Rush 00 – Piscine Python for Data Science

MovieLens Analytics

Summary: This rush will help you to strengthen the skills acquired in the previous days

Contents

Chapter I

Foreword

Why do we like movies? What makes them so attractive to us?

Even though the movie is a relatively modern thing for humanity, it has a pretty old mechanism inside - it is the story.

People have loved stories since ancient times. Think about them as a universal container that effectively transfers some useful information from a source to a person.
By sparkling emotions and imagination in us, it establishes a good connection and packages information in a way that can be easily consumed by a human being.
Stories were crucial for surviving to our ancestors.
Stories contain the personal experience that can be applied to your life. For example, you may discover that some areas around your village are pretty dangerous.
Or there are some really good places to gather mushrooms.

Our attention to stories has survived through the centuries.
If a speaker starts their presentation by telling a story, they catch our attention. We love books. We love music and songs.
We love movies.

How can you use stories in data science? Good reports have elements of storytelling. Try to tell a story by your analysis

Chapter II

Instructions

- Use this page as the only reference. Do not listen to any rumors and speculations about how to prepare your solution.
- Here and further we use Python 3 as the only correct version of Python.
- The python files for python exercises (module01, module02, module03) must have a block in the end: if __name__ == ‘__main__’.
- Pay attention to the permissions of your files and directories.
- To be assessed your solution must be in your GIT repository.
- Your solutions will be evaluated by your piscine mates.
- You should not leave in your directory any other file than those explicitly specified by the exercise instructions. It is recommended that you modify your .gitignore to avoid accidents.
- When you need to get precise output in your programs, it is forbidden to display a precalculated output instead of performing the exercise correctly.
- Have a question? Ask your neighbor on the right. Otherwise, try with your neighbor on the left.
- Your reference manual: mates / Internet / Google.
- Remember to discuss on the Intra Piscine forum.
- Read the examples carefully. They may require things that are not otherwise specified in the subject.
- And may the Force be with you!

Chapter III

Specific instructions of the day

- No code in the global scope. Use functions!
- Any exception not caught will invalidate the work, even in the event of an error that was asked you to test
- The interpreter to use is Python 3
- Any built-in function is allowed
- You can import the following libraries: os, sys, urllib, requests, beautifulsoup, json, pytest, collections, functools, datetime, re
- Use Jupyter Notebook for creating the report

Chapter IV

Mandatory part

In this rush, you are going to work on your own analytical report.
You will analyze data from the MovieLens database.
By the end of the rush, you will have two files:
movielens_analysis.py and movielens_report.ipynb.
In the first file, you will need to create your own module with classes and methods.
In the second file, you will create the report itself using only your module.

Module

Remember that the goal of the rush is to strengthen your skills.
Try to use as much as you can from what you have learned from the previous days.

- Use a smaller version of MovieLens dataset, download it, please
- Read the README.txt very carefully. Focus on the file structures
- In your module, you will need to create 4 classes corresponding to 4 files from the data and 1 class for testing
- The classes and methods below are obligatory but you can add to them anything that suits your needs

⚠ Class Ratings, Tags, Movies, Links can be found in attachments

Class Tests:

Create tests using PyTest for each and every method of the classes above.
They should check:

- if the methods return the correct data types

- if the lists elements have the correct data types
- if the returned data sorted correctly

Run the tests before going to the next stage of the rush.

Report

Using only the classes and methods from movielens_analysis.py,
prepare your report.
You should do it in Jupyter Notebook.
It is a great tool especially if you are a data scientist.
It gives you an opportunity to work with the code interactively
by launching and relaunching different cells with different values.
You do not have to rerun your whole code from the beginning.
Also, you can put in the cells not only code but text too,
which is a great feature for making reports.
Install it to your environment.

In this part of the rush, we will give you more freedom.
We are not going to define the structure of your report.
The goal of the report is to tell us an interesting story
about the MovieLens dataset.
Find the good structure and the right sequence.
The only constraints:

- you must use each and every method from movielens_analysis.py except class Tests
- every cell in your notebook should contain magic command %timeit
- all other imports are prohibited as well as using built-in functions. If you need them, put them in your module in advance

Chapter V

Bonus part

- Add to the classes more methods that you may find useful and interesting for your report. Do not forget to test them too
- Improve the tests. Check the correctness of your calculations as well. Precalculate manually some results and metrics and check if the methods return the correct information if you give them the specific input