Hyperiondev

TASK

Capstone Project - Tracker App

Visit our website

Introduction

WELCOME TO THE CAPSTONE PROJECT - TRACKER APP TASK!

This Capstone is an important milestone in your learning. In this task, you will be consolidating the programmatic knowledge that you have gained up to this point and applying it to a real-world program! This Capstone project will give you a great opportunity to demonstrate your abilities using Python and SQL.

Don't be too worried about 'getting it right'. This is your chance to be creative and apply what you've learnt. You'll be tasked with a set of criteria to meet, but the rest is up to you. It is worth spending time and effort to make this a project that you can be proud of.

This project is a way for you to test your programming skills while creating an application that you can add to your portfolio. This project also allows you to highlight your development skills to a prospective employer!

In this project, you will have the option to create one of two tracker apps.

Option 1:

Build an expense and budget tracking app that allows users to monitor their spending, create expense categories, view spending summaries by category, and track their budget by calculating remaining funds based on their income and expenses.

Option 2:

Develop a fitness tracking app that enables users to create custom workout routines, save different types of exercises, and track progress towards their fitness goals by recording the number of sets/reps completed for each exercise.

Instructions

You may choose **either** Compulsory Task (Option 1) or Compulsory Task (Option 2).

You **do not** need to complete both options.

Compulsory Task (Option 1)

Follow these steps to build the expense and budget tracker app:

- Create a program that allows the user to:
    - add new expense categories to the database
    - update an expense amount
    - delete an expense category from the database
    - track their spending
    - add income
    - add income categories
    - delete an income category from the database
    - track their income
    - View expense or income categories
    - The program should be able to calculate the user's budget based on the income and expenses that they have provided

- Install the SQLite library. This will allow your app to communicate with the SQLite database.
- Connect to the SQLite database. You can do this by using the "connect" function from the sqlite3 library.
- Next, you will need to create your database tables to store your data. You can use the "execute" function to execute SQL commands to create tables.
- Insert data: After creating tables, ensure that users are able to insert data into them. You can use the "execute" function to execute SQL commands to insert data.
- Ensure that users can retrieve data from the database using SQL queries.
- Close the connection to the database using the "close" function.

- The program should present the user with the following menu:

```
1. Add expense
2. View expenses
3. View expenses by category
```

```
4. Add income
5. View income
6. View income by category
7. Set budget for a category
8. View budget for a category
9. Set financial goals
10.   View progress towards financial goals
11.   Quit
```

The program should perform the function that the user selects. The implementation of these functions is left up to you.

Compulsory Task (Option 2)

Follow these steps to build the fitness tracker app:

- Create a program that allows the user to:
    - add new workout categories to the database
    - update a workout category
    - delete a workout category from the database
    - add a workout goal
    - add goal categories
    - delete a goal category from the database
    - The program should be able to calculate the user's fitness goal progress based on the workouts and goals that they have provided
- The program could prompt users to input details about each exercise, such as the exercise name, muscle group, or number of reps/sets.
- Users could then create custom workout routines by selecting from their list of exercises.

- Install the SQLite library. This will allow your app to communicate with the SQLite database.

- Connect to the SQLite database. You can do this by using the "connect" function from the sqlite3 library.
- Next, you will need to create your database tables to store your data. You can use the "execute" function to execute SQL commands to create tables.
- Insert data: After creating tables, ensure that users are able to insert data into them. You can use the "execute" function to execute SQL commands to insert data.
- Ensure that users can retrieve data from the database using SQL queries.
- Close the connection to the database using the "close" function.

- The program should present the user with the following menu:

```
1. Add exercise category
2. View exercise by category
3. Delete exercise by category
4. Create Workout Routine
5. View Workout Routine
6. View Exercise Progress
7. Set Fitness Goals
8. View Progress towards Fitness Goals
9. Quit
```

The program should perform the function that the user selects. The implementation of these functions is left up to you.

Completed the task(s)?

Ask an expert to review your work!

Review work

Rate us

Share your thoughts

HyperionDev strives to provide internationally-excellent course content that helps you achieve your learning outcomes.

Think that the content of this task, or this course as a whole, can be improved? Do you think we've done a good job?

Click here to share your thoughts anonymously.