

FET Timetabling Software

Beginner User Guide

A practical, step-by-step guide to building your first working timetable in FET.

1. What FET is

FET (Free Timetabling Software) is a free, open-source program that **automatically builds school and university timetables for you**. You describe your week and your lessons; FET's solver finds a clash-free arrangement that places every lesson into a day, an hour, and a room — usually in seconds.

- **Free & open-source** (AGPL licence), and it runs on **Windows, macOS, and Linux**.
- You work in a single project file with the **.fet** extension.
- When it's done, FET can **export your timetable as HTML and CSV** — ready to print, share, or publish — with separate views for each teacher, class, and room.

A timetable in FET is built from eight kinds of information: **Days, Hours, Teachers, Students, Subjects, Activities, Rooms, and Constraints**.

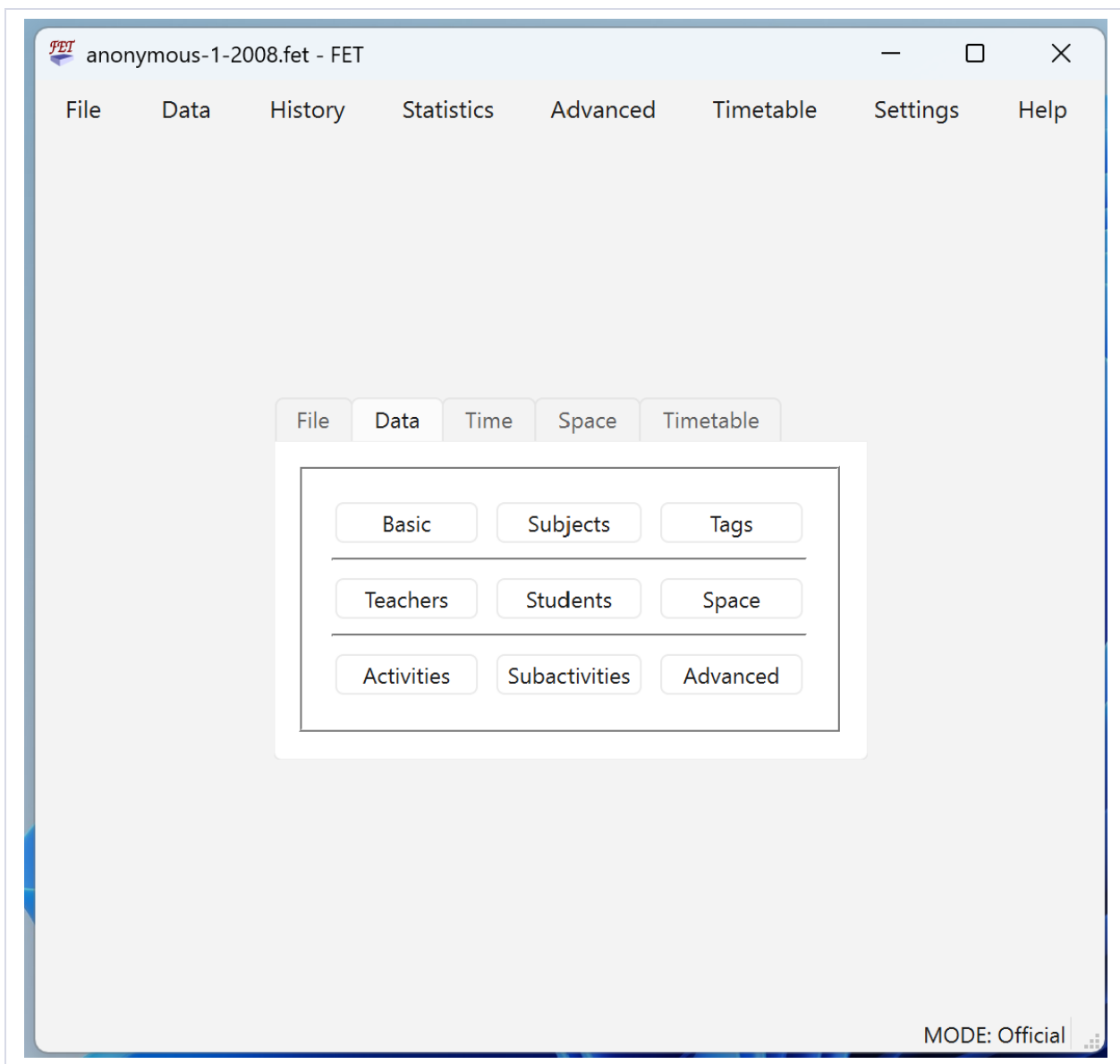


Figure 1. The FET main window on a fresh file — the menu bar and the Data tab.

2. The one big idea: Activities

If you remember only one thing, remember this:

An Activity is one lesson to be scheduled. It links a Subject, the Teacher(s) who teach it, the Student group(s) who attend, and a duration (how many hours).

Everything else exists to support activities. Without activities there is nothing for FET to schedule — so the teachers, subjects, and classes are really just the ingredients that activities combine.

- An activity can have **more than one teacher** (co-teaching) or even **none** (e.g. supervised study).
- A weekly lesson that meets several times (say Maths three times a week) is entered as one activity that you **split** into three sessions — FET keeps them apart for you.

Add activity

Teachers

Teacher 1
Teacher 2
Teacher 3
Teacher 4
Teacher 5
Teacher 6
Teacher 7
Teacher 8

Selected

Activity tags

Activity tag 1
Activity tag 2

Selected

Subject

Students

Students year 1
Students group 1_1
Students group 1_2
Students group 1_3
Students group 1_4
Students group 1_5
Students group 1_6
Students year 2

Clear

☒ All
☐ Qualified
☐ Separately

Duration

1

Split

1

☐ Half

Min days

1

Weight %

95.0

☒ Consecutive

☒ Years ☒ Groups ☐ Subgroups

Add **Help** **Close**

Figure 2. The 'Add activity' dialog: Teacher, Students, Subject and Duration.

3. The FET building blocks

Block	What it is
Days	Your teaching days, e.g. Monday–Friday.
Hours (Periods)	The periods in a day, e.g. Period 1–Period 6.
Subjects	What is taught — Mathematics, English, Science...
Teachers	Who teaches.
Students	Who attends — organised as Years → Groups → Subgroups (e.g. Year 7 → 7A → 7A-boys). A 'year' is shorthand for all the groups beneath it.
Rooms	Where lessons happen; each room has a capacity, and rooms can be grouped into Buildings.
Activities	The lessons to schedule (see §2).
Constraints	The rules FET must respect — availability, gaps, daily limits, preferred rooms, and so on.

Optional but handy: Activity tags are labels you can stick on activities — e.g. "lab" or "exam" — so one rule can target a whole group of lessons at once. You can ignore them at first.

4. Minimum requirements

Before FET can generate anything, you need at least:

- **Days**
- **Hours (Periods)**
- **Student sets (classes)**
- **Activities**

Teachers and **Subjects** can be created **automatically** while you import activities (see §9) — but **student sets are the exception: you must define or import them first**. Every activity needs a class to attend, and FET can't invent your Years → Groups → Subgroups structure from an activity list.

5. The recommended workflow

1. **Create a new FET file** (choose the Official mode).
2. **Define Days**.
3. **Define Hours**.
4. **Define your Student sets (classes)** — by hand, or import them (Data → Students). Do this **before** importing activities.
5. **Import Activities** (recommended) — or add them by hand.
6. **Review** the Teachers and Subjects FET created for you.
7. **Add Rooms**.
8. **Add Constraints**.
9. **Generate** the timetable — then view and export it.

6. Step 1 — Create a new file

Choose **File → New**. FET asks which **mode** to use. For a normal weekly school timetable, pick **Official** — it's the standard mode and what this guide assumes. (The other modes — Mornings-Afternoons, Block planning, Terms — are for special calendars; explore them later in the full guide.)

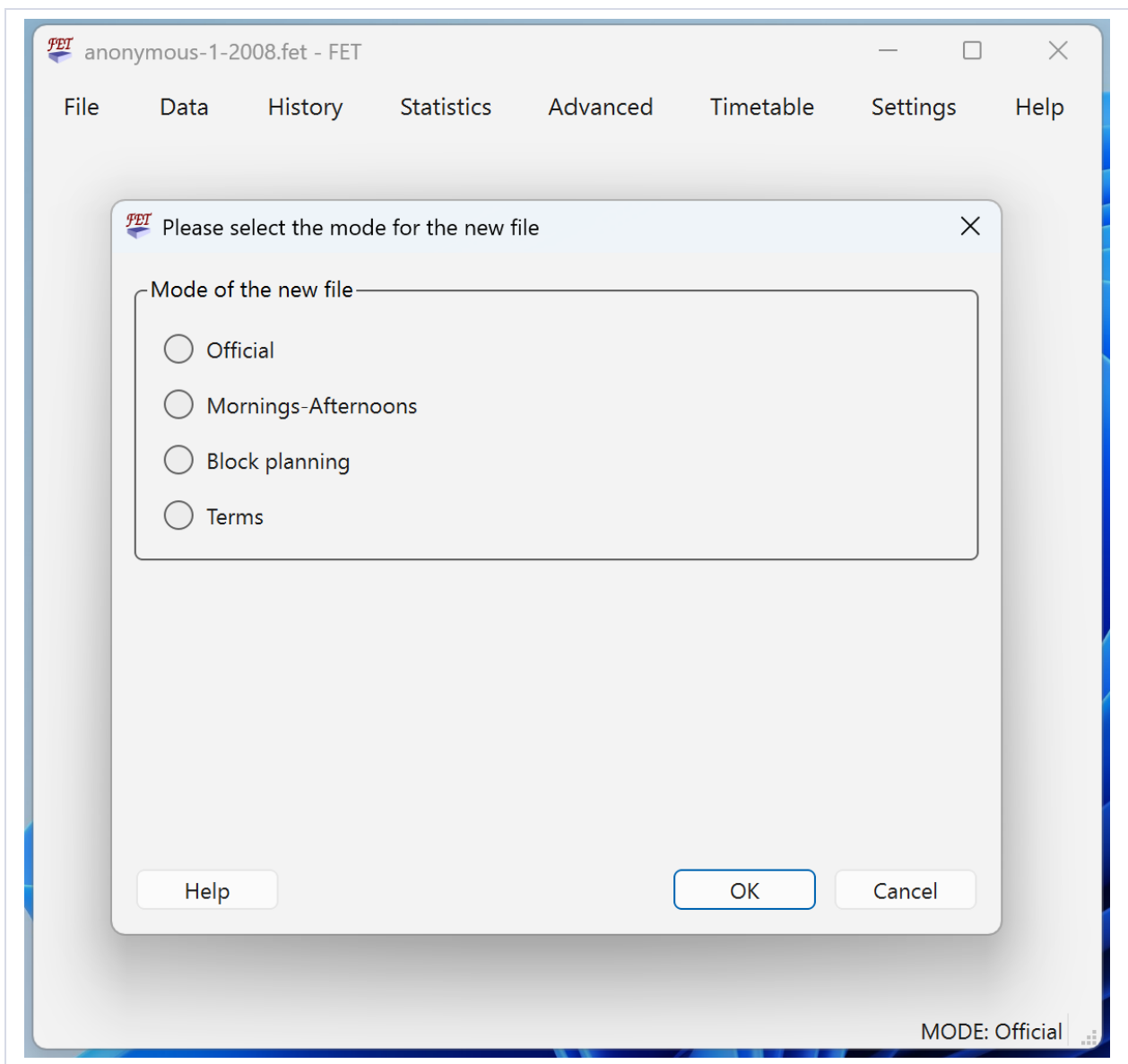


Figure 3. Selecting the Official mode when creating a new file.

7. Step 2 — Days and Hours

These two lists define the empty grid that every lesson drops into, so set them first.

- **Days:** Monday–Friday.
- **Hours (Periods):** Period 1–Period 6.

Open **Data** → **Basic** → **Days and hours**. FET starts you with a sensible default that you can rename or resize. Getting these right early matters — changing them after you've built activities is disruptive.

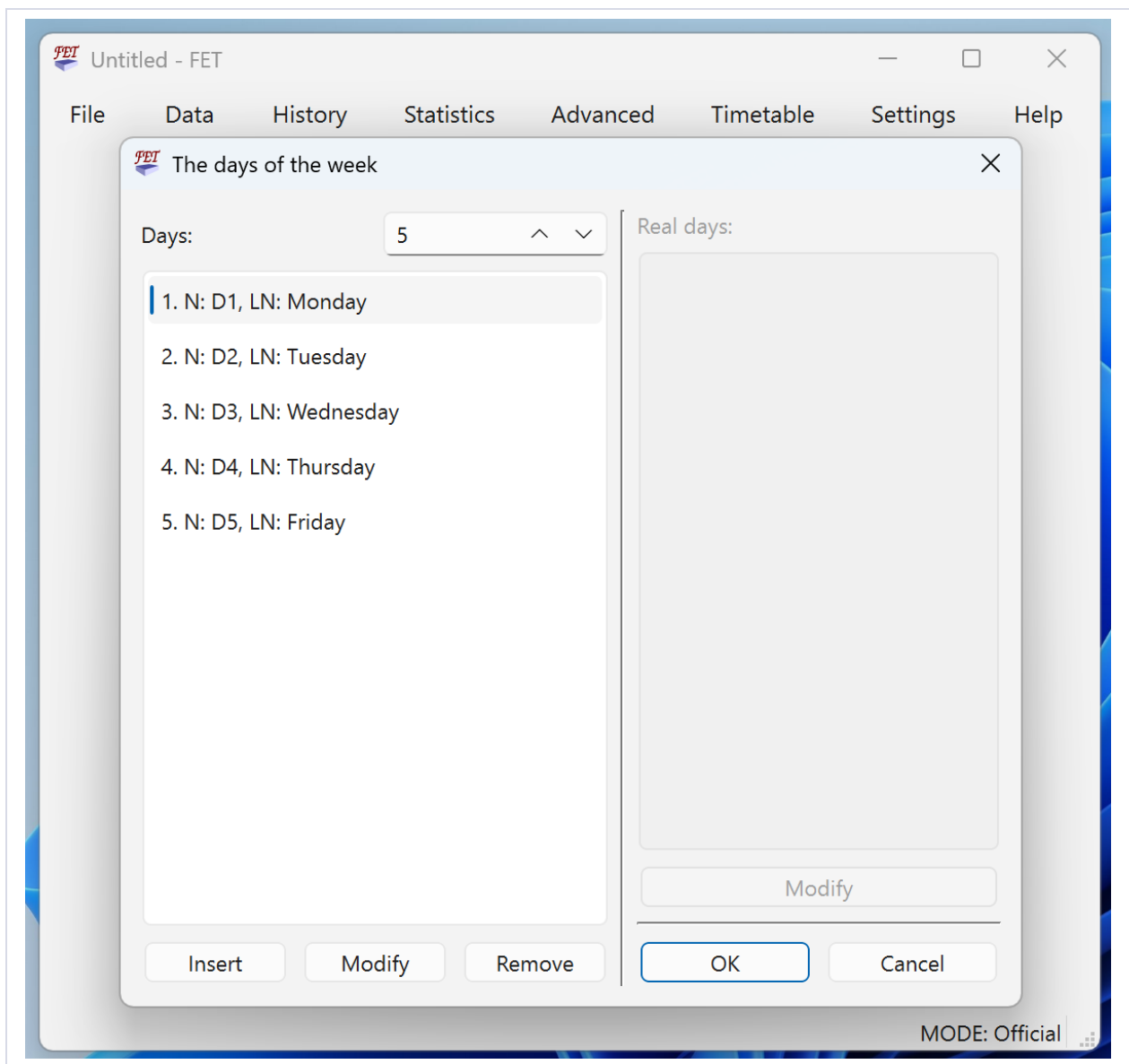


Figure 4. 'The days of the week' — Monday to Friday.

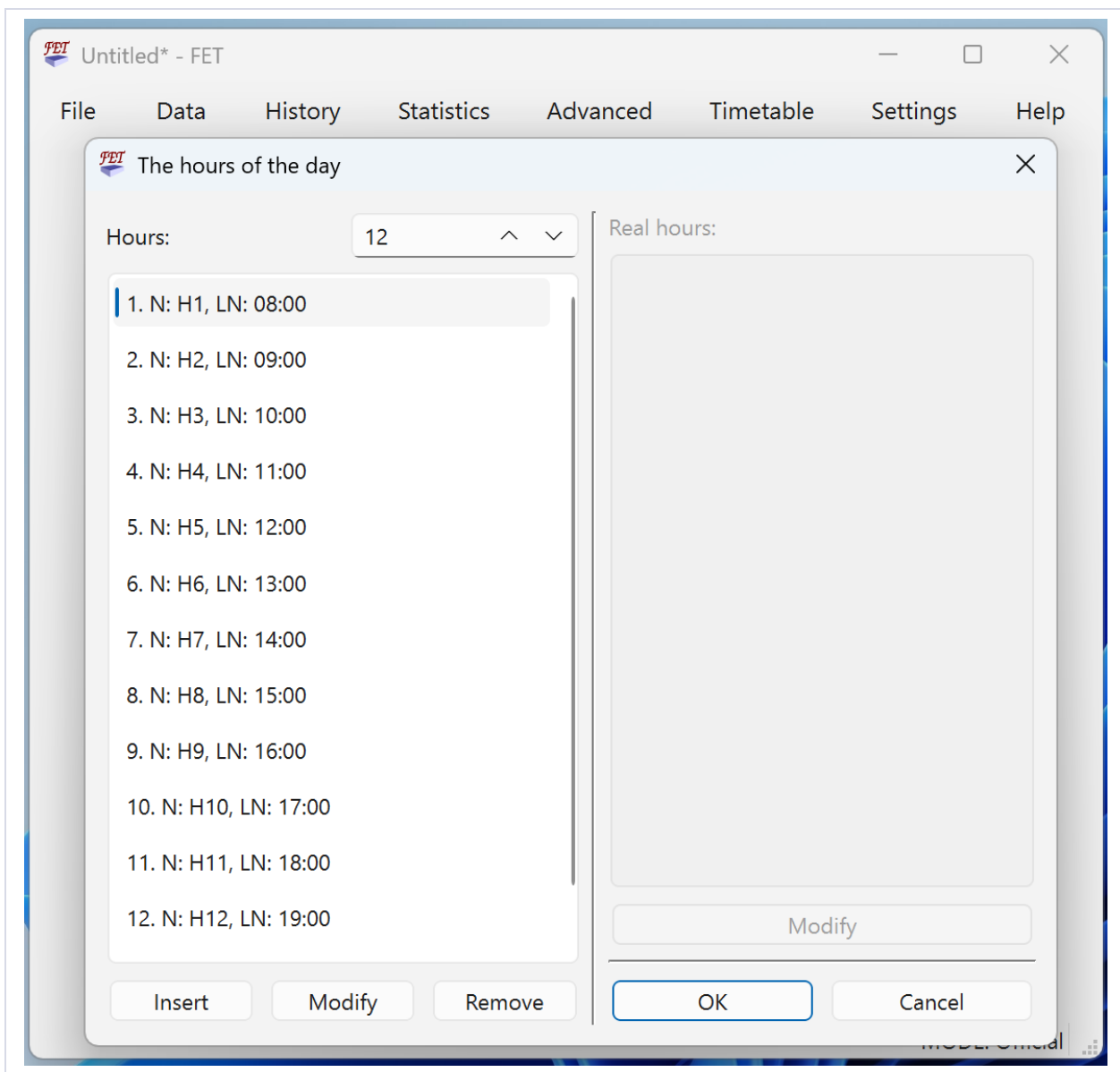


Figure 5. 'The hours of the day' — the periods of your day.

8. Step 3 — Teachers, Students, and Subjects

These three answer who teaches, who attends, and what is taught.

- **Subjects** — add them first (e.g. Mathematics, English, Science), because teachers can be linked to the subjects they teach.
- **Teachers** — the staff who deliver lessons.
- **Students** — remember the Years → Groups → Subgroups structure. A small school might have just Year 7 split into groups 7A and 7B.

You can type these in by hand under **Data → Subjects / Teachers / Students**. Teachers and Subjects can also be created for you during activity import (next step) — but enter your **Students** here first, because activity import won't create them.


Teachers

Teacher 1
Teacher 2
Teacher 3
Teacher 4
Teacher 5
Teacher 6
Teacher 7
Teacher 8
Teacher 9
Teacher 10
Teacher 11

Teacher
Name=Teacher 1
Long name=
Code=
Number of active hours=6
Target number of hours=0
Qualified subjects:

Time constraints directly related to this teacher:

Time constraint
All teachers must respect the maximum number of gaps per week
(breaks and teacher not available not counted)
Weight (percentage)=100%
Maximum gaps per week=1

Time constraint

☐ Colors

Add
Modify
Remove

Long name
Code
Comments

Target hours
Qualified subjects

Activate all activities for selected teacher

Deactivate all activities for selected teacher

Up
Down
Sort
Close

Figure 6. The Teachers list.

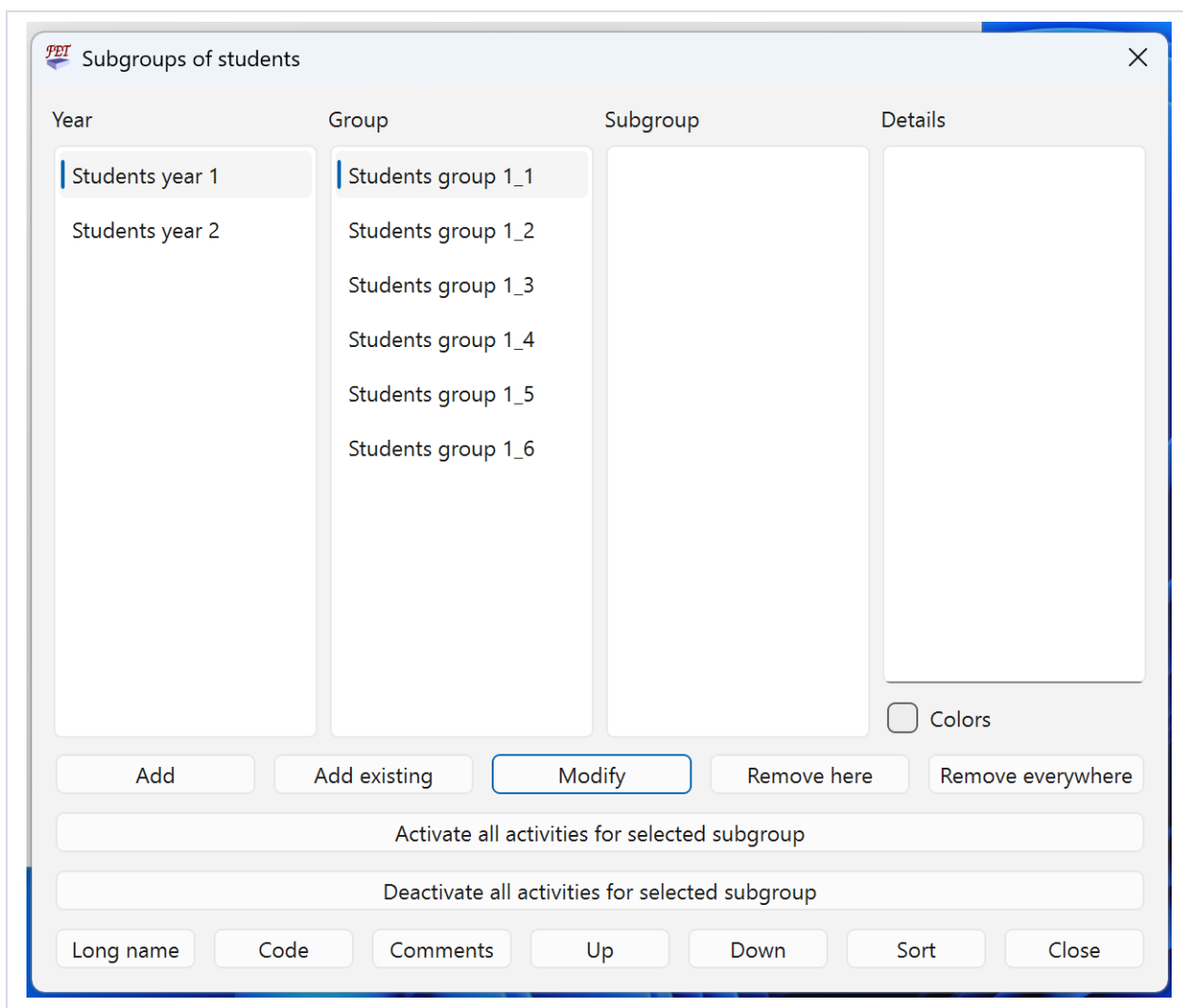


Figure 7. The Students view — a Year with its Groups.

9. Step 4 — Importing Activities

Plan before you build

Before you enter activities, sketch them on a spreadsheet and sanity-check: does every class have the right subjects and number of hours, and can each teacher realistically cover their load? The hardest timetables are the nearly-full ones — five minutes of planning here prevents an “it won't generate” headache later.

Most schools already keep their lessons in a spreadsheet with columns such as:

Teacher	Students	Subject	Duration	Lessons / week
Mr Smith	7A	Mathematics	1	4
Ms Jones	7A	English	1	3

When you import this list (**Data → Activities → Import**, or **File → Import**), FET will **automatically create** anything that doesn't exist yet:

- Teachers
- Subjects
- Activity tags
- Activities

One exception: student sets

FET does **not** create your classes from an activity import — define your **Student sets** first (Data → Students, or a separate students CSV), then import the activities that reference them.

This is usually the fastest way to build a timetable database — one import can bootstrap most of your roster. After importing, glance back at your Teachers and Subjects lists to confirm they look right.

FET's own activity columns are a little more detailed — **Students Sets, Subject, Teachers, Activity Tags, Total Duration, Split Duration, Min Days, Weight, Consecutive** — but the import wizard lets you map your spreadsheet's columns onto them.

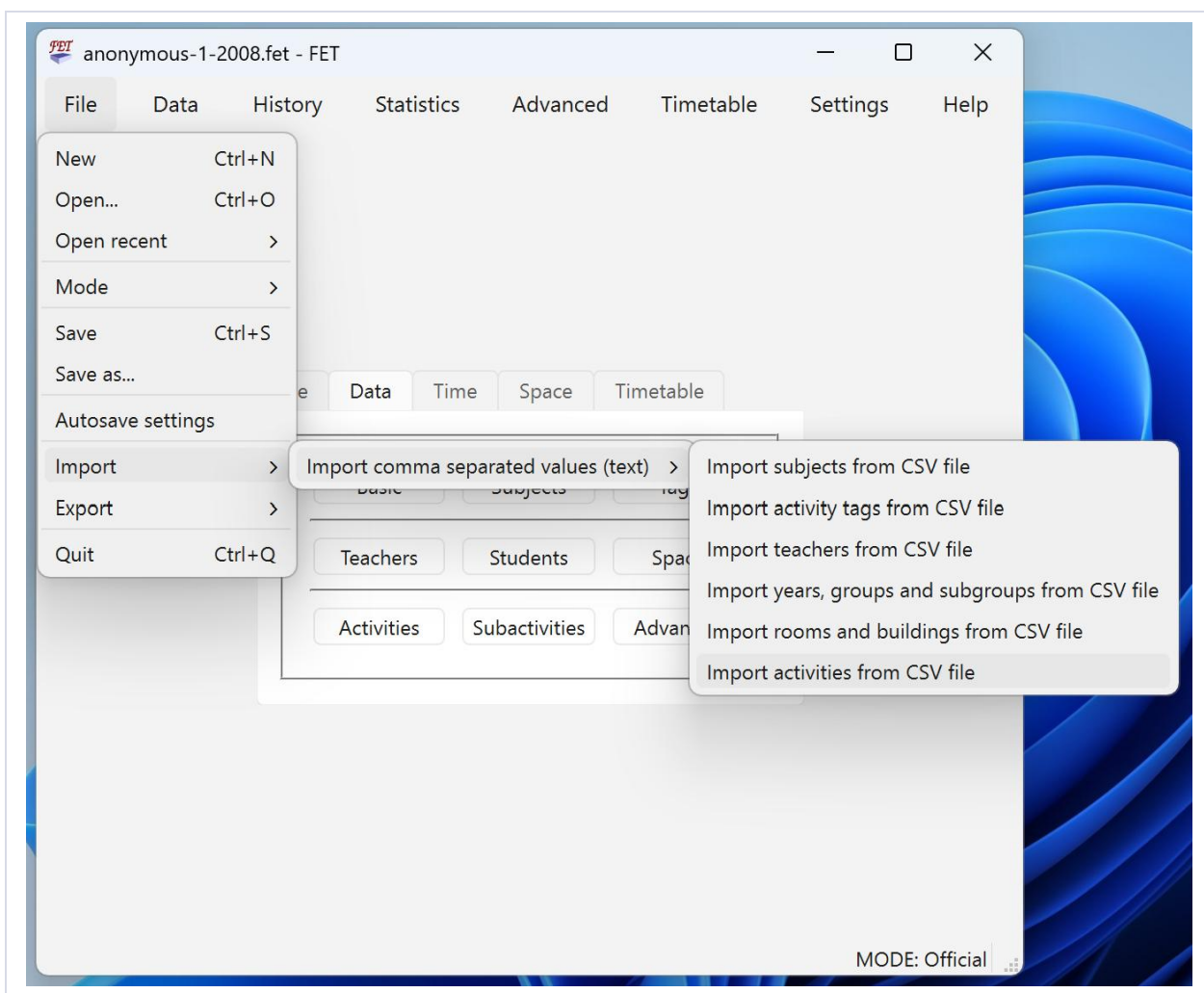


Figure 8. Importing activities from a CSV file.

10. Step 5 — Rooms

Rooms are the physical places lessons happen, for example: Classroom 101, Physics Lab, Computer Lab, Gym.

Add them under **Data → Space → Rooms**. Give each room a **capacity** (how many students it seats) so FET won't put a class of 30 into a room for 20. Rooms can also belong to a **Building**, which matters if you later want to limit how far teachers or classes travel. Rooms are optional, but recommended whenever room allocation matters.

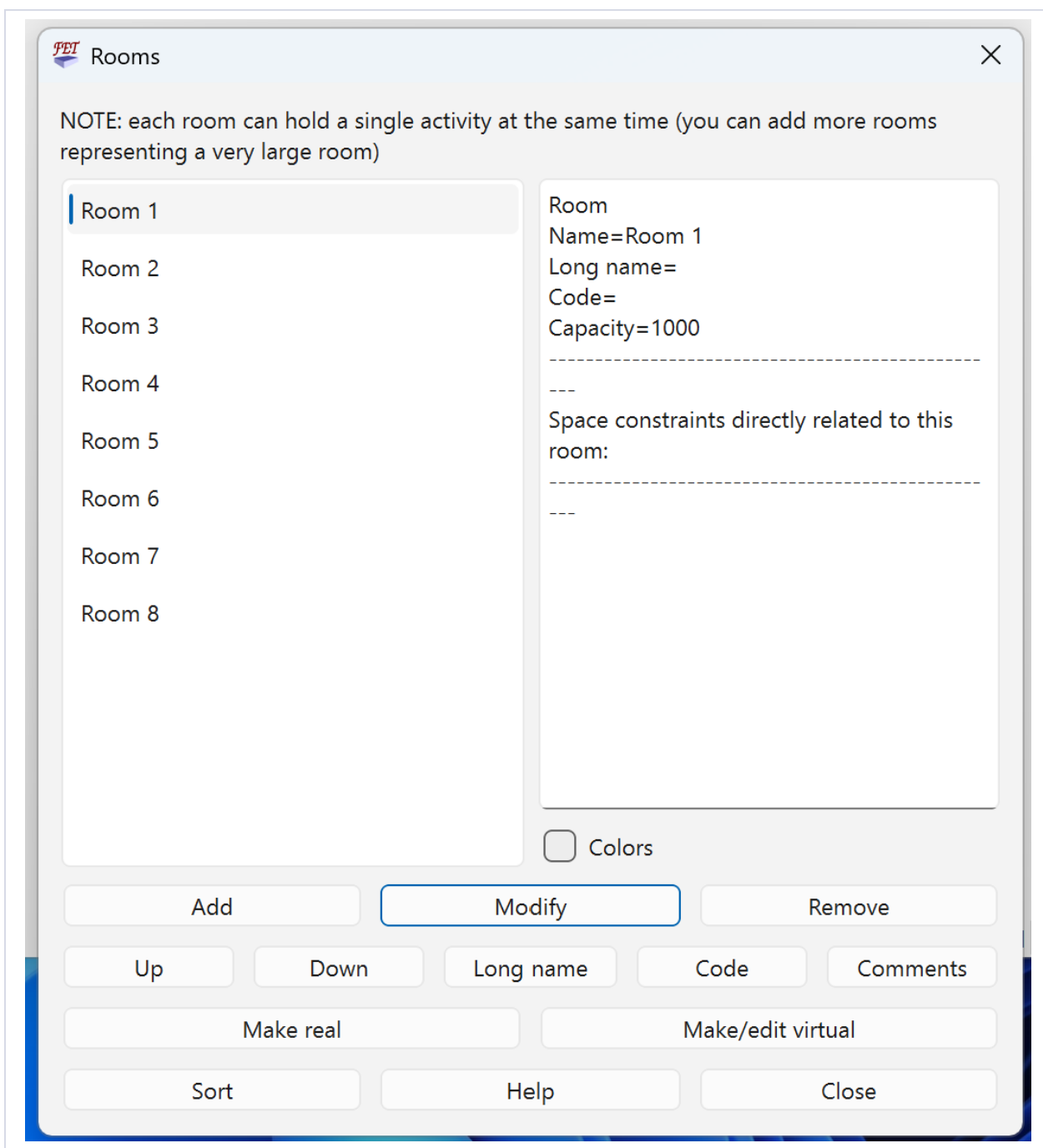


Figure 9. The Rooms list, with capacities.

11. Step 6 — Activity Preferred Room(s)

This constraint tells FET **which room or rooms an activity is allowed to use**.

- **Chemistry → Chemistry Lab only** (one room).
- **Mathematics → Room 101, Room 102, or Room 103** (several allowed rooms).

Allowing **several** rooms gives FET more freedom and makes a solution easier to find; pinning to **one** room is stricter. Add it under **Data → Space → Rooms → Activity preferred room(s)**.

Add activity preferred room

Filter

Activity: 9 - 1/3 - Teacher 26 - Subject 47 - Students group 1_1

Selected

Real rooms

- Room 1
- Room 2
- Room 3
- Room 4
- Room 5
- Room 6
- Room 7

Room 1

Clear

☒ Permanently locked

Select this if you want this constraint not to be unlockable from the 'Timetable' menu (recommended). You can always remove this constraint from the constraints dialogs or uncheck "permanently locked" checkbox by editing this constraint

Weight percentage (allowed: 0.0%-100.0%, necessary 100% if list nonempty) 100

Add constraint Help Close

Figure 10. The 'Activity preferred room(s)' constraint dialog.

12. Understanding constraints: hard vs soft

Before adding constraints, learn the one field that appears on **every** constraint dialog: the **weight percentage**.

Weight percentage — the most important field in FET

100% = a hard rule (mandatory). FET will never break it. If two hard rules contradict each other, no timetable is possible.

Below 100% = a soft rule (preference). FET tries to honour it but may break it if needed — the

higher the percentage, the harder it tries.

A good habit for beginners: keep genuine requirements (a teacher is truly unavailable on Friday) at **100%**, and make nice-to-haves (a teacher would prefer to finish earlier on Fridays) soft, e.g. 80–95%. Making everything 100% is the most common reason a timetable "can't be generated."

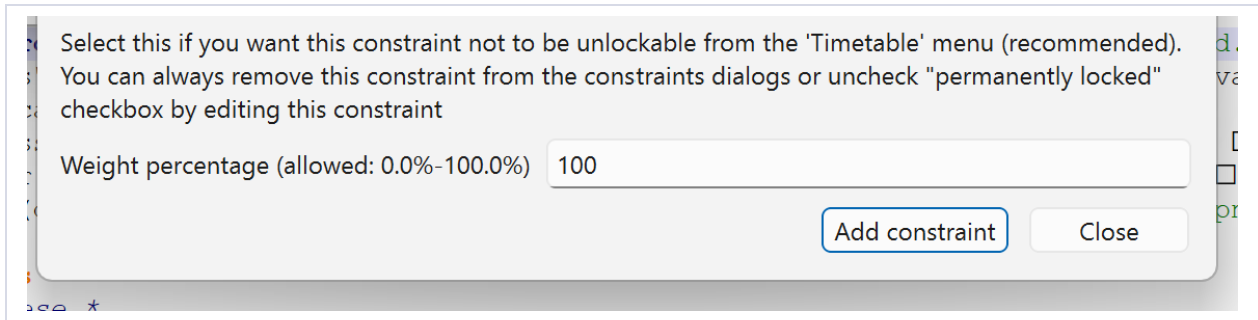


Figure 11. The Weight percentage field, shown on a constraint dialog.

13. Step 7 — Essential constraints

You don't need many rules to get a good timetable. Start with these and add more only if you need them:

- **Teacher not available times** — when a teacher can't teach (e.g. Friday afternoons).
- **Students not available times** — when a class isn't available.
- **Break times** — reserve lunch/assembly slots for the whole school.
- **Minimum days between activities** — spread a subject's repeated lessons across the week (e.g. the three Maths lessons on different days). Very commonly used.
- **Students max gaps per day** (ideally 0) — no idle holes in a class's day.
- **Teacher max gaps per day / per week** — keep teacher schedules compact.
- **Maximum hours daily** (teachers and students) — humane daily limits.
- **Activity preferred starting time** — pin a fixed lesson to a slot; tick locked to freeze it so generation won't move it.
- **Subject preferred room(s)** — send every Science lesson to the lab.
- **Students home room** — keep a class in one room while teachers move.

Tips from FET's own help (paraphrased)

Max gaps per day and **max gaps per week** only work at **100%** — FET's algorithm accepts no other weight for them (it's 100%, or don't add the rule). Add only **one** "max gaps per week" rule; start with a loose number of gaps and tighten it as solutions appear.

"Not available" times and breaks do not count as gaps — a teacher or class with lessons on either side of a free block is not penalised.

For **minimum hours daily**, start around **2** and raise it gradually; too high too soon can make the timetable impossible to solve.


Add teacher not available times
✕

Note: this constraint does not induce gaps for teachers. If a teacher has activities before and after a not available period, there will be no gaps counted

X (red)=not allowed, empty (green)=allowed

Teacher

Teacher 1
▼

▼	Monday	Tuesday	Wednesday	Thursday	Friday
1					
2					
3					
4					
5					
6					
7					

Weight percentage (necessary: 100%)

☐ Colors

Figure 12. Blocking slots with 'Teacher not available times'.

14. Step 8 — Generate the timetable

With data and constraints in place, choose **Timetable → Generate**. FET's solver places every active activity into a day, hour, and room, respecting all hard rules and minimising soft-rule penalties. You'll see **live progress**; a small problem solves in seconds.

When it finishes you have a complete, clash-free timetable. You can **lock** any placement you like (so a re-generate won't move it) and regenerate the rest — handy for fine-tuning by hand.

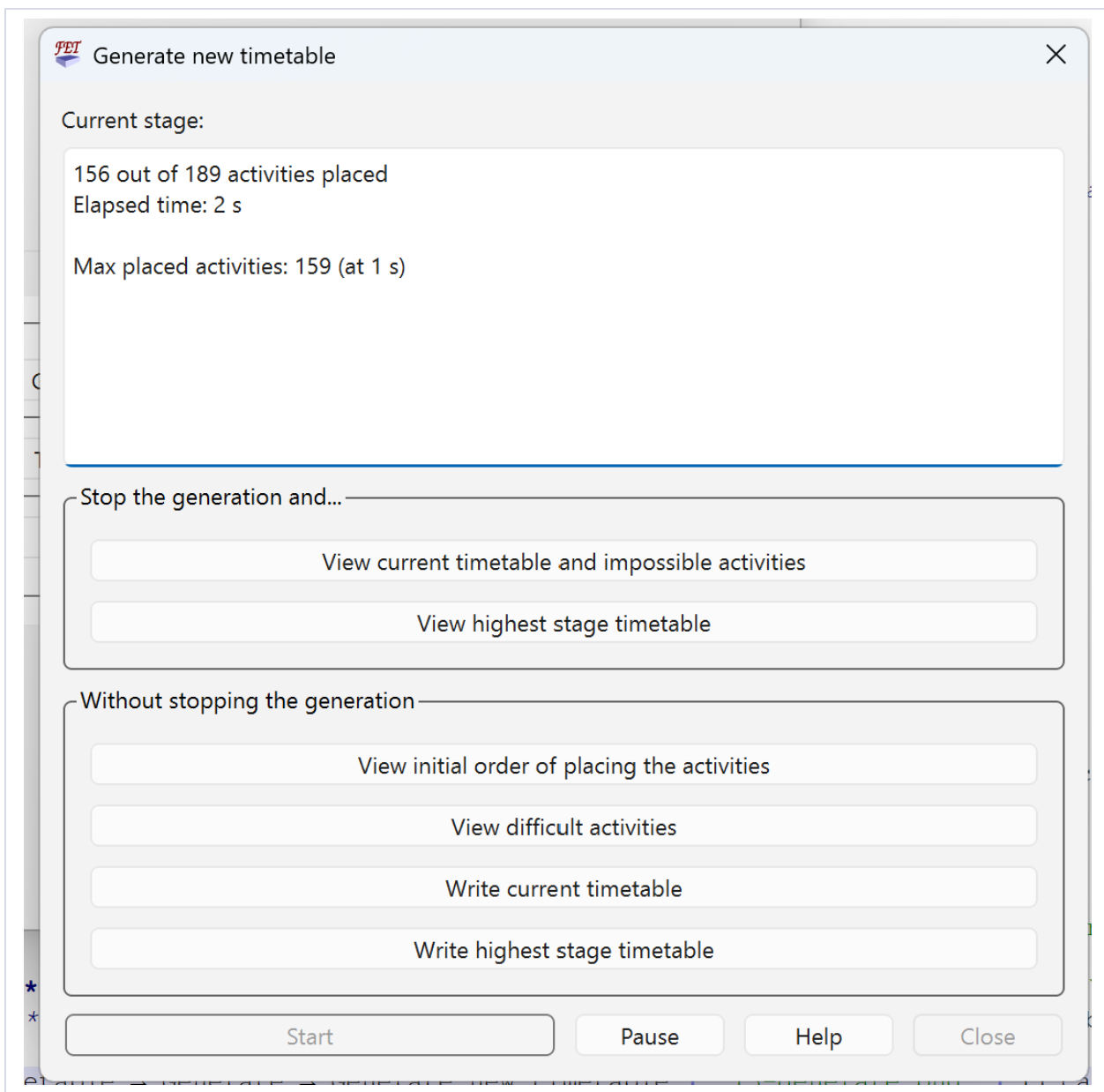


Figure 13. Generating the timetable — progress and success.

15. Step 9 — View and export your timetable

Open the **Timetable** menu to see your result from every angle:

- **Per teacher, per class (students), and per room.**
- In **days-horizontal**, **days-vertical**, or **time-horizontal** layouts.

Here's the nice part: **every time you generate, FET writes the timetable to your Output Directory automatically** — a full set of HTML pages (and CSV) for every teacher, class, and room. You can **print directly from FET**, and you can also open those HTML files in a web browser to print or share them. Your .fet file stays your master copy, to reopen and edit anytime.

Before you publish: take a quick look at **Statistics** and the **teachers' free-periods** view — you want a few teachers free each period so an absence can be covered.

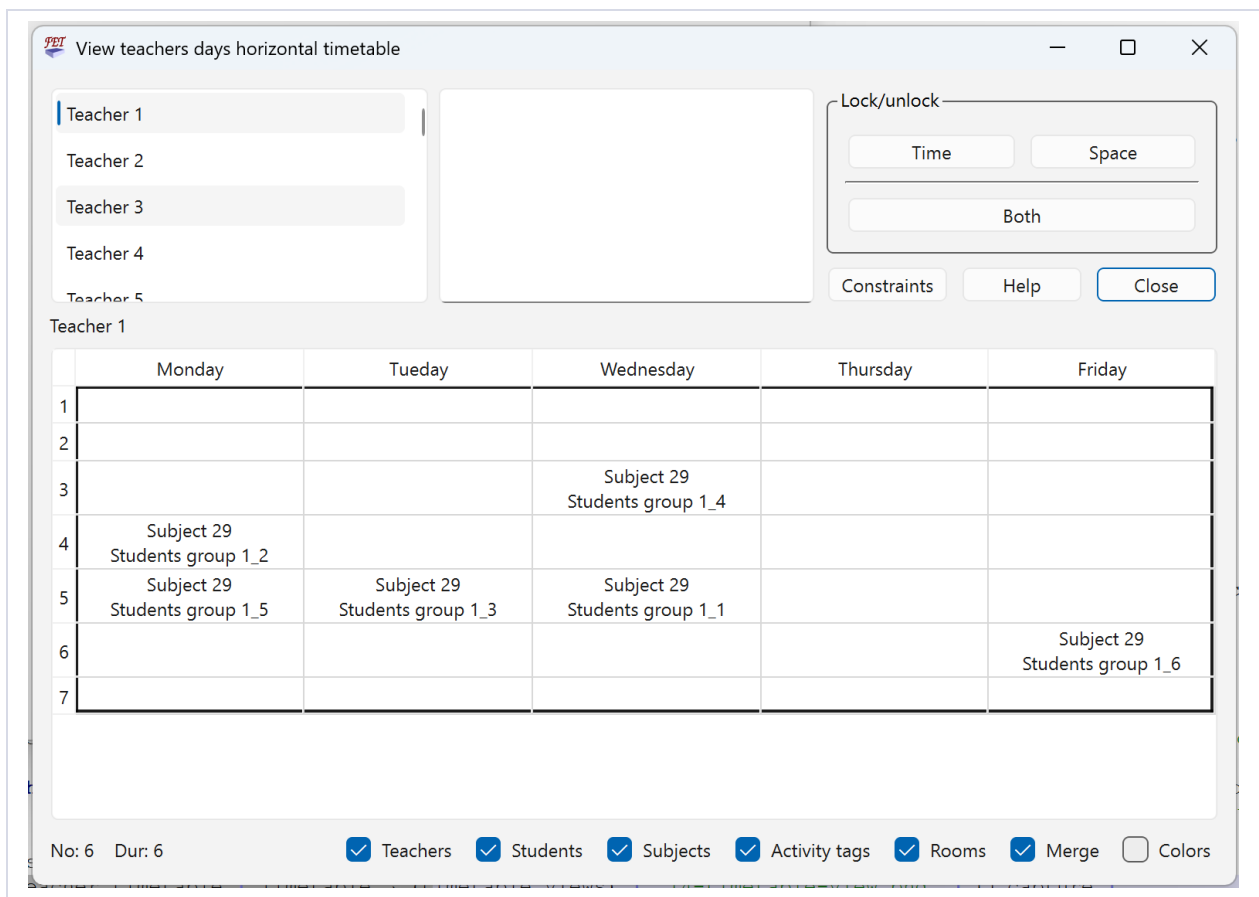


Figure 14. A generated timetable on screen — the payoff.

16. If it won't generate

If FET reports activities it couldn't place, don't panic — it's almost always one of these:

10. **Too many hard rules.** Lower a constraint's weight from 100%, or remove a contradictory one.
11. **Not enough room in the week.** A teacher (or class) needs more lesson-hours than there are slots — add hours/days or reduce activities.
12. **An activity has nowhere legal to go.** Check that its allowed rooms and allowed times aren't blocked by other rules.
13. **"Minimum days between" can't fit.** You can't spread 6 weekly sessions across 5 days — reduce the requirement or the count.

Relax one thing at a time and regenerate.

17. Common beginner mistakes

- Missing activities (nothing to schedule).
- More lessons than the available periods can hold.
- Teacher overload (too many hours for one person).
- Student overload.
- Excessive constraints, or too many set to 100%.
- Over-using single-room requirements (pin to several rooms when you can).

18. Data completeness checklist

- ✓ Days defined
- ✓ Hours defined
- ✓ Activities imported or created
- ✓ Teachers available
- ✓ Student groups available
- ✓ Subjects available
- ✓ Rooms created (if needed)
- ✓ Constraints reviewed (and weights sensible)

19. Beginner success formula

Days + Hours + Activities are the core. Import activities whenever possible, add a handful of rooms and constraints, then generate. Start simple, get a working timetable, and refine from there.

Next steps: when you're ready for CSV column details, the full constraint catalogue, virtual rooms, buildings, and the other file modes, see the FET Comprehensive User Guide.