# Introduction to Expyriment

**Programming Psychology Experiments (CORE-1)**

Barbu Revencu & Maxime Cauté

Session 2 | 17 September 2025

# The plan for today

1. Finish last week’s exercises (25′)
2. Your feedback (5′)
3. Introduce expyriment (15′)
4. Start coding with expyriment (45′)

# Last week’s exercises

# Tasks for you

```
Barbu@Mac % cd your-path/Programming/Assignments/Week-1/Exercises
```

**Those of you who solved Exercise 1.1 only**: Solve the next exercises

```
Barbu@Mac Exercises % python Exercise-1.1.py
```

**Those of you who only solved Exercise 1**: Solve Ex. 2–7 in VS Code

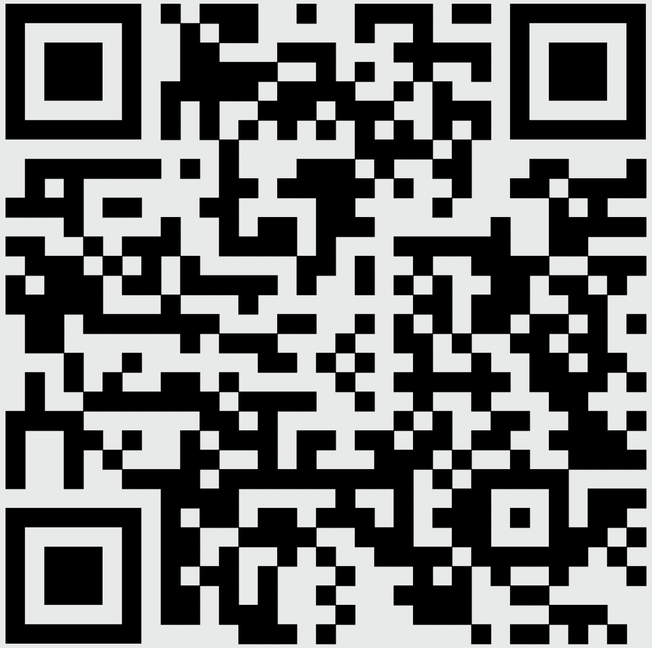
**Those of you who solved Exercises 1–7**: Raise your hand, we will come and look at your solutions

When done:

```
Barbu@Mac Exercises % cd ..\..
Barbu@Mac Assignments % git add .
Barbu@Mac Assignments % git commit -m "Week 1 Exercises"
Barbu@Mac Assignments % git push origin
```

# Difficulty of Week 1's assignments

Fill in the form at https://forms.gle/TPDjfrC3Ejww1q26A



# Expyriment

# What is expyriment?

A Python library for designing and running psychology, neuroscience, and psychophysics experiments

It’s meant for researchers who need to **present stimuli** (text, images, sounds) and collect responses (e.g., key presses) **with good timing precision**

# Pros of expyriment

A **clean and simple** psychology experiment generator, which promotes good programming practices (readability)

It relies on Python, so it aims to be **reproducible** across platforms (we'll see about that!)

It allows researchers to **focus on the high-level, abstract structure** of experiments without having to code low-level timing or graphics routines themselves

# Cons of expyriment

It relies on Python, so it's **not possible to run remote online experiments** (for this, you will learn jsPsych later on in the course)

It has a **small user community**, which means that there are not many demonstrations/examples on the web (the interface, however, is very well documented)

> *Note*: This also means that **LLMs will often hallucinate** when prompted about expyriment since the training data is sparse

# What does the following code do?

```
from expyriment import stimuli

fixation = stimuli.FixCross()
circle = stimuli.Circle(radius=50)

fixation.present()
clock.wait(1000)
circle.present()

keyboard.wait()
```

Let’s dig into it: https://github.com/barburevencu/PPE/blob/main/Week-2/Instructions.md