Code to process data exported from EDT (school timetable software)

Luca Manini*

14 ottobre 2020

1 Introduction

The code in this repo concerns the EDT software used in many schools to create and manage the class schedule (teachers, classes, subjects, lessons, etc.).
EDT exports its data in various ways and formats, the more useful one being a full export of all lessons in CSV format and UTF-16 (!) enconding.
From that data you can extract, process, format and output a lot of other "tables" like the full timetable (one row for each teacher, one column for each hour, with the class code in each cell) ot the class timetable (one column per day, one row per hour and subject and teacher info in each cell) and so on.

2 Code structure

Code structure will change during development and even more when we will implement some sort of "packaging and distribution" (github, zip, docker. . .).
The basic idea is to have:

1. one dir with input data (data)
2. one dir with output data (out)
3. one common module (odv)
4. some "programs" (odv-full-timetable, odv-class-timetable, etc.)

Eventually there will be some more docs (docstring) and some testing (doctest).

3 Use

3.1 Command line interface

All "programs" can be used "from the command line", that is writing a command at the prompt (usually indicated with a dollar sign). Being Python programs, thay have to be interpreted by the python program. For example, if the file `odv-class-timetable` is the current directory:

```
$ python odv-full-timetable
```

All programs accept arguments (usually filenames) and options. Among the options there is always a a **-h** or **–help** one that gives come information on how to run the program and which arguments and options it accepts. For example:

```
$ pyhton odv-class-timetable --help
usage: odv-class-timetable.py [export-csv-file]
```

3.2 Graphical user interface

May be in the future?

3.3 Input output

Almost all programs use the data exported from the EDT software; the path to the export file can be given on the command line and the default path is (at the moment) `data/export.csv`.
File produces by the program are usually written to the `out` directory (or some subdir of it).
A the momento there is no way to change those paths from the command line, that will surely change in the future.

3.4 Debug messages

All programs write debugging information during execution. Messages are written to the *standard output*, so they will appear ont the "screen". You can just ignore them but they can be useful when reporting problems. Example:

```
$ python odv-class-timetable.py
DEBUG: Reading input file 'data/export.csv'
DEBUG: 1867 rows found
DEBUG: 110 classes found
DEBUG: Writing output CSV file 'out/class-timetable-csv/class-timetable.csv'
```