Task 1: Quiz App

Develop a Python-based quiz software to facilitate online quizzes for students at UIU during online classes. This system will allow students to complete quizzes for their trimester efficiently.

Requirements:

Student Identification:

- As the application starts, prompt the student to enter their ID.

Quiz Questions File:

- Prepare a text file named ***questions.txt*** containing 10 Multiple Choice Questions (MCQs).
- Each question should have four options and a designated correct answer.
- Format the file in a way that is easy to parse and read programmatically.

Reading Quiz Questions:

- Implement functionality to read questions and their respective options from the *questions.txt* file.

Quiz Interface:

- Display questions one by one to the student after they enter their ID.
- Allow the student to select an answer for each question.

Answer Evaluation:

- The system should automatically evaluate the student's answers.
- Assign 2 marks for each correct answer.

Saving Results:

- After the quiz is completed, save the student's total marks alongside their ID in another text file called ***marks.txt***.
- If the student attempts the quiz again and scores a mark greater than the one obtained before, update their marks in the *marks.txt* file.

Task 2: Currency Conversion

Develop a Python application that converts an amount from one currency to another.

Requirements:

User Input:

- The application should prompt the user to enter an amount in their local currency (for example, BDT).

Currency Selection:

- The user should then select the currency they want to convert to (for example, EUR, GBP, JPY, etc.).
- Provide a list of at least 3-4 different currencies for the user to choose from.

Conversion Rates:

- Create a file named ***conversion_rates.txt*** that contains the conversion rates from the local currency to the selected currencies. The file should have pre-defined conversion rates in a readable format.

Reading Conversion Rates:

- Your program must read the conversion rates from the *conversion_rates.txt* file.

Perform Conversion:

- The program should calculate the converted amount based on the user's choice and the amount entered.
- Display the converted amount in the chosen currency.

Simple Interface:

- The program should have a clear and user-friendly interface, with instructions displayed for the user.

Task 3: Phone Book

Develop a Python application to create and manage a contact book.

Requirements:

User Interface:

- The application should offer a menu with options to
    1. Add a new contact
    2. View all contacts
    3. Search for a contact, and
    4. Exit the program.

Contact Information:

- Each contact should have a name and phone number.

Adding Contacts:

- When adding a new contact, the program should prompt the user to enter the contact's name and phone number.
- Store this information in a data structure (like a dictionary or a list of dictionaries or whatever you think is best for this scenario).

Viewing Contacts:

- Implement a feature to display all saved contacts in a readable format.

Searching Contacts:

- Allow the user to search for a contact by name. Display the contact's details if found.

File Handling:

- Save the contacts to a file named ***contacts.txt***.

Task 4: Movie Rating Tracker

You are tasked with creating a movie ratings tracker app. The program should allow users to view, add, and update movie ratings.

Requirements:

User Interface:

- The application should offer a menu with options to
    1. View all movies and their ratings
    2. Add a new movie along with a rating
    5. Update an existing movie rating, and
    6. Exit the program.

Move Information:

- Each movie should have a name and a rating (out of 10).

Adding Movie Ratings:

- When adding a new movie, the program should prompt the user to enter the movie's name and its rating.

Viewing Movie Ratings:

- Implement a feature to display all movies and their corresponding rating.

Updating Movie Ratings:

- If the user chooses the “update” option, they will shown a list of all existing movies. The user will also be prompted to enter the name of the movie from that list and enter a new rating.
- Update the movie’s rating in the text file that stores all the information about the movie.

File Handling:

- Save the movies and their corresponding ratings to a file named ***movies.txt***.