December 10, 2024

Python Advanced Course

List 10.

Please program the following tasks using a relational database (e.g. SQLite). Solutions should meet the following conditions:

- the program should contain at least three classes (tables), without additional tables used to build many-to-many relationships;
- data creation and access is to be performed using the ORM layer, e.g. using SQLAlchemy1 discussed during the lecture ;
- relationships between data (e.g. one-to-many) should be explicitly described in data declarations;
- table definitions should contain at least one validation;
- prepare test data (e.g. in json format) that will be automatically pulled in when initializing a new database;
- the user interface is to call the program with the appropriate arguments there, e.g.

```
python3 myprogram.py meetings --add --description=Lectures python3
myprogram.py meetings --list
```

Assume that for each model there is at least the ability to add, update and search.

To analyze the arguments of the program call, use ready-made packages,2 e.g. argparse or getopt. Also take care of the information about the arguments of the program call, e.g. after providing the argument —help.

Task 1.

Write a program that stores information about books you have (at least author, title, year) and a list of friends (at least name, email) who borrow books from you. Also program the operations of adding new books, borrowing books, and returning them.

Task 2.

Program a system that stores information about films (this can also be other pieces, e.g. music). The data to be stored is a description of the film itself (title, year of creation) and the people involved in its creation (director, cameraman, producer).

Task 3.

Program your own calendar that stores events (start and end times, description), e.g. classes, meetings, etc., along with a list of people (name, email) who are assigned to the event. When adding events, inform whether the event does not clash with an already saved event.

For class, you must complete one of these tasks. Each task is worth 5 points. This task will be developed in subsequent task lists.

Marcin Mÿotkowski

[1] other ORMs: e.g. peewee, Django ORM or Pony ORM

2using argument parsing tools is part of the task, for manual handling the score may be lowered