# First year psychology lab classes: lecturer’s notes

## License

These lab classes were developed at the University of Nottingham as part of the first year course in BSc (Hons) Psychology, with support from the Higher Education Association Psychology Network (HEA-PN). The materials provided here may be distributed freely under the [Creative Commons Attribution-NonCommercial 3.0 Unported License](http://creativecommons.org/licenses/by-nc/3.0), but please acknowledge the University of Nottingham and the HEA if you use them.

## Project history

The HEA provided a partial buyout of teaching for Jonathan Peirce at University of Nottingham to write these lectures 2010-11. We subsequently decided that it would be worth having one year where the undergraduates learnt about both E-Prime and PsychoPy, for evaluation purposes. As a result, for 2010-11 only the practicals 1-6 have so far been used in the classroom with the other classes that year using E-Prime instead. The later lectures, on Memory and Inhibition of Return are therefore less well developed and should be carefully road-tested by you!

## Using these lectures yourself

For each class the necessary lectures are provided (roughly as given at Nottingham).

The lectures are deliberately rather verbose and you probably want to cut this down considerably to avoid students falling asleep or failing to show up. They were written this way because it’s hard to write a lecture for *somebody else* to give, without being very explicit about what verbal material was intended alongside the slides.

In addition, each class has lecturers notes file with it, explaining more about the ideas to be conveyed and a batch analysis script where that’s feasible. To run the batch analysis simply open it in PsychoPy (Coder view not Builder) and press run. It will ask you to select the data files you want to run the analysis on (the lecturer materials contain some example data files to try that out). For some experiments, where students create their own experiment, batch analysis is too difficult because each student uses a slightly different name for the variables etc. In these cases we get students to write down their values on a piece of paper and a demonstrators types them into SPSS. Old-school!

## Contributing to this project

If you would like to provide feedback or contribute improved versions of these lectures then please send them to lpzjwp (at) exmail.nottingham.ac.uk.

## Module design

At Nottingham, students attend one lab class every two weeks and each lasts up to 3 hours. Thus this module contains 10 weeks of materials that are spread across a year of teaching. This module runs alongside a module in statistics and is designed so that the materials being taught in the two classes match up as closely as possible (the notable exception is that in the first few weeks no inferential tests have been taught but t-tests are being used in labs).

The classes at Nottingham run in computer labs with roughly 60 computers. Students typically run with one student per computer, although one computer between two is also perfectly feasible (this has been done when fewer computers have been available).

Most of the classes involve a period of lecture, about the study, the software or report writing, followed by data collection, followed by analysis.

## Assessments

At Nottingham, three practical write-ups are assessed per semester.

In Semester 1 the classes 3, 4 and 5 require assessed reports from the students. In the first two classes the students have not been taught about how to write a report. In class 3 they get taught this, but still have insufficient stats to understand t-tests. Therefore, the first of the assessed practicals really considers only the ability to get the information in the right structure.

In Semester 2 two of the practicals are double-classes, so assessments are made of the 3 distinct experiments (ie weeks 6, 8, 10 of the overall module).

## Learning objectives for each class

### Stroop: An easy introduction, with an effect that always works.

1. How to use the lab computers (info about where to fetch and store data)
2. Why are we running experiments at all?
3. PsychoPy views and data files
4. Basic data handling in Excel (cells, formulae, relative v absolute references)
5. Finding relevant reading materials

### Sternberg: Same themes but developing in complexity

1. PsychoPy routines, components and flow
2. More complex data analysis in Excel (sorting, filtering incorrect answers, smart-fill)

### Navon: how to write a report

1. How to write a report
2. Global versus local processing
3. Getting summary data (gender age of participants) from SPSS

### Pseudo-homophones

1. A more detailed study developing PsychoPy skills
2. Using more precise stimulus timing and device-independent layout

### BART and digit ratio (actually about demand charactersistics)

1. Social psych can use experiments too!
2. The importance of double-blind studies
3. PsychoPy can run complex dynamic studies (but needs some inline coding)

### Mental Rotation: generating a PsychoPy study from scratch

1. Broad (too long?) intro to mental imagery
2. Create a replication of Cooper and Shepard from scratch in the Builder
3. Students should be able to conduct most of the analyses themselves by this stage

7. & 8. Memory

1. Teaches more independent study
2. Flexible prac with more decisions for the student
3. Choice of 3 study ideas, 2 of which can be done with pen and paper designs (1 uses PsychoPy)

9. & 10. Inhibition of Return (IOR)

1. Attention
2. One-way ANOVA