

Practical Programming for Cognitive Scientists syllabus, Winter 2023

Class: Tuesdays 10:50-12:30, QS B505. **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. 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: Complete the assessment survey!

(https://qualtrics.ceu.edu/jfe/form/SV_2602Rt9nV1inu74) Please complete the assessment survey by midnight before the first class. Also, **please download and install PsychoPy** (<https://www.psychopy.org/download.html>)

Week 1 (10 January): Introduction, getting set up

- Class design, goals, and expectations
- Basic programming concepts, gaps from assessment survey
- 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 (17 January): 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 (24 January): 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 (31 January): 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 (7 February): Advanced PsychoPy stimulus design

- Movies and audio, including microphones.
- More complicated experiment designs
- Screen recording
- **Homework:** Code exercise 4 – advanced PsychoPy experiment design

Week 6 (14 February): PsychoPy, iohub, and advanced python

- Communicating with eye-trackers and other systems
- Make-up for any delays from weeks 1-5
- PsychoPy's Builder and code elements – interacting with machine-generated code
- [If there's time] Regular Expressions and text manipulation (RegEx)
- **Homework:** Final project proposal

Week 7 (21 February): 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 (28 February): 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
- The JSPsych Timeline
- **Homework:** Code exercise 5 – fun with JavaScript

Week 9 (7 March): Online studies with JSPsych and Pavlovia Part II

- Further topics and practice with JSPsych.
- Incorporating webgazer.js
- **Homework:** Code exercise 6 - building two simple experiments in JSPsych

Week 10 (14 March): MatLab and the Psychophysics Toolbox

- Installation, compatibility issues
- Advantages and disadvantages
- Making stimuli in MatLab
- **Homework:** TBD

Week 11 (21 March): MatLab and PTB's advanced features

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

Week 12 (28 March): Collaborative workshop and troubleshooting for final projects

- Bring technical issues with final (or other) projects and we will go through them together
- General troubleshooting
- Pilot each other in your projects.
- Catch-up for anything from previous weeks