

Practical Programming for Cognitive Scientists syllabus, Fall 2023

Class: Thursdays, 13:30-15:10, C503. **Bring a laptop to class every week.**

Credits: 2 credits

Eligibility: Graduate students in Cognitive Science for credit, open to all for auditing.

Instructor: Jonathan F. Kominsky (KominskyJ@ceu.edu)

Office hours: By appointment

Description: The goal of this class is to help you implement experiments using state-of-the-art programming tools like PsychoPy, MatLab and the Psychophysics Toolbox, and JSPsych. Rather than just teaching you how to program experiments in each of these systems, the aim is to give you a sufficient foundation in the principles and concepts of programming, and practice using those skills, to the point where you know how to learn anything the course does not teach you.

Learning outcomes: By the end of the course, you should be able to

- Program an experiment in PsychoPy, MatLab, and/or JSPsych.
- Understand how programming concepts are implemented in different programming languages, and how to find out how something is implemented in an unfamiliar language.
- Interpret and resolve error messages and unexpected behavior.

Assignments: Most weeks will have a homework assignment that we will start in class. Homework is **due at midnight the night before the following class**. Google and external resources are allowed and encouraged. Collaborative work is permitted, but each person is expected to turn in their own assignments. LLMs (e.g. ChatGPT) will just give you broken code but you're welcome to try as long as you say you used it. All assignments should be uploaded through the Moodle interface at <https://ceulearning.ceu.edu> . Grading is simple: If it works, you pass. We'll also go over the homework in class, and, you can also turn in a corrected file by midnight that day for credit (but you should still turn in an initial version by midnight).

Final project: The final project in the class will involve writing a detailed experimental procedure for an experiment that you want to build using programming tools. You can use an actual experiment you plan to run or try to replicate an existing design. You will turn in the written procedure and the code. Evaluation will be based on the detail of the written procedure and whether the code successfully implements it.

Before Week 1: please download and install PsychoPy
(<https://www.psychopy.org/download.html>)

Zero Week (14 Sept): Introduction, getting set up

- Class design, goals, and expectations
- Basic programming concepts, assessment survey

Week 1 (21 September): Python 101

- Installing PsychoPy and accessing the Coder interface
- Basic python concepts & syntax review
- **Homework:** <https://psych750.github.io/index.html> “self-study guides” up through “Review of fundamentals A” (you don’t need to turn anything in)

Week 2 (28 September): Python data structures, objects, and libraries.

- Python objects, functions, and properties.
- Using existing libraries, especially PsychoPy’s libraries.
- Randomization and counterbalancing.
- Thinking in data structures.
- **Homework:** Code exercise 1 – orders and randomization

Week 3 (5 October): Making stimuli in PsychoPy

- PsychoPy windows
- Drawing different kinds of visual stimuli
- Animation and frames
- **Homework:** Code exercise 2 – making some simple stimuli

Week 4 (12 October): Recording responses and storing data

- PsychoPy’s event and keyboard libraries
- Keyboard and mouse recording options
- GUI objects
- Accessing external files and saving data safely
- **Homework:** Code exercise 3 – Stroop experiment script

Week 5 (19 October): Advanced PsychoPy

- Recording audio, screens, and webcams.
- More complicated experiment designs
- **Homework:** Code exercise 4 – advanced PsychoPy experiment design. **Due in 2 weeks**

Week 6 (26 October): **National day, no class**

Week 7 (2 November): Translating between programming languages

- Identifying basic concepts in a new language
- Looking up a language reference
- Overview of MatLab and JS differences from Python
- **Homework:** Code translation exercise

Week 8 (9 November): Online studies with jsPsych and Pavlovia Part I

- The relationship between JS, CSS, HTML, and what Pavlovia is doing
- What happens on the client computer vs. the server
- Using Git to interface with Pavlovia (and in general)
- Basic JS syntax
- **Homework:** Final project proposal

Week 9 (16 November): Online studies with jsPsych and Pavlovia Part II

- The jsPsych Timeline
- jsPsych stimulus elements and response types
- **Homework:** Code exercise 5 – Simple experiment in jsPsych

Week 10 (23 November): Online studies with jsPsych and Pavlovia Part III

- Advanced JSPsych designs
- Using webgazer.js
- JS code elements and PsychoJS
- **Homework:** Code exercise 6 – two jsPsych experiments

Week 11 (30 November): MatLab and the Psychophysics Toolbox

- Installation, compatibility issues
- Advantages and disadvantages
- Making stimuli in MatLab
- Collecting responses
- **Homework:** Code exercise 7 – Simple PTB experiment

Week 12 (7 December): MatLab and PTB's advanced features

- Comparing performance of a PsychoPy experiment and a PTB experiment
- MatLab interfaces with EEG and fMRI systems
- Time to work on final projects
- **Homework:** None, work on your final projects.