

Index

- How to run it?
- Features

How to run it?

- Clone the repository from GitHub: git clone https://github.com/gianniprocida/habit_tracking_app.git

- Navigate to the habit-tracker directory: cd habit-tracking_app

- Install any dependencies required by the project:
pandas,matplotlib

- Run the project by executing the menu Python file: python menu.py

This should start the habit tracking program and allow you to interact with it via the user interface provided by the project.

Features

Create new habits and retrieve habit by id or by name

To start tracking habits, you need to create a HabitTracker instance by providing the name of the user. Then we pass in the name of the habit, "Study SQL", the start date of the habit period, "2023-04-01", the end date of the habit period, "2023-04-22", and the frequency, "W". Then we can call the `get_habit_by_id` method on the HabitTracker class and pass in the id of the habit we want to retrieve. This will return a 'Habit' instance with the specified id.

```
tracker = HabitTracker("John")
tracker.add_habit("Study SQL", "2023-04-01", "2023-04-22", "W")
tracker.add_habit("Study Python", "2023-04-01", "2023-04-22", "W")
tracker.add_habit("Study OOP", "2023-05-01", "2023-05-22", "W")

print(" ")
myhabit = tracker.get_habit_by_id(1)
print("The habit with id = 1 is :", myhabit.name)

print(" ")
myhabit = tracker.get_habit_by_id(2)
print("The habit with id = 2 is :", myhabit.name)

print(" ")
myhabit = tracker.get_habit_by_id(3)
print("The habit with id = 3 is :", myhabit.name)
```

```
✓ import pandas as pd ...
```

```
... Habit not found
Adding Study SQL...
Habit not found
Adding Study Python...
Habit not found
Adding Study OOP...

The habit with id = 1 is : Study SQL
The habit with id = 2 is : Study Python
The habit with id = 3 is : Study OOP
```

Retrieve habit by id or by name

Then we can call the `get_habit_by_name` method on the `HabitTracker` class and pass in the name of the habit we want to retrieve. This will return a `Habit` instance with the specified name.

```
print(" ")
myhabit = tracker.get_habit_by_id(1)
print("The habit with id = 1 is :",myhabit.name)

print(" ")
myhabit = tracker.get_habit_by_id(2)
print("The habit with id = 2 is :",myhabit.name)

print(" ")
myhabit = tracker.get_habit_by_id(3)
print("The habit with id = 3 is :",myhabit.name)

print(" ")
myhabit = tracker.get_habit_by_name("Study SQL")
print(myhabit.name)

print(" ")
myhabit = tracker.get_habit_by_name("Study Python")
print(myhabit.name)

print(" ")
myhabit = tracker.get_habit_by_name("Study OOP")
print(myhabit.name)
```

```
✓ import pandas as pd ...
```

```
... Habit not found
Adding Study SQL...
Habit not found
Adding Study Python...
Habit not found
Adding Study OOP...

The habit with id = 1 is : Study SQL

The habit with id = 2 is : Study Python

The habit with id = 3 is : Study OOP

Study SQL

Study Python

Study OOP
```

Retrieve dates by name

Then we can call the `get_dates_by_name` method by providing the name of the habit as input on the HabitTracker to retrieve the dates on which a habit needs to be completed. This will print a dataframe that contains the dates relative to the habit.

```
360 |         print(f"Checking off {myhabit.name}")
361 |         myhabit = tracker.get_habit_by_name("Study SQL")
362 |         print(myhabit.name)
363 |
364 |         print(" ")
365 |         tracker.get_dates_by_name("Study SQL")
366 |
367 |         print(" ")
368 |         print("Checking_off Study SQL")
369 |
370 |         tracker.checkoff_by_name("Study SQL", "y")
371 |         tracker.checkoff_by_name("Study SQL", "n")
```

PROBLEMS 145 OUTPUT DEBUG CONSOLE TERMINAL

The habit with id = 3 is : Study OOP

Study SQL

	Date
0	2023-04-01
1	2023-04-08
2	2023-04-15
3	2023-04-22

Checkoff “Study SQL” habit

We call the `checkoff_by_name` method on the `HabitTracker` class and pass in the name of the habit we want to check off. This will add the string "y" to the `checkoffList` of the `Habit` instance with the specified name, which indicates that the habit was completed on the first day of the listed dates. We use the checking off method for the same number of times as the number of dates on which the habit needs to be completed.

```
352 | tracker.get_dates_left_by_name("Study SQL")
353 |
354 | print(" ")
355 | print("Checking_off Study SQL")
356 |
357 | tracker.checkoff_by_name("Study SQL","y")
358 | tracker.checkoff_by_name("Study SQL","n")
359 | tracker.checkoff_by_name("Study SQL","y")
360 | tracker.checkoff_by_name("Study SQL","n")
361 | print(" ")
362 | print("CheckoffList",tracker.get_habit_by_name("Study SQL").checkoffList)
363 |
364 |
365 | print(" ")
366 | print(" ")
367 |
368 | print("Checking_off Study Python")
```

PROBLEMS 65 OUTPUT DEBUG CONSOLE TERMINAL

```
Date
0 2023-04-01
1 2023-04-08
2 2023-04-15
3 2023-04-22

Checking_off Study SQL
Other 3 check marks left
Other 2 check marks left
Other 1 check marks left
Other 0 check marks left
The daily habit of Study SQL was completed within the period of 2023-04-01 to 2023-04-22

CheckoffList ['y', 'n', 'y', 'n']
```


Checkoff “Study Python” habit

```
367
368     print("Checking_off Study Python")
369
370     tracker.checkoff_by_name("Study Python","y")
371     tracker.checkoff_by_name("Study Python","y")
372     tracker.checkoff_by_name("Study Python","y")
373     tracker.checkoff_by_name("Study Python","n")
374     print(" ")
375     print("CheckoffList",tracker.get_habit_by_name("Study Python").checkoffList)
376
377     print(" ")
378     print(" ")
379
380     print("Checking_off Study OOP")
381
382     tracker.checkoff_by_name("Study OOP","y")
383     tracker.checkoff_by_name("Study OOP","y")
384     tracker.checkoff_by_name("Study OOP","n")
385     tracker.checkoff_by_name("Study OOP","n")
```

PROBLEMS 65 OUTPUT DEBUG CONSOLE TERMINAL

```
Checking_off Study Python
Other 3 check marks left
Other 2 check marks left
Other 1 check marks left
Other 0 check marks left
The daily habit of Study Python was completed within the period of 2023-04-01 to 2023-04-22

CheckoffList ['y', 'y', 'y', 'n']
```

Checkoff “Study Python” habit

We have completed the task of checking off three habits.

```
379
380     print("Checking_off Study 00P")
381
382     tracker.checkoff_by_name("Study 00P","y")
383     tracker.checkoff_by_name("Study 00P","y")
384     tracker.checkoff_by_name("Study 00P","n")
385     tracker.checkoff_by_name("Study 00P","n")
386     print(" ")
387     print("CheckoffList for 'Study 00P'",tracker.get_habit_by_name("Study 00P").checkoffList)
388
389     # Add another habit
390     tracker.add_habit("Study ML","2023-05-01","2023-05-07","D")
391
392     print(" ")
393     print("Checking_off Study ML")
394     tracker.checkoff_by_name("Study ML","y")
395     tracker.checkoff_by_name("Study ML","y")
```

PROBLEMS 65 OUTPUT DEBUG CONSOLE TERMINAL

```
CheckoffList ['y', 'y', 'y', 'n']
```

```
Checking_off Study 00P
```

```
Other 3 check marks left
```

```
Other 2 check marks left
```

```
Other 1 check marks left
```

```
Other 0 check marks left
```

```
The daily habit of Study 00P was completed within the period of 2023-05-01 to 2023-05-22
```

```
CheckoffList for 'Study 00P' ['y', 'y', 'n', 'n']
```

Results

We have added a new habit titled "Study ML" and have successfully completed it.

```
388
389 # Add another habit
390 tracker.add_habit("Study ML", "2023-05-01", "2023-05-07", "D")
391
392 print(" ")
393 print("Checking_off Study ML")
394 tracker.checkoff_by_name("Study ML", "y")
395 tracker.checkoff_by_name("Study ML", "y")
396 tracker.checkoff_by_name("Study ML", "n")
397 tracker.checkoff_by_name("Study ML", "y")
398 tracker.checkoff_by_name("Study ML", "y")
399 tracker.checkoff_by_name("Study ML", "y")
400 tracker.checkoff_by_name("Study ML", "y")
401
402 print("CheckoffList for 'Study ML'", tracker.get_habit_by_name("Study ML").checkoffList)
403
```

PROBLEMS 65 OUTPUT DEBUG CONSOLE TERMINAL

```
CheckoffList for 'Study OOP' ['y', 'y', 'n', 'n']
Habit not found
Adding Study ML...

Checking_off Study ML
Other 6 check marks left
Other 5 check marks left
Other 4 check marks left
Other 3 check marks left
Other 2 check marks left
Other 1 check marks left
Other 0 check marks left
The daily habit of Study ML was completed within the period of 2023-05-01 to 2023-05-07
CheckoffList for 'Study ML' ['y', 'y', 'n', 'y', 'y', 'y', 'y']
```

Results

For each habit in our tracker, we retrieve the habit with the specified name and check its `longest_habit_streak` attribute, which is in agreement with our expectations.

```
407 res = myhabit.longest_habit_streak
408 print("Longest run streak for Study SQL:",res)
409 print(" ")
410 myhabit = tracker.get_habit_by_name("Study Python")
411 res = myhabit.longest_habit_streak
412 print("Longest run streak for Study Python:",res)
413 print(" ")
414 res = tracker.get_habit_by_name("Study OOP").longest_habit_streak
415 print("Longest run streak for OOP:",res)
416 print(" ")
417 res = tracker.get_habit_by_name("Study ML").longest_habit_streak
418 print("Longest run streak for ML:",res)
419 print(" ")
420 res = tracker.get_habit_with_longest_run_streak_of_all()
421 print("Longest run streak of all defined habits:",res)
422 print("")
423 myhabits = tracker.get_habits_with_same_property("freq")
424 print("Habit with the same freq:",myhabits)
```

PROBLEMS 65 OUTPUT DEBUG CONSOLE TERMINAL

CheckoffList for 'Study ML' ['y', 'y', 'n', 'y', 'y', 'y', 'y']

Longest run streak for Study SQL: 1

Longest run streak for Study Python: 3

Longest run streak for OOP: 2

Longest run streak for ML: 4

Longest run streak of all defined habits: {'Study ML': 4}

Habit with the same freq: {'W': ['Study SQL', 'Study Python', 'Study OOP'], 'D': ['Study ML']}

-The user established a 1-day streak of studying sql

-The user established a 3-days streak of studying Python

-The user established a 2-day streak of studying OOP

-The user established a 4-day streak of studying ML

Results

By calling the `get_habit_with_longest_run_streak_of_all` method, we discover that the habit with the longest streak among all defined habits in our tracker is "Study ML", which has a run streak value of 4.

```
407 res = myhabit.longest_habit_streak
408 print("Longest run streak for Study SQL:",res)
409 print(" ")
410 myhabit = tracker.get_habit_by_name("Study Python")
411 res = myhabit.longest_habit_streak
412 print("Longest run streak for Study Python:",res)
413 print(" ")
414 res = tracker.get_habit_by_name("Study OOP").longest_habit_streak
415 print("Longest run streak for OOP:",res)
416 print(" ")
417 res = tracker.get_habit_by_name("Study ML").longest_habit_streak
418 print("Longest run streak for ML:",res)
419 print(" ")
420 res = tracker.get_habit_with_longest_run_streak_of_all()
421 print("Longest run streak of all defined habits:",res)
422 print("")
423 myhabits = tracker.get_habits_with_same_property("freq")
424 print("Habit with the same freq:",myhabits)
```

PROBLEMS 65 OUTPUT DEBUG CONSOLE TERMINAL

CheckoffList for 'Study ML' ['y', 'y', 'n', 'y', 'y', 'y', 'y']

Longest run streak for Study SQL: 1

Longest run streak for Study Python: 3

Longest run streak for OOP: 2

Longest run streak for ML: 4

Longest run streak of all defined habits: {'Study ML': 4}

Habit with the same freq: {'W': ['Study SQL', 'Study Python', 'Study OOP'], 'D': ['Study ML']}

Results

We invoke the `get_habit_with_same_property` by providing the string ‘freq’ in one case and ‘time_period_string’ in the other case to retrieve habits with same frequency and same time period, respectively.

```
431 res = tracker.get_habit_by_name('Study ML', longest_run_streak)
432 print(" ")
433 res = tracker.get_habit_with_longest_run_streak_of_all()
434 print("Longest run streak of all defined habits:",res)
435 print(" ")
436 myhabits = tracker.get_habits_with_same_property("freq")
437 print("Habits with the same freq:",myhabits)
438 print(" ")
439 myhabits = tracker.get_habits_with_same_property("time_period_string")
440 print("Habits with the same time period:",myhabits)
441
442
443
444 res = tracker.data_visualization()
445
446 print("Number of times a user has completed a habit over a certain period of time",res)
447
448
```

PROBLEMS 145 OUTPUT DEBUG CONSOLE TERMINAL

bash + ▾ □ 🗑️ ⋮ ^ ×

Longest run streak for OOP: 2

Longest run streak for ML: 4

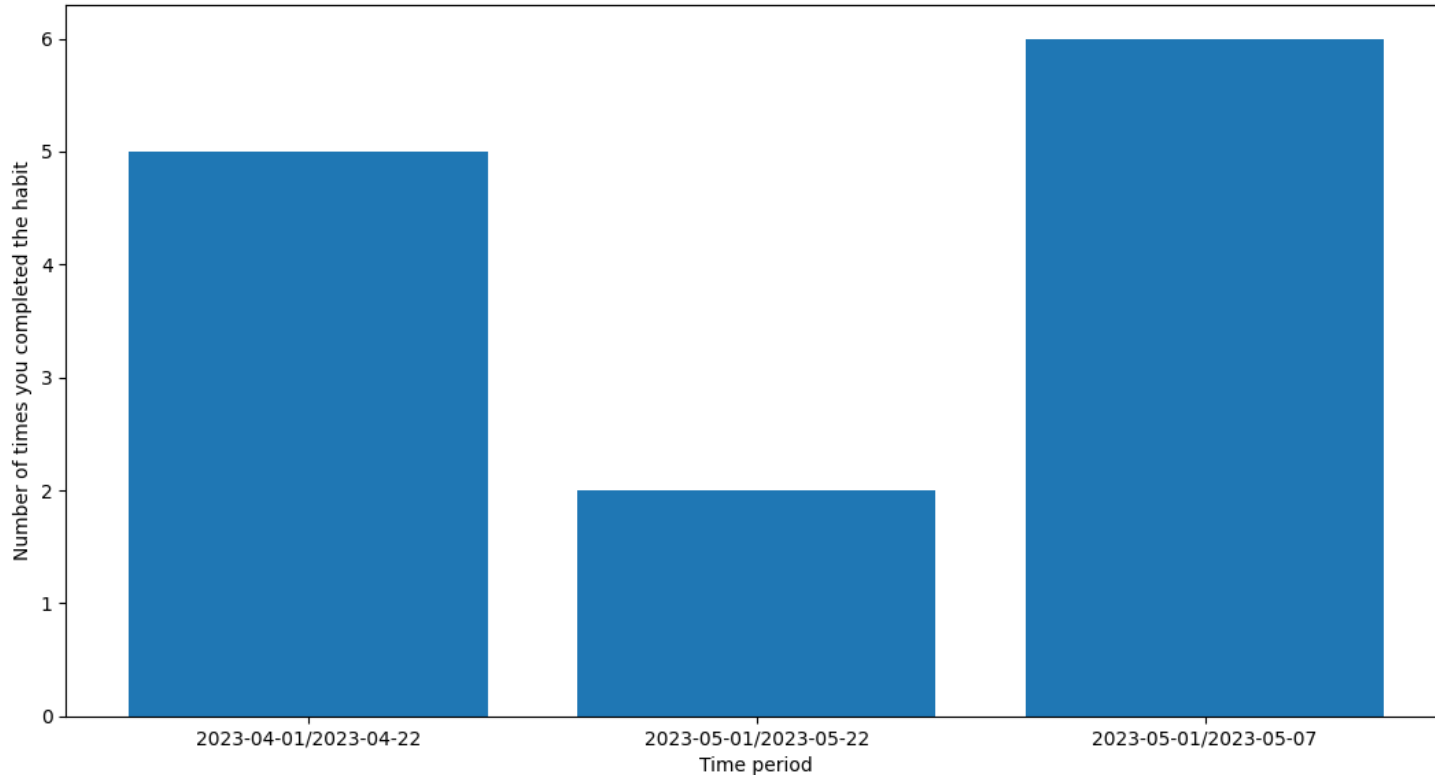
Longest run streak of all defined habits: {'Study ML': 4}

Habits with the same freq: {'W': ['Study SQL', 'Study Python', 'Study OOP'], 'D': ['Study ML']}

Habits with the same time period: {'2023-04-01/2023-04-22': ['Study SQL', 'Study Python'], '2023-05-01/2023-05-22': ['Study OOP'], '2023-05-01/2023-05-07': ['Study ML']}

Results

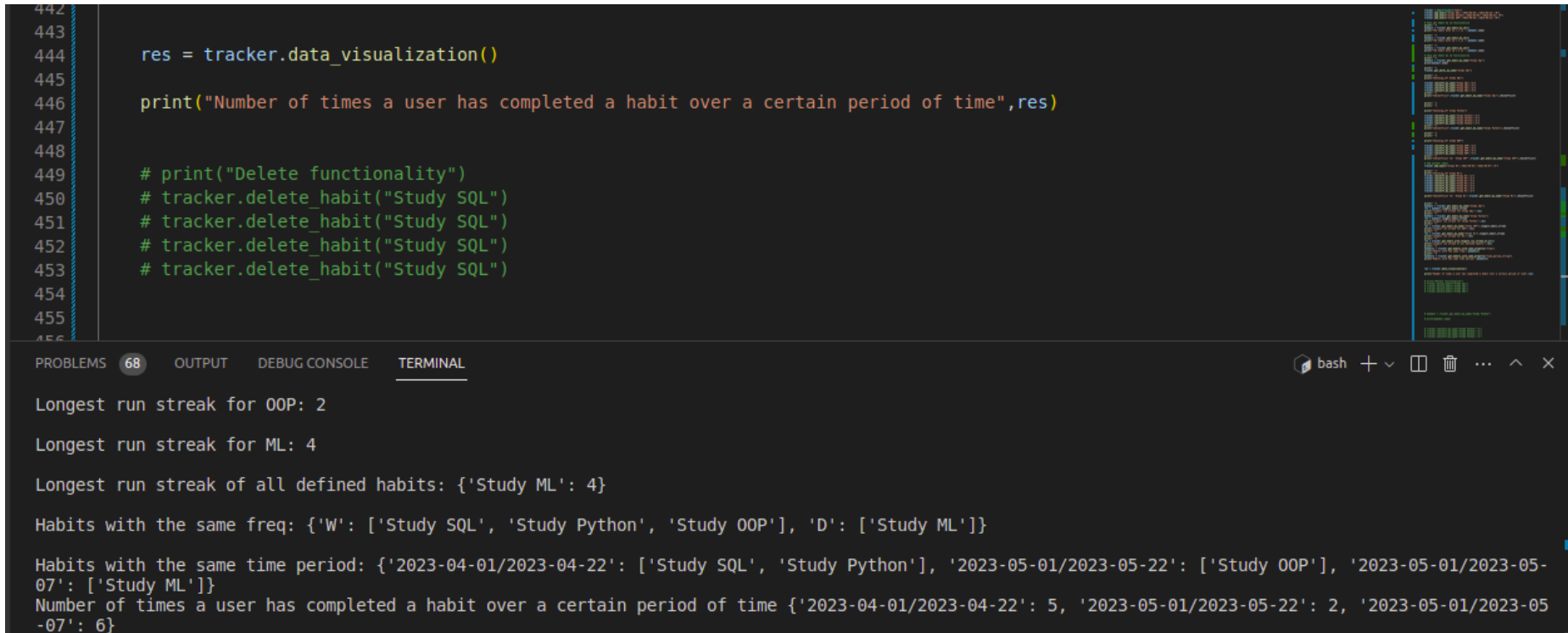
We use the `data_visualization` method of the `HabitTracking` class to create a bar chart that shows the total number of times all tracked habits were completed over a certain period of time. Each bar represents a different time period, and its height corresponds to the total number of habit completions during that period, regardless of which specific habit was completed



Results

During the time period of April 1, 2023 to April 22, 2023, we completed the habit of studying SQL twice and the habit of studying Python three times so the total sum is five.

During the time period of May 1, 2023 to May 22, 2023, we completed the habit of studying OOP twice, etc.



```
442
443
444     res = tracker.data_visualization()
445
446     print("Number of times a user has completed a habit over a certain period of time",res)
447
448
449     # print("Delete functionality")
450     # tracker.delete_habit("Study SQL")
451     # tracker.delete_habit("Study SQL")
452     # tracker.delete_habit("Study SQL")
453     # tracker.delete_habit("Study SQL")
454
455
456
```

PROBLEMS 68 OUTPUT DEBUG CONSOLE TERMINAL

bash + - [] [] ... ^ x

Longest run streak for OOP: 2

Longest run streak for ML: 4

Longest run streak of all defined habits: {'Study ML': 4}

Habits with the same freq: {'W': ['Study SQL', 'Study Python', 'Study OOP'], 'D': ['Study ML']}

Habits with the same time period: {'2023-04-01/2023-04-22': ['Study SQL', 'Study Python'], '2023-05-01/2023-05-22': ['Study OOP'], '2023-05-01/2023-05-07': ['Study ML']}

Number of times a user has completed a habit over a certain period of time {'2023-04-01/2023-04-22': 5, '2023-05-01/2023-05-22': 2, '2023-05-01/2023-05-07': 6}

Delete habits

We call the `delete_habit` method on the HabitTracking class and pass in the name of the habit we want to delete. This will remove the habit from the tracker object.

```
447
448
449     print("Delete functionality")
450     tracker.delete_habit("Study SQL")
451     tracker.delete_habit("Study OOP")
452     tracker.delete_habit("Study Python")
453     tracker.delete_habit("Study ML")
454
455     assert tracker.habits == []
456
457
458
```

PROBLEMS 68 OUTPUT DEBUG CONSOLE TERMINAL bash + ~

```
Habits with the same freq: {'W': ['Study SQL', 'Study Python', 'Study OOP'], 'D': ['Study ML']}
Habits with the same time period: {'2023-04-01/2023-04-22': ['Study SQL', 'Study Python'], '2023-05-01/2023-05-22': ['Study OOP'], '2023-05-07': ['Study ML']}
Number of times a user has completed a habit over a certain period of time {'2023-04-01/2023-04-22': 5, '2023-05-01/2023-05-22': 2, '2023-05-07': 6}
Delete functionality
Study SQL deleted
Study OOP deleted
Study Python deleted
Study ML deleted
```

This will remove all the habits from the tracker object.