The context

We are a tutoring company specialized in the international baccalaureate. Our goal is not only to give good tutoring lessons, but also create IT tools to help the students. We realized one of the problems the students have is that they don't organise enough their revisions. Hence came the idea of creating a customized timetable app for them on our website.

The projects

1st project: create a week calendar

2nd project: create a GUI for the week calendar

3rd project: create a year calendar adapted to the exams deadlines

Programs used: python, VBA for Excel sheets

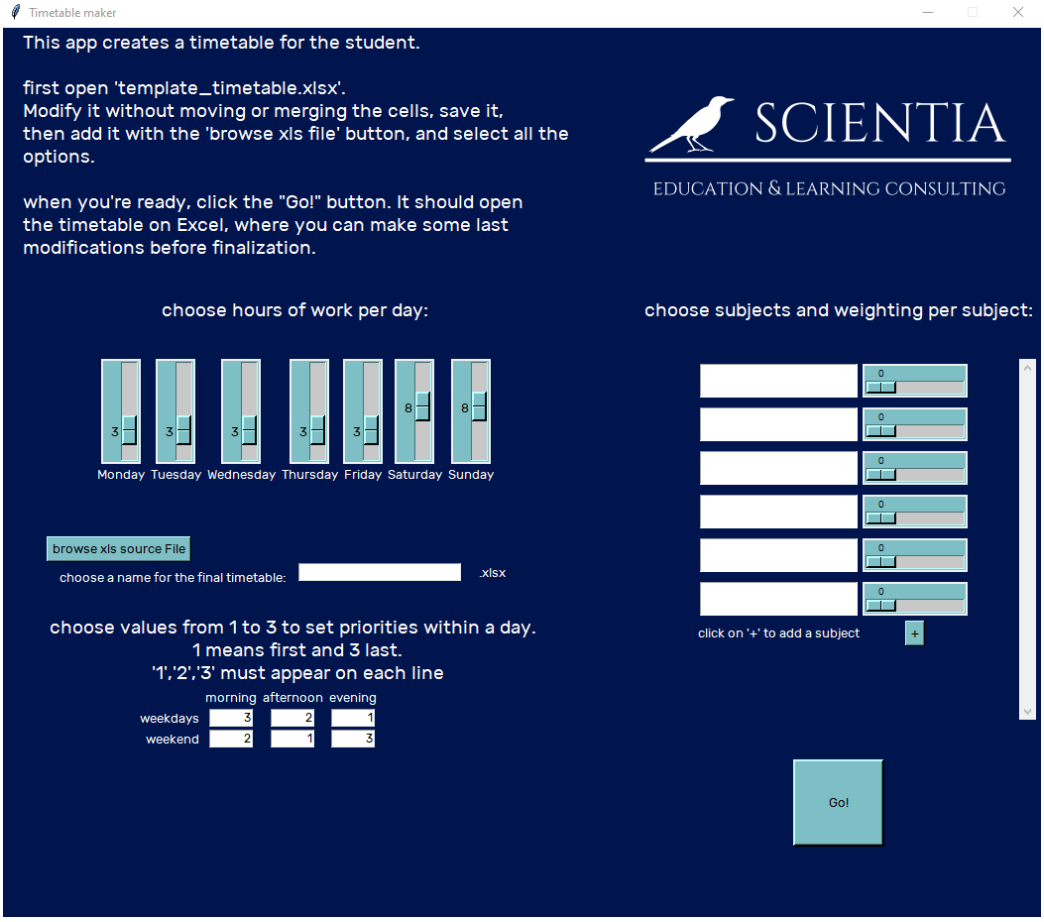
Week calendar

week calendar (main function "week_calendar_for_student") creates a pandas dataframe and an excel sheet containing revision hours per day (a macro creates the layout):

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday	
08:00								08:00
09:00						chemistry	chemistry	09:00
10:00						history	maths	10:00
11:00								11:00
12:00								12:00
13:00			physics			physics	physics	13:00
14:00								14:00
15:00								15:00
16:00			physics					16:00
17:00	maths	physics	maths	physics	physics	biology	biology	17:00
18:00	physics				maths			18:00
19:00								19:00
20:00								20:00
21:00	chemistry	biology	chemistry	geography	chemistry			21:00
22:00								22:00
23:00								23:00

The inputs from the student are explained in the GUI section

GUI



Left part, from top to bottom:

Cursors to select the number of hours of revision work per day

File browser to get original calendar from student (it has to respect the structure of template_calendar.xls, available in week_calendar/xls_files)

Name for the final timetable

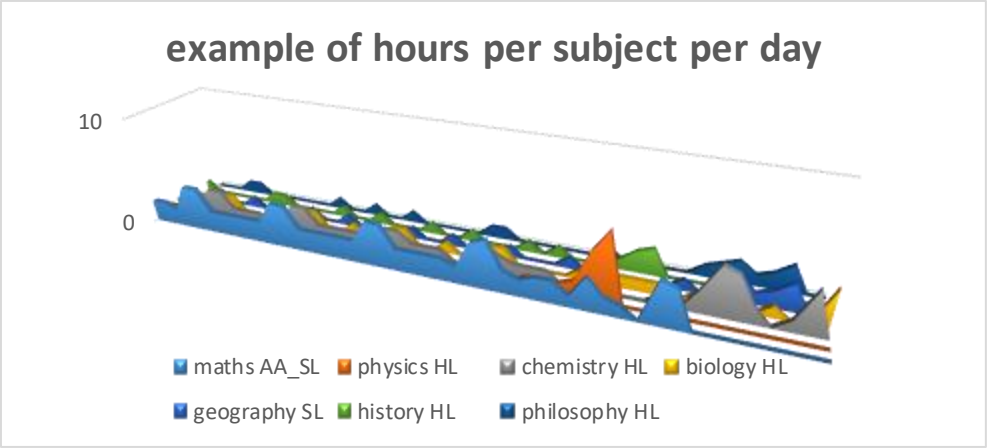
Order of priority for morning, afternoon and evening (3 2 1 means priority on evening, then afternoon and then morning)

Right part:

Type subject name and weighting for the subject

Year calendar

The year calendar creates n week calendars (for n the number of weeks before last exam) and puts them in an excel workbook. The number of hours spent on a given subject is a function of the coefficient chosen by the student and the remaining days before the exam (see illustration below, here physics end first). I used simulations with an objective function to try to have to correct number of hours per subject throughout the year.



Quick way to test the programs

To test the week calendar, run week_calendar\python_files\use_week_calendar_for_students.py

To test the GUI, run GUI\GUI.py

To test the year calendar, run year_calendar\secondary_files\test_on_year_calendar.py

The folders

The main folder contains *Archives*, *info_on_scripts* and *Scripts* folders

Archives

Not useful for current project, but kept here in case some of the old codes are needed

Info_on_scripts

Contains this document, and a folder *explanation* containing tree diagrams for *week_calendar_for_student* and *year_calendar_for_student* main functions

Scripts

Contains the three projects: *GUI*, *week_calendar* and *year_calendar*

GUI

Not much to say her

Week_calendar

The folder *excel_templates* contains xls files that the student fills with their school schedule

The folder *python_files* contains all the python modules involving *week_calendar*

The folder *secondary_files* contains python tests on the *python_files* modules

Year_calendar

The folder *python_files* contains all the python modules involving *year_calendar*

The folder *secondary_files* contains python tests on the *python_files* modules

The folder *xls_files* contains xls files made from different tests on the *year_calendar* module