Introduction to Python & Programming

Hacker Night 1

Gaël Guibon
Essec - GBBA - A1
27/10/2020 16:30 - 22:45

Subject

Browse and Query Movies

Objectives

- Create a program that will handle Movie data

1. Implement **required actions**
2. Merge them into a terminal interface (`input()` function)
3. Be creative and create **creative actions**

Required Actions: todo list

Todo List:

1. get the total number of movies
2. get a movie by its title (**title**)
3. get titles of all rumored movies (**status**)
4. get all the different statuses possible (**status**)
5. get the title of the movie with the maximum number of votes (**vote_count**)
6. get the maximum number of votes (**vote_count**)
7. get the movie with the highest revenue (**revenue**)
8. get the movie with the highest budget (**budget**)
9. get the movie with highest difference between revenue and budget (**revenue**, **budget**)

10. get the number of movies per original language (**original_language**)
11. get the number of movies per genre (**genres**)
12. get the frequencies of genres associated with the comedy genre (**genres**)
13. get titles of the movies with a modified title (**original_title**, **title**)
14. get the total number of movies with a modified title (**original_title**, **title**)
15. get the titles of all movies from a specific country (`'United Kingdom'` for instance)
16. get the total number of violent movies
17. get the percentage of violent movies
18. get the average movie duration (**runtime**)

Required Actions: textual interface and export

- Each todolist item should be able to print result or export them into a file
- Each todolist item should be triggered through a terminal query interface

Creative (bonus) Actions

Creative actions example ideas:

- Add a **seen** field to indicate if you saw the movie or not
- Add one of your favorite movies that is missing (examples: '2046', 'Matrix', *etc.*)
- Create a likely movie taking into account the most frequent information (genre, country, language, etc.)
- Create your own movie from scratch ! And add it into the JSON file
  - Add custom metadata to be able to easily find your custom movie(s)
- Stats queries : show all kinds of information from basic stats (counts, average, etc.)

You have a superb novel idea? **Just do it**

Format

- **Small groups** (max 3 students)
- **Scheduled breaks** (snack, nap)
  - 17h45-18h15
  - 19h30-20h
  - 21h15-21h45
- **Work presentation** starting from 21h45 to 22h45 (online, screen sharing)
- Otherwise **stay in zoom**

Evaluation

- Project presentation (last hour)
- Diversity of notions used
- Todolist completion
- Creativity actions
- Code clarity and good practices

Project delivery by email before leaving/presenting

Advises

- Use local **python install** + **visual studio code**
- Use `pprint.pprint()` to better print a dictionary

```
from pprint import pprint
myDict = {'greeting': 'hello A1'}
pprint(myDict)
```

- A problem with your machine set up? Ask for help!
  - ...but for the code you are on your own.

Good luck!