Main Project Outline

Scenario

A police department is considering using wearable fitness trackers (like Fitbits) to monitor the physical fitness of its officers. Instead of traditional fitness tests, they want to use continuous data from the trackers to assess overall health and fitness levels, discouraging the practice of only exercising in the month leading up to a fitness test. Each officer is given a fitness score which can change month to month, as determined by a combination of several measurements - including:

- Heart rate variability (HRV) - typical measure of cardiovascular health, measured in milliseconds. Decreases with age, so it would be wise to also measure age of officers.

- Heart rate - can be measured alongside HRV in order to determine high-stress periods of time where it may be advantageous to exclude these time periods from the HRV calculation, as it wouldn't be representative of the individual's direct health.

- Steps taken per day - measure of daily activity to deter from sedentary lifestyle

- Age of officers - to age-correct HRV because a healthy HRV changes over the age of an individual.

Config.py

- Constants
  - HRV_WEIGHT (int)
  - STEPS_WEIGHT (int)
  - MAX_DEPT_SIZE (int)
  - MAX_HEART_RATE (int)
  - FILENAME_STEPS (string)
  - FILENAME_AGE (string)
  - OUTPUT_FOLDER (string)

Module 1: IndividualData

- Parent class: FitnessData
  - Attributes:
    - person name
    - person age
    - dataframe steps
    - dataframe HRV
    - CONSTANTS (dictionary)
      - HRV_WEIGHT
      - STEPS_WEIGHT
      - MAX_HEART_RATE
      - FILENAME_STEPS
      - FILENAME_AGE
  - Methods
    - view steps table
    - view HRV table
    - view steps line graph
    - view HRV line graph
- Child class: FitnessDataProcessing
  - Attributes:
    - fitness score
    - avg steps per day
    - avg HRV
    - folderName
    - dataframe averageHRV (each day's average * age correction)
  - Methods

- import_csv

- clean data (remove bad HRV data when heart rate > max)

- calcHRV – create dataframe for average HRV per day, age corrected

- calcStepScore

- calcHRVScore

- calcFitnessScore

- graph HRV per day (whisker plot)

- graph steps per day (whisker plot)

- Show stats for the month:

  - Average steps per day

  - Average HRV per day

  - Fitness score

Module 2: DepartmentData

- Parent class: DepartmentData
  - Attributes:
    - departmentName
    - set of IndividualDataProcessing (data for individuals in department)
      - Not to exceed MAX_DEPT_SIZE
    - days in given month
    - dataframe: department steps per day
    - dataframe: department HRV average per day
    - dataframe: fitness score per individual
  - Methods:
    - Add individual to set
    - Remove individual from set
    - Generate dataframes

- department steps per day

- department HRV average per day

- fitness score per individual

- age per fitness score

- Child class: DepartmentDataProcessing

  - Attributes:

    - CONSTANT: string outputFolder

    - dataFile (string)

    - dataframe: steps stats

    - dataframe: HRV stats

    - dataframe: fitness score stats

    - dataframe: age stats

    - plots:

      - steps per individual

      - HRV per individual

      - fitness score per individual

      - age per fitness score

  - Methods:

    - loadData – load from pickle file

    - saveData – dump to pickle file

    - calcStats – calculate statistical values such as mean, min, max

      - steps, HRV, fitness score, age

    - graphing

      - average steps per individual (histogram)

      - HRV averages per individual (histogram)

      - fitness score per individual (histogram)

      - age per fitness score (scatter plot)

main.py

- Method: user interface
  - Options:
    - Process data for individual
      - Provide name of individual
    - Create department
    - Add individuals to department
    - Process department data
    - View graphs
      - Individuals
      - Department