

Description

Whether training for a race or just improving performance, a running application (app) will be created to provide runners with a simple and easy solution to track their milage. The app makes it easier for runners to input their data to conduct several calculations to evaluate and understand their performance.

The user would use a desktop Graphical User Interface (GUI) to interact and input running data in the app. The user would input the date (as month, day, and year), select road or trail running, the number of miles they ran, and the amount of time (in hours, minutes, and seconds) it took them to complete the run. The app would store data in an Excel spreadsheet. The user can also edit and delete their data from the GUI.

Based on the data the user inputs, the user will be able to view their data in the app. Also, the app will calculate the average pace or rate for each run, the weekly total average, weekly average for trail running, and weekly average for road running for a 4-week period.

Task Vignettes

There are several tasks that are completed by the user. They would need to open and run the GUI. Next, they would input their date, select road or trail running, input the number of miles they ran, and input the amount of time it took them to complete the run.

- App assigns user's running values to corresponding variables.
- App sends information and stores data in Excel.

The user can view their running data.

- App displays the row(s) of running data.
- App calculates and displays average pace for each run, weekly total average, weekly average for trail running, and weekly average for road running.

The user can edit or delete a row of data within the GUI.

- Based on what the user does, data in the Excel spreadsheet would reflect the changes.

Running App

Month:

Day:

Year:

Run Type: ☐ Road run ☐ Trail Run

Hours:

Minutes:

Seconds:

Miles:

Weekly Total Average: ##.##

Weekly Road Run Average: ##.##

Weekly Trail Run Average: ##.##

Excel Spreadsheet

Technical Flow

The app will utilize TkInter to create the GUI and Pandas to communicate with Excel. Each row in the table will have the date (as month, day, and year), user selected road or trail running, the number of miles they ran, and the amount of time (in hours, minutes, and seconds), and pace (calculated by the app). Month, day, and year will be an integer. The run type will be a radio button where the user selects road or trail run. Miles will be a float number. Hours, minutes, and seconds will be integers. The user will have the ability to edit or delete a row of data in the GUI.

Core parts:

- Class GUI importing TkInter
 - Layout of the GUI will be handled here
 - Assign buttons
 - Display Excel spreadsheet in GUI
- Error handling
 - Makes sure the user is inputting proper value types.
- User inputs
 - Assign variables as user inputs.
- Communicating with Excel spreadsheet via Pandas
 - Display data
 - Insert new rows of data
 - Be able to delete and edit data
- Calculations

- Pace for each run
- Calculate weekly averages for a 4-week period
 - weekly total average
 - trail running
 - road running

Self-Assessment

An unexpected change I had to make was including the ability to edit and delete data from the GUI. There is always a possibility that the user will make a mistake when inputting their running data. I feel somewhat confident that I can implement this spec. I believe I will most likely run into problems with having the app communicate and manipulate data in Excel. I believe I will somewhat run into problems with pulling the correct data to complete the calculations and doing the GUI's layout. I am least familiar with using Pandas to communicate and manipulate data in Excel.