUP ONLY.   
  
9. WITH THIS PROJECT, YOU ARE NOT REQUIRED TO ADD DATA VALIDATION LIKE CHECKING IF   
 AN EMAIL FORMAT IS CORRECT BUT YOU ARE REQUIRED TO VALIDATE USERNAMES AND   
 PASSWORDS.   
 You are expected to figure the data types needed like Booleans, strings etc…  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
10. BELOW ARE THE FIELDS/DATA/INFO YOU NEED TO COLLECT FROM THE USER  
  
  
  
  
ACCOUNT DATABASE STRUCTURE = {full\_name, school\_name, username, password}  
  
SCHOOL DATABASE STRUCTURE = {first\_name, last\_name, date\_of\_birth, age(auto   
 calculated field), class, is\_disabled, has\_paid\_fees}  
  
SUGGESTED FUNCTION NAMES = main(), edit\_record(), delete\_record(), add\_record(),   
 check\_account()  
  
  
  
  
  
  
  
  
  
  
  


def main():  
 //call other function here and code here

DATE ISSUED: Tues 8|13|2024 DUE: Wed 8|28|2024