

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

SCALE FOR PROJECT PYTHON MODULE 03 (/PROJECTS/PYTHON-MODULE-03)

You should evaluate 1 student in this team



Git repository

`git@vogsphere-v2.1337.ma:vogsphere/intra-uuid-ca131c0f-1620`

Comments

This fourth 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 Numpy.

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: You will follow the PEP 8 standards.
- 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.

Attachments

-  subject.pdf (<https://cdn.intra.42.fr/pdf/pdf/42655/en.subject.pdf>)
-  solar_system_census.csv (/uploads/document/document/7262/solar_system_census.csv)
-  42AI.png (/uploads/document/document/7263/42AI.png)
-  elon_canaGAN.png (/uploads/document/document/7264/elon_canaGAN.png)

Exercise 00 - NumPyCreator

Error management

First, verify the different methods are implemented in the class NumpyCreator.

The different methods should handle wrong type, ragged nested list/tuple.

Perform at least the following cases:

- 'npc.from_list("toto)". The method returns None.
- 'npc.from_list([[1,2,3],[6,3,4],[8,5,6,7]])'. The method returns None.

- 'npc.from_tuple(3.2)'. The method returns None.
- 'npc.from_tuple(((1,5,8),(7,5)))'. The method returns None.
- 'npc.from_shape((-1, -1))'. The method returns None.
- 'npc.identity(-1)'. The method returns None.

☒ Yes

☐ No

Basic tests

Check if all the following cases have a correct behaviour:

- 'npc.from_list([[[]], [[]]])'
- 'npc.from_list([[1,2,3],[6,3,4],[8,5,6]])'
- 'npc.from_tuple(("a","b","c"))'
- 'npc.from_iterable(range(5))'
- 'npc.from_shape((0, 0))'
- 'npc.from_shape((3, 5))'
- 'npc.random((3, 5))'
- 'npc.identity(4)'

☒ Yes

☐ No

Exercise 01 - ImageProcessor

Implementation

- Verify the class ImageProcessor and that the methods 'load' and 'display' are implemented.
As a class, special method '__init__' has to be present.

- The method 'load' handles empty and non existing file.
- The method 'display' handles non array-like, empty or None argument.

☒ Yes

☐ No

Exercise 02: ScrapBooker

Error managment

First, verify the class is correctly implemented and has the 4 methods and the constructor ('__init__').

Verify the different methods handles incorrect parameters:

- Non numpy.array for arr parameter. The functions return None.

- dim different than a tuple of 2 acceptable integers for 'crop' and 'mosaic'. The functions return None.
- position being different than (int1, int2) for 'crop'. The function returns None.
- axis parameter for 'thin' and 'juxtapose' different than 0 and 1. The functions return None.

Verify the other combinaison of incorrect parameters.

☒ Yes

☐ No

Basic tests

Perform basic tests to verify the proper behaviour of the different methods.

☒ Yes

☐ No

Exercise 03: ColorFilter

Error managment

First, verify the 6 methods plus the constructr are implemented in the class 'ColorFilter'.

As usual, check the error managment of the different methods.

☒ Yes

☐ No

Basic tests

Test the different methods, by taking a picture of your choice and applying the filters.

You should observed the 'to_green' method generate a new array corresponding to the original image with a green filter, and so on.

☒ Yes

☐ No

Exercise 04 - K-means Clustering

Implementaion & Basic tests

- First verify the correct implementation of the class (constructor and methods).
- The centroids are randomly initialized.
- Program can handles different set of values (raisonnable ones) for 'ncentroid' and 'max_iter' arguments.
- For a fix value of 'ncentroid', centroids coordinates slightly differ when running the algorithm multiple times.

- Results are coherent and consistent through multiple program execution.
Most of the time, centroids converged to similar coordinates and number people associated to each centroid varies little.

☒ Yes☐ No

Ratings

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

☒ Ok☐ Empty work☐ Norme☐ Cheat☐ Crash☐ Incomplete group☐ Forbidden function

Conclusion

Leave a comment on this evaluation

Finish evaluation

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

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

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

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

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

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