

(https://profile.intra.42.fr)

SCALE FOR PROJECT PYTHON MODULE 04 (/PROJECTS/PYTHON-MODULE-04)

You should evaluate 1 student in this team



Git repository

git@vogsphere-v2-bg.1337.ma:vogsphere/intra-uuid-cd5c48cc-3



Comments

This fifth module is the evolution of two original projects created by the Paris-based student organization 42 AI.

They are named Bootcamp Python and Bootcamp Machine Learning.

active members of 42 AI re-designed both of them for the school curriculum.

Bootcamps has been developped between August 2019 and March/April 2020.

Active 42AI members organized severals sessions of 2 weeks to 42Paris students to offer them the possibility to get famliar with Python and basics concepts of machine learning.

The success of those sections brings the pedagogy to accept the idea to integrate the 2 bootcamps to the curriculum (initial discussion (01-05/2019) with 42 Paris pedago team highlighted a categorical opposition/refusal to this idea)

The transcription had been realized over the direction of Matthieu David.

Several 42AI members contributed to the redaction on the correction scales.

For futur corrections on the scale, please contact the 42AI association via contact@42ai.fr or the current 42AI pedagogical supervisor.

Introduction

The Bootcamp Python and Bootcamp Machine Learning were originally created by [42AI](https://github.com/42-AI) active members and were adapted to 'piscine' format for the school 42 curriculum.

For any issue or suggestion: [42paris_staff_pedagogy](https://42born2code.slack.com/archives/C7NF60E0Z) and

[42AI](https://github.com/42-AI/bootcamp_python/issues).

As usual, you have to observe the following courtesy rules:

- Remain polite, courteous, respectful, and constructive throughout the evaluation process. The well-being of the community depends on it.
- Identify with the evaluated person or group the eventual dysfunctions of the assignment. Take the time to discuss and debate the problems you may have identified.
- You must consider that there might be some differences in the understanding of and approach to project instructions, and the scope of its functionalities, between you and your peers. Always remain open-minded and grade them as fairly as possible. The pedagogy is valid only and only if peer-evaluation is conducted seriously.

The goal of this module is to get started with the library Pandas and the manipulation of dataframes.

Disclaimer

The serie of modules started to be produce at the time of the release of Python 3.7. Students are free to use later version of Python as long as they verified the producted code complies with all the aspects precised in the subjects.

As a consequence we recommend to students to perform the modules with the the Python version 3.7 (but this is just an advice).

Version can be checked with the command ``python -V``.

Guidelines

General rules

- Only grade the work that is in the student or group's GiT repository.
- Double-check that the GiT repository does belong to the student.
Ensure that the work is the one expected for the corrected exercise and don't forget to verify that the command "git clone" is run in an empty folder.
- Check carefully that no malicious aliases were used to make you evaluate files that are not from the official repository.
- To avoid any surprises, carefully check that both the evaluating and the evaluated students have reviewed the possible scripts used to facilitate the grading.
- If the evaluating student has not completed that particular project yet, it is mandatory for them to read the entire subject prior to starting the defense.

- Use the flags available on this scale to signal an empty repository, non-functioning program, a Norm error (specified next in general rules), cheating, and so forth.

In these cases, the grading is over and the final grade is 0, or -42 in case of cheating. However, except the exception of cheating, you are encouraged to continue to discuss your work even if the later is in progress in order to identify any issues that may have caused the project failure and avoid repeating the same mistake in the future.

- Use the appropriate flag.

- Remember that for the duration of the defense, no other unexpected, premature, or uncontrolled termination of the program, else the final grade is 0.

- You should never have to edit any file except the configuration file if the latter exists. If you want to edit a file, take the time to explain why with the evaluated student and make sure both of you agree on this.

- The Norm: The PEP 8 standards is not mandatory, but recommended.

- The function eval is never allowed.

- Your exercises are going to be evaluated by other students, make sure that your variable names and function names are appropriate and civil.

- Copiable code is included in the code_blocks.md in attachments.
(block xx.xx.xx) is an indication that the block of code with the same id on top should be used

Attachments

 code_blocks.md (/uploads/document/document/12500/code_blocks.md)

 subject.pdf (<https://cdn.intra.42.fr/pdf/pdf/66600/en.subject.pdf>)

 athlete_events.csv (/uploads/document/document/12501/athlete_events.csv)

Exercise 00 - FileLoader

The goal of this exercise is to create a Fileloader class containing a load and a display method. put the path to the csv file in an environment variable

Load and display methods tests

- (block 04.00.00)

- (block 04.00.01)
- (block 04.00.02)
- (block 04.00.03)
- (block 04.00.04)

☒ Yes☐ No

Exercise 01 - YoungestFellah

The goal of this exercise is to create a function that will return a dictionary containing the age of the youngest woman and the youngest man who took part in the Olympics a given year.

Basic tests

Check the results of the following cases:

The name of the function YoungestFellah may be changed to make this code run

(block 04.01.00)

If something does not match, the exercise is failed.

☒ Yes☐ No

Exercise 02: ProportionBySport

The goal of this exercise is to create a function displaying the proportion of participants who played a given sport, among the participants of a given genders.

Basic tests

Verify the results of the following cases: (please adapt the path to csv file)

(block 04.02.00)

The rounding of the result does no matter (0.023 or 0.02307969707897584 or 2.3 % are all acceptable)

If something does not match, the exercise is failed.

☒ Yes☐ No

Exercise 03: HowManyMedals

The goal of this exercise is to code a function that will return a dictionary of dictionaries giving the number and type of medals for each year during which the participant won medals.

Basic tests

Verify the results of the following cases: (Please adapt the path to csv file)

(block 04.03.00)

If something does not match, the exercise is failed.

☒ Yes

☐ No

Exercise 04 - SpatioTemporalData

The goal of this exercise is to write a class called `SpatioTemporalData` that takes a dataset (`pandas.DataFrame`) as argument in its constructor and implements two methods.

Basic tests

Check the result of the following cases:

(block 04.04.00)

You should try other locations and years.

If something does not match, the exercise is failed.

☒ Yes

☐ No

Exercise 05 - HowManyMedalsByCountry

The goal of this exercise is to write a function that returns a dictionary of dictionaries giving the number and type of medal for each competition where the country delegation earned medals.

Basic tests

Print the result of `how_many_medals_by_country` functions calls with various countries and check that the format respects the following:

(block 04.05.00)

☒ Yes

☐ No

Exercise 06 - MyPlotLib

The goal the exercise is to introduce plotting methods among the different libraries Pandas, Matplotlib, Seaborn or Scipy

Basic tests

Performs the following basic tests with the given dataset
with at least 3 numerical features:

- run the method histogram with one, two and three valid features
- run the method density with one, two and three valid features
- run the method pair_plot with one, two and three valid features
- run the method box_plot with one, two and three valid features

☒ Yes

☐ No

Exercise 07 - Komparator

The goal the exercise is to introduce plotting methods among the different libraries Pandas, Matplotlib, Seaborn or Scipy.

Basic tests

Verify you get the expected plots for each methods:

- giving 'Medal' and 'Age' for the method compare_box_plots, you should observe 3 boxes: Bronze. Silver and Gold
- giving 'Medal' and 'Height' for the method compare_histograms, you should observe 3 histograms
- giving 'Medal' and 'Weight' for the method density, you should observe 3 curves of density plot

☒ Yes

☐ No

Ratings

Don't forget to check the flag corresponding to the defense

☒ Ok

☐ Empty work

☐ Cheat

☐ Crash

☐ Incomplete group

☐ Forbidden function

Conclusion

Leave a comment on this evaluation

Finish evaluation

Terms of use for video surveillance (<https://profile.intra.42.fr/legal/terms/1>)

Rules of procedure (<https://profile.intra.42.fr/legal/terms/4>)

Declaration on the use of cookies (<https://profile.intra.42.fr/legal/terms/2>)

General term of use of the site (<https://profile.intra.42.fr/legal/terms/6>)

Legal notices (<https://profile.intra.42.fr/legal/terms/3>)

Privacy policy (<https://profile.intra.42.fr/legal/terms/5>)