Python Athletics (py_athletics)

A tool for gathering, organizing and analyzing fitness activities

Richard Robbins

Usage

`python py_athletics.py`

`py_athletics.py` is contained in the `py_athletics/py_athletics/src/` directory of the repository. All examples in this document assume that the user's current directory is that directory. If the user is in another directory, examples with filename references included in this document or in the system's help system will not work without making changes to relative pathname prefixes. The examples are illustrative only, there are no special limits on imposed by **py_athletics** on the pathname component of filenames supplied as optional arguments to its commands.

Quick Start

The information about **py_athletics** included below is also available as part of the program's help system, accessed with the `help` command.

The `py_athletics/py_athletics/test` directory contains useful sample scripts and data sets that users can experiment with. In particular, `test/garmin_data` includes several smaller exercise data sets.

To read a full set of data, use `read ../test/Activities.csv` which provides nearly ten full months of exercise data.

To load a representative set of goals instead of crafting your own, use `run_script ../test/goals.cmd`

To restore a representative full session, use `load ../test/py_athletics.pickle`

As noted below, detailed documentation derived from the source code by `pdoc3` can be found here. That same collection is also included as part of the **py_athletics** repository in both `html` and `md` format. See the `py_athletics/py_athletics/documents/modules/` directory. Information about `pdoc3` can be found here.

Condensed Project Directory Tree

```
py_athletics
├── README.md
├── README.pdf
├── Reflections.md
├── Reflections.pdf
└── py_athletics
    ├── documents
    ├── misc
    ├── src
    └── test
```

The User Interface

The system is invoked from a shell with the `python py_athletics.py` command. See the note above about the location of `py_athletics.py`.

The command line interpreter is an instance of the `PythonAthleticsShell` class which is implemented as a subclass of the Python `cmd` class.

Command	Description
load	Restore py_athletics session from a file.
read	Read a Garmin activity file and create Activity objects.
run_script	Run py_athletics commands from a script.
save	Save py_athletics session to a file.
add_goal	Add a goal.
delete_goal	Delete a goal.
show_activities	Display a list of activities.
show_goals	Display a list of goals.
summarize_activities	Display a summary of activities.
summarize_goals	Display a summary of goals.
help or ?	List available commands with "help" or detailed help with "help cmd".
shell or !	Run an OS shell command.
exit	Exit.

The **py_athletics** command line interpreter provides tab-key command line completion and bash-like history editing.

More detailed summaries of the various commands are included below. The same summaries are included in the program's help system.

Command Summary Descriptions with Examples

load

```
Restore py_athletics session from a file.

        The default filename is py_athletics.pickle, a different name can be
        specified with the filename argument.

        Optional Parameters
        -------------------
        filename: string

        Examples
        --------
        load
        load ..test/py_athletics_session.pickle
```

read

```
Read a Garmin activity file and create Activity objects.

        Garmin fitness data is stored at http://connect.garmin.com.
        Subscribers can download comprehensive activity data into a CSV file.
        This command reads a Garmin activity file, creates py_athletics
        Activity objects and adds them to the Athlete's activity collection.

        The  default filename is Activities.csv, a different name can be
```

```
        specified with the filename argument.

        Optional Parameters
        -------------------
        filename: string

        Examples
        --------
        read
        read ..test/garmin_data/2021-01.csv
```

run_script

```
Run py_athletics commands from a script.

        The default filename is py_athletics.cmd, a different name can be
        specified with the filename argument.

        Optional Parameters
        -------------------
        filename: string

        Examples
        --------
        run_script
        run_script ..test/goals.cmd
```

Here is the body of a command file that an athlete could use to establish a set of `Goals` that can be executed with the `run_script` command.

```
add_goal exercise=Cycle metric=distance timeframe=year target=1500
add_goal exercise=Cycle metric=distance timeframe=month target=125
add_goal exercise=Run metric=distance timeframe=year target=150
add_goal exercise=Run metric=distance timeframe=month target=13
add_goal exercise=Tennis metric=count timeframe=month target=8
add_goal exercise=Workout metric=count timeframe=month target=8
```

save

```
Save py_athletics session to a file.

        The default filename is py_athletics.pickle, a different name can be
        specified with the filename argument.

        Optional Parameters
        -------------------
        filename: string

        Examples
        --------
        save
        save ..test/py_athletics_session.pickle
```

add_goal

```
Add a Goal.

        Adds a goal and replaces existing Goals for the same exercise,
        timeframe and metric.  The distance metric is only valid for Cycle,
        Run and Walk activities.

        Keyword Parameters
        ------------------
        exercise: string = {Cycle|Run|Tennis|Walk|Workout}
        metric: string = {count|distance|duration}
        timeframe: string = {month|year|cumulative}
        target: a positive integer

        Examples
        --------
        add_goal exercise=Cycle metric=distance timeframe=year target=1500
        add_goal exercise=Tennis metric=count timeframe=month target=8
```

delete_goal

```
Delete a Goal.

        Keyword Parameters
        ------------------
        exercise: string = {Cycle|Run|Tennis|Walk|Workout}
        metric: string = {count|distance|duration}
        timeframe: string = {month|year|cumulative}

        Examples
        --------
        delete_goal exercise=Cycle metric=distance timeframe=year
        delete_goal exercise=Tennis metric=count timeframe=month
```

show_activities

```
Display a list of Activities.

        If exercise is specified the listing is limited to that exercise.
        A timeframe for the listing can be established with one or both of the
        start and end keywords.

        Optional Parameters
        -------------------
        exercise: string = {Cycle|Run|Tennis|Walk|Workout}
        start: string in the form YYYY-MM-DD
        end: string in the form YYYY-MM-DD

        Examples
        --------
        show_activities
        show_activities exercise=Tennis
        show_activities exercise=Tennis start=2021-05-01 end=2021-06-30
```

For example, here is an activity list for March 2021.

```
py_athletics: show_activities start=2021-03-01 end=2021-03-31
```

[Cycle (Longboat Key Cycling) on 2021-03-25 at 10:16 for 1:00:05 Calories: 410 Max HR: 144 Avg HR: 130 Distance (miles): 16.56 Max Speed (mph): 19.2 Avg Speed (mph): 16.5 Normalized Power (watts): 109]
[Cycle (Chicago Road Cycling) on 2021-03-22 at 14:45 for 1:43:05 Calories: 638 Max HR: 141 Avg HR: 126 Distance (miles): 22.19 Max Speed (mph): 22.5 Avg Speed (mph): 12.9 Normalized Power (watts): 122]
[Cycle (Longboat Key Cycling) on 2021-03-18 at 07:43 for 1:31:15 Calories: 500 Max HR: 123 Avg HR: 115 Distance (miles): 23.11 Max Speed (mph): 18.5 Avg Speed (mph): 15.2 Normalized Power (watts): 85]
[Cycle (Longboat Key Virtual Cycling) on 2021-03-12 at 09:43 for 1:45:33 Calories: 775 Max HR: 145 Avg HR: 124 Distance (miles): 30.03 Max Speed (mph): 20.1 Avg Speed (mph): 17.0 Normalized Power (watts): 115]
[Cycle (Chicago Road Cycling) on 2021-03-09 at 06:57 for 1:02:16 Calories: 460 Max HR: 145 Avg HR: 133 Distance (miles): 14.01 Max Speed (mph): 21.2 Avg Speed (mph): 13.5 Normalized Power (watts): 144]
[Cycle (Longboat Key Cycling) on 2021-03-06 at 09:26 for 2:54:36 Calories: 958 Max HR: 136 Avg HR: 124 Distance (miles): 40.03 Max Speed (mph): 20.3 Avg Speed (mph): 13.8 Normalized Power (watts): 104]
[Run (Chicago Running) on 2021-03-23 at 07:38 for 0:42:52 Calories: 508 Max HR: 158 Avg HR: 148 Distance (miles): 4.00 Max Speed (minutes/mile): 08:06 Avg Speed (minutes/mile): 10:42]
[Run (Indoor Running) on 2021-03-15 at 08:28 for 0:20:14 Calories: 247 Max HR: 176 Avg HR: 147 Distance (miles): 2.01 Max Speed (minutes/mile): 08:54 Avg Speed (minutes/mile): 10:04]
[Run (Indoor Running) on 2021-03-13 at 08:19 for 0:29:44 Calories: 380 Max HR: 158 Avg HR: 148 Distance (miles): 3.03 Max Speed (minutes/mile): 08:33 Avg Speed (minutes/mile): 09:49]
[Run (Indoor Running) on 2021-03-10 at 08:12 for 0:21:18 Calories: 264 Max HR: 154 Avg HR: 145 Distance (miles): 2.00 Max Speed (minutes/mile): 09:13 Avg Speed (minutes/mile): 10:39]
[Run (Indoor Running) on 2021-03-08 at 08:29 for 0:04:17 Calories: 53 Max HR: 159 Avg HR: 146 Distance (miles): 0.51 Max Speed (minutes/mile): 07:05 Avg Speed (minutes/mile): 08:29]
[Run (Indoor Running) on 2021-03-07 at 09:25 for 0:19:46 Calories: 254 Max HR: 158 Avg HR: 150 Distance (miles): 2.00 Max Speed (minutes/mile): 08:17 Avg Speed (minutes/mile): 09:52]
[Run (Indoor Running) on 2021-03-02 at 07:28 for 0:43:06 Calories: 552 Max HR: 157 Avg HR: 147 Distance (miles): 4.00 Max Speed (minutes/mile): 09:04 Avg Speed (minutes/mile): 10:47]
[Tennis on 2021-03-30 at 09:04 for 0:55:54 Calories: 377 Max HR: 131 Avg HR: 112]
[Tennis on 2021-03-27 at 08:55 for 1:05:59 Calories: 667 Max HR: 163 Avg HR: 130]
[Tennis on 2021-03-21 at 07:59 for 1:03:21 Calories: 496 Max HR: 150 Avg HR: 115]
[Tennis on 2021-03-20 at 08:58 for 1:02:38 Calories: 574 Max HR: 152 Avg HR: 125]
[Tennis on 2021-03-16 at 07:00 for 0:56:53 Calories: 507 Max HR: 150 Avg HR: 122]
[Tennis on 2021-03-13 at 09:09 for 0:52:42 Calories: 600 Max HR: 157 Avg HR: 138]
[Tennis on 2021-03-10 at 07:00 for 0:58:56 Calories: 534 Max HR: 137 Avg HR: 123]
[Tennis on 2021-03-07 at 08:04 for 0:57:11 Calories: 560 Max HR: 150 Avg HR: 125]
[Tennis on 2021-03-04 at 07:05 for 0:52:57 Calories: 421 Max HR: 133 Avg HR: 114]
[Walk (Chicago Walking) on 2021-03-21 at 13:43 for 1:08:11 Calories: 336 Max HR: 131 Avg HR: 103 Distance (miles): 3.09 Max Speed (minutes/mile): 16:26 Avg Speed (minutes/mile): 22:05]
[Walk (Chicago Walking) on 2021-03-21 at 11:12 for 1:03:23 Calories: 334 Max HR: 128 Avg HR: 105 Distance (miles): 3.00 Max Speed (minutes/mile): 15:27 Avg Speed (minutes/mile): 21:07]
[Walk (Chicago Walking) on 2021-03-13 at 15:55 for 0:39:27 Calories: 189 Max HR: 122 Avg HR: 93 Distance (miles): 1.70 Max Speed (minutes/mile): 13:04 Avg Speed (minutes/mile): 23:11]
[Walk (Chicago Walking) on 2021-03-13 at 13:19 for 0:23:11 Calories: 111 Max HR: 102 Avg HR: 88 Distance (miles): 1.04 Max Speed (minutes/mile): 14:18 Avg Speed (minutes/mile): 22:23]
[Walk (Chicago Walking) on 2021-03-08 at 14:10 for 0:34:34 Calories: 176 Max HR: 103 Avg HR: 72 Distance (miles): 1.71 Max Speed (minutes/mile): 16:04 Avg Speed (minutes/mile): 20:14]
[Workout (Strength and Flexibility) on 2021-03-29 at 07:21 for 0:54:33 Calories: 321 Max HR: 125 Avg HR: 102]
[Workout (Strength and Flexibility) on 2021-03-26 at 07:47 for 1:09:23 Calories: 416 Max HR: 127 Avg HR: 106]
[Workout (Strength and Flexibility) on 2021-03-22 at 07:19 for 0:56:02 Calories: 332 Max HR: 129 Avg HR: 103]
[Workout (Strength and Flexibility) on 2021-03-17 at 07:01 for 1:07:13 Calories: 340 Max HR: 119 Avg HR: 99]
[Workout (Strength and Flexibility) on 2021-03-15 at 07:14 for 1:00:54 Calories: 299 Max HR: 124 Avg HR: 96]
[Workout (Strength and Flexibility) on 2021-03-11 at 07:29 for 1:22:22 Calories: 495 Max HR: 134 Avg HR: 105]
[Workout (Strength and Flexibility) on 2021-03-08 at 07:22 for 0:53:52 Calories: 247 Max HR: 118 Avg HR: 90]

```
[Workout (Strength and Flexibility) on 2021-03-05 at 07:32 for 1:34:02 Calories: 483 Max HR: 123
Avg HR: 95]
[Workout (Strength and Flexibility) on 2021-03-01 at 07:24 for 0:49:35 Calories: 270 Max HR: 120
Avg HR: 97]
py_athletics:
```

show_goals

```
Display a list of Goals.

        If exercise is specified, the listing is limited to that exercise.

        Optional Parameters
        -------------------
        exercise: string = {Cycle|Run|Tennis|Walk|Workout}

        Examples
        --------
        show_goals
        show_goals Cycle
```

Here is a representative goal listing.

```
py_athletics: show_goals
[Goal: Cycle metric: distance timeframe: year target: 1,500]
[Goal: Cycle metric: distance timeframe: month target: 125]
[Goal: Run metric: distance timeframe: year target: 150]
[Goal: Run metric: distance timeframe: month target: 13]
[Goal: Tennis metric: count timeframe: month target: 8]
[Goal: Workout metric: count timeframe: month target: 8]
py_athletics:
```

summarize_activities

```
Display a summary of Activities.

        If exercise is specified the listing is limited to that exercise.
        A timeframe for the listing can be established with one or both of the
        start and end keywords.

        Optional Parameters
        -------------------
        exercise: string = {Cycle|Run|Tennis|Walk|Workout}
        start: string in the form YYYY-MM-DD
        end: string in the form YYYY-MM-DD

        Examples
        --------
        summarize_activities
        summarize_activities exercise=Tennis
        summarize_activities exercise=Tennis start=2021-05-01 end=2021-06-30
```

Here is a summary activity report for March 2021.

```
py_athletics: summarize_activities start=2021-03-01 end=2021-03-31
Cycle   Summary: Activity Count:  6 Exercise Time (h:m:s):   9:56:50 Calories Burned:  3,741
Distance (miles):   145.93
Run     Summary: Activity Count:  7 Exercise Time (h:m:s):   3:01:17 Calories Burned:  2,258
Distance (miles):    17.55
Tennis  Summary: Activity Count:  9 Exercise Time (h:m:s):   8:46:31 Calories Burned:  4,736
Walk    Summary: Activity Count:  5 Exercise Time (h:m:s):   3:48:46 Calories Burned:  1,146
Distance (miles):    10.54
Workout Summary: Activity Count:  9 Exercise Time (h:m:s):   9:47:56 Calories Burned:  3,203
py_athletics:
```

summarize_goals

```
Display a summary of Goals.

        If exercise is specified, the listing is limited to that exercise.

        A goal summary shows the goal and how the athlete is tracking relative
        to the goal for the appropriate timeframe.  In addition, for monthly
        goals, summary information for prior months is also shown.

        Optional Parameters
        -------------------
        exercise: string = {Cycle|Run|Tennis|Walk|Workout}

        Examples
        --------
        summarize_goals
        summarize_goals Cycle
```

Here is a goal summary.

```
py_athletics: summarize_goals
Goal: Cycle 1,500 miles each year, year to date: 1,328.26 deficit: 171.74
Goal: Cycle 125 miles each month, month to date: 91.32 deficit: 33.68
      2021-09: 126.71 goal achieved with surplus: 1.71
      2021-08: 131.36 goal achieved with surplus: 6.36
      2021-07: 148.64 goal achieved with surplus: 23.64
      2021-06: 80.29 deficit: 44.71
      2021-05: 216.66 goal achieved with surplus: 91.66
      2021-04: 146.20 goal achieved with surplus: 21.20
      2021-03: 145.93 goal achieved with surplus: 20.93
      2021-02: 78.00 deficit: 47.00
      2021-01: 163.15 goal achieved with surplus: 38.15

Goal: Run 150 miles each year, year to date: 132.69 deficit: 17.31
Goal: Run 13 miles each month, month to date: 3.00 deficit: 10.00
      2021-09: 13.27 goal achieved with surplus: 0.27
      2021-08: 14.34 goal achieved with surplus: 1.34
      2021-07: 13.43 goal achieved with surplus: 0.43
      2021-06: 6.32 deficit: 6.68
      2021-05: 14.12 goal achieved with surplus: 1.12
      2021-04: 15.23 goal achieved with surplus: 2.23
      2021-03: 17.55 goal achieved with surplus: 4.55
      2021-02: 16.06 goal achieved with surplus: 3.06
      2021-01: 19.37 goal achieved with surplus: 6.37

Goal: Tennis 8 times each month, month to date: 3 deficit: 5
      2021-09: 7 deficit: 1
      2021-08: 7 deficit: 1
      2021-07: 10 goal achieved with surplus: 2
```

```
      2021-06: 7 deficit: 1
      2021-05: 8 goal achieved with surplus: 0
      2021-04: 5 deficit: 3
      2021-03: 9 goal achieved with surplus: 1
      2021-02: 8 goal achieved with surplus: 0
      2021-01: 11 goal achieved with surplus: 3

Goal: Workout 8 times each month, month to date: 2 deficit: 6
      2021-09: 3 deficit: 5
      2021-08: 5 deficit: 3
      2021-07: 5 deficit: 3
      2021-06: 3 deficit: 5
      2021-05: 5 deficit: 3
      2021-04: 9 goal achieved with surplus: 1
      2021-03: 9 goal achieved with surplus: 1
      2021-02: 7 deficit: 1
      2021-01: 9 goal achieved with surplus: 1
```

help

```
List available commands with "help" or detailed help with "help cmd".
```

shell

```
Run an OS shell command.

        Example
        -------
        ! ls test/
```

exit

```
Exit py_athletics.
```

Modules and Classes

The `py_athletics` module instantiates an instance of `PythonAthleticsShell` and calls the `cmdloop` method it inherits from the Python standard `cmd` class. That module, in its entirety is:

```python
"""py_athletics

Invoke with python py_athletics.py from a command line.
"""
from shell.shell import PythonAthleticsShell

if __name__ == "__main__":
    PythonAthleticsShell().cmdloop()
```

py_athletics is comprised of the following sub-modules:

- `activity` This module supplies the `Activity` base class and the `Cycle`, `Run`, `Tennis`, `Walk` and `Workout` subclasses. Documentation for the module and the classes it provides can be found here.
- `athlete` This module supplies the `Athlete` class. Documentation for the module and the classes it provides can be found here.
- `goal` This module provides the `Goal` base class and the `CumulativeGoal`, `MonthGoal` and `YearGoal` subclasses. Documentation for the module and the classes it provides can be found here.

- `helpers` This module contains a pair of modules with various utility functions.
    - `garmin_helpers` Documentation for the module can be found here here.
    - `helpers` Documentation for the module can be found here.
- `shell` This module provides the `PythonAthleticsShell` class. Documentation for the module and the class it provides can be found here.

A collection of all of the documentation referenced above can be found here. That same collection is also included as part of the **py_athletics** repository in both `html` and `md` format. See the `py_athletics/py_athletics/documents/modules/` directory.

In addition to the detailed descriptions in the documentation referenced above, there a few items of note.

- With the exception of `help`, `run_script`, `shell` and `exit`, the **py_athletics** commands map directly to `Athlete` methods.
- The `Athlete.read_garmin_activity_file` method parses the data we need from Garmin activity files. It is responsible for a great deal of cleanup and also for handling the fact that Garmin records speed for cycling in MPH and for running and walking in minutes/mile while using the same field key.
- Garmin Activity files are cumulative, so in order to avoid redundant entries, the `Athlete.add_activity` method only adds an `Activity` if it is not already present. However, when new `Goals` are added, `Athlete.add_goal` will replace old `Goals` with new ones.
- The `Activity.tally` method is at the heart of the summarization methods, `Athlete.summarize_goals` and `Athlete.summarize_activities`. It uses a `Counter` dictionary to aggregate the various relevant `Activity` data elements.

Test Data

py_athletics/py_athletics/test

- `Activities.csv` 2021 activity data through mid-October
- `goals.cmd` a collection of year and month goals that can be used to showcase the `run_script` command
- `goals_annual.cmd` a subset of `goals.cmd` that omits monthly goals
- `py_athletics.pickle` a sample saved **py_athletics** session that can be restored using the `load` command

py_athletics/py_athletics/test/garmin_data

A collection of CSV files representing cumulative activity date through the end of the month represented by the filename, i.e., `2021-01.csv` includes activity for January 2021, `2021-02.csv` contains activity for January and February etc.

Project Directory Tree (Expanded)

```
py_athletics
├── README.md
├── README.pdf
├── Reflections.md
├── Reflections.pdf
└── py_athletics
    ├── documents
    |   ├── classes.dot
    |   ├── classes.dot.png
    |   ├── modules
    |   ├── pdf-source-listings
    |   ├── py_athletics_design_concept.md
    |   └── py_athletics_design_concept.pdf
    ├── misc
    |   ├── generate_module_documentation.sh
    |   ├── generate_source_code_listings.sh
    |   └── lint_source_code.sh
    ├── src
    |   ├── __init__.py
```

```
    |   ├── activity
    |   ├── athlete
    |   ├── goal
    |   ├── helpers
    |   ├── py_athletics.py
    |   └── shell
    └── test
        ├── Activities.csv
        ├── garmin_data
        ├── goals.cmd
        ├── goals_annual.cmd
        └── py_athletics.pickle
```

Test Directory Tree

```
test
├── Activities.csv
├── garmin_data
|   ├── 2021-01.csv
|   ├── 2021-02.csv
|   ├── 2021-03.csv
|   ├── 2021-04.csv
|   ├── 2021-05.csv
|   ├── 2021-06.csv
|   ├── 2021-07.csv
|   ├── 2021-08.csv
|   └── 2021-09.csv
├── goals.cmd
├── goals_annual.cmd
└── py_athletics.pickle
```