Habit Tracker Application

8.1.2024

Habit Tracker

A habit tracker is a tool that helps users develop and maintain habits by allowing them to create habits with specific frequencies (daily or weekly) and track their completion.
Users can mark habits as completed at any time. The app tracks the dates, noting streaks of consecutive completions.
In addition, the tracker provides analytics on habits and their performance about the longest streak.

Overview

Frameworks and Tools

- Python
- JSON

Implementing Components

- Habit Class
- Console Interface

Habit Creation, Management, and Analysis

- Implementation of functionalities to add and manage user-defined habits
- Features for tracking habit completion

Frameworks and Tools

Python Programming Language

- Easy readability, simplicity,powerful standard libraries

JSON Data Storage Format

- Seamless human-readable format

Datetime Module for Date and Time Management

- Tracking habit completion dates and times
- Calculation of streaks and analysis of habit patterns over time

```
from datetime import datetime, timedelta #This module is used for handling dates and times
import json #The json module is needed for reading and writing data to and from a JSON file
```

Implementing Components – Habit Class

```python
class Habit:

    # The __init__ method is the constructor for the Habit class. It initializes each new instance of the class.
    def __init__(self, name, periodicity):
        self.name = name # stores the name of the habit
        self.periodicity = periodicity #stores the frequency of the habit (1 for daily and 7 for weekly)
        self.completed_habit = [] #list that tracks when the habit was completed

    # The 'checkoff_habit' method is used to record when a habit is completed
    def checkoff_habit(self):
        current_time = datetime.now()
        self.completed_habit.append(current_time)
        print("{} checked off at {}.".format(self.name, current_time))
```

Data Storage and Management

Habit Data in JSON Format

- Each habit is represented as a JSON object with key-value pairs
- The completed_habit list is stored as an array of date-time strings

Interface (Examples)

Habit Menu

```
Menu:
1. View All Habits
2. Check off Task for a Habit
3. Create a New Habit
4. Exit
5. Analytics
Enter your choice (choose a number from 1 to 5):
```

```
Enter your choice (choose a number from 1 to 5): 1
1. Exercise
2. Learning
3. Reading
4. Yoga
5. Run
6. Tea
7. Coffee
```

Analytics Menu

```
Analytics
1. List all habits
2. List habits by periodicity
3. Display habit information
4. Longest streak for a specific habit
5. Return to main menu
Enter your choice (choose a number from 1 to 5):
```

```
Analytics
1. List all habits
2. List habits by periodicity
3. Display habit information
4. Longest streak for a specific habit
5. Return to main menu
Enter your choice (choose a number from 1 to 5): 3
1. Exercise
2. Learning
3. Reading
4. Yoga
5. Run
6. Tea
7. Coffee
Enter the habit number (choose a number from 1 to 7): 7

Information for Coffee:
Periodicity: 1
Completed:
 - 2024-01-06 16:37:41
```

Conclusion and Future Enhancements

Key Functionalities

- Create and customize habits
- Track habit completion
- Analyze performance through streaks

Future Enhancements and Features

- Web Application Development
  - Extending the tracker to a mobile app.
- Advanced Analytics:
  - Providing more detailed reports and insights, such as habit trends over time and personalized suggestions for habit improvement