PsychoPy for FYP students

John Allen and Frank Gasking

While you’re waiting…

At the end of this session you will be familiar with:

1. Create directory `c:\work`
2. Create directory `c:\work\{your_username}`
3. Download todays materials from Moodle or from
   1. https://www.kent.ac.uk/psychology/downloads/PsychoPy_FYP.zip
4. Save the `‘.zip’` file into your folder (see #2 above)
5. Right click on the downloaded file and select ‘`Extract all`’

Learning outcomes from training

At the end of this session you will be familiar with:

- What is PsychoPy used for.
- For a pre-existing experiment
  - Getting the gist of what it does.
  - Making basic edits to it.
- Keys parts of the PsychoPy programming environment.
- Using Excel files to provide your trial data.
- Running an experiment.
- Interpreting the contents of results data output files.

PsychoPy:
Why might I want to use it?

- In Psychology we often want to:
  - display stimuli PRECISELY
  - capture user responses
    - key presses, mouse clicks
    - eye movements, brain responses (EEG), heart rate,…
  - (often) with very accurate response times (RTs)

PsychoPy is free and is written in a computing language called **Python, which means:**

- works on Windows/MAC/LINUX
- has lots of support

We focus on **PsychoPy** – alternatives exist,
e.g. E-Prime, Matlab (PsychToolbox), Superlab, Inquisit

Using and installing PsychoPy

For all of 2017/18 we are using v1.85.03 of PsychoPy (v1.85.04 for Mac)

- Fixed for consistency across School
- Experiments created in an older version should upgrade fine.
  - Let us know if you have any problems with any upgrade process.
- Warning! - Experiments created in an newer version will not run properly on an older version!
- Any upgrade to the version will only occur if a major bug is found with PsychoPy, and will be communicated.

Installation instructions

Student/staff PC's (like you are on now), should already have it installed on the PC you are on. We show how to load it a few slides on…

For home use, visit:

http://psychopy.org/

1. Select the DOWNLOAD link - this takes you to the SOURCEFORGE website which hosts the download files
2. Select the PsychoPy folder and then choose the appropriate installer to download and then run
3. If in doubt, choose the latest 'STANDALONE' version, MAC or Windows as appropriate
4. see also www.psychopy.org/installation

Know your file types

Experimental files:

- `*.psyexp` : these are PsychoPy `BUILDER` source files
- `*.py` : these are Python source code files. Your PsychoPy `CODER` files

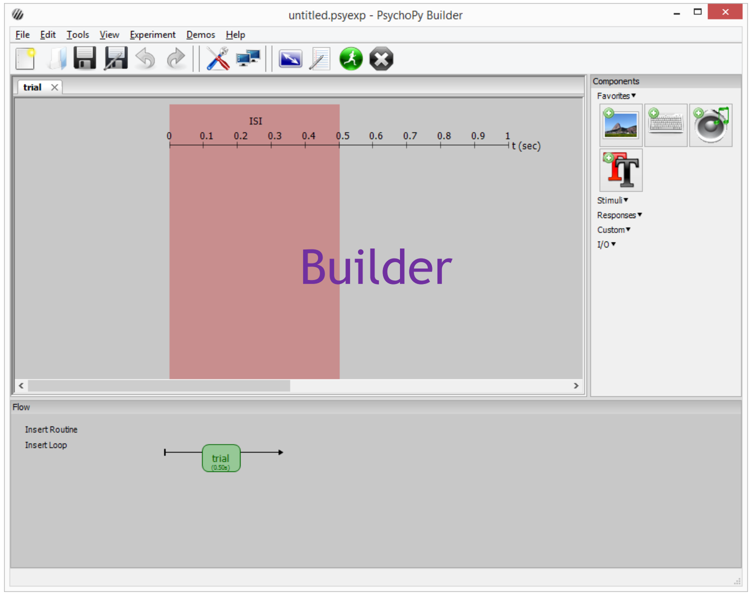
(see FAQ for how to see full filenames)

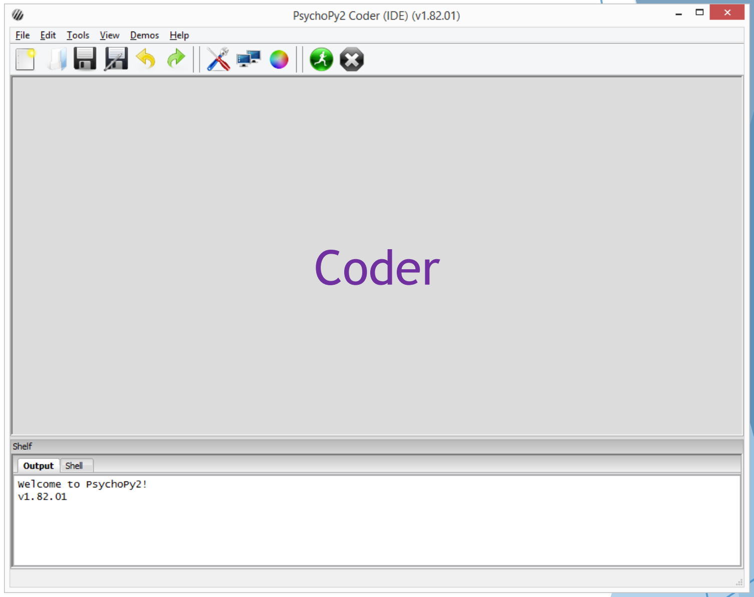
Starting PsychoPy

- PsychoPy icon …
- On UoK 'managed build' Windows PCs
  - Student PC's (N1.04)
    - `START`
    - Enter `PsychoPy` in the search field
  - Lab / Staff PC's
    - `START | PsychoPy`
    - `If not present, please contact psychsupport@kent.ac.uk`

Builder Vs Coder views

- Mostly we only use `Builder`


untitled.psyexp - PsychoPy Builder
File Edit Tools View Experiment Demos Help
trial
ISI
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 t (sec)
Builder
Components
Favorites
Stimuli
Responses
Custom
I/O
Flow
Insert Routine
Insert Loop
trial
(0.50s)

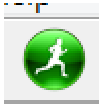

PsychoPy2 Coder (IDE) (v1.82.01)
File Edit Tools View Demos Help
Coder
Shelf
Output
Shell

```
Welcome to PsychoPy2!
v1.82.01
```

- If you're in *`CODER`* view, then in the menu click *`View/Goto Builder view`*

Exercise 1
Run a stroop demo

Exercise 1.1

- Open up `stroop/stroop.psyexp` from within your `C:/work` directory
- Don't worry what everything is just yet, we'll cover it later!
- Click the green icon  at the top to run it.
- Follow the instructions and see what it does.

Exercise 1.2

- Look inside the `/data` folder and see what is produced
- Open one of the most recent CSV files that you find
  - Study the file, noting the column headings.

Demo:
What comes out from PsychoPy?

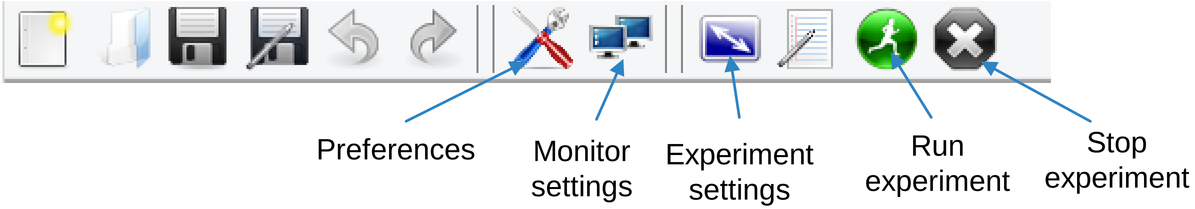
Data output

- Default area for data output is within a `data` sub-folder
- 4 files are generated:
  - `csv` – main data
  - `xlsx` – slightly simplified version of the csv file
  - `psydat` – complex - but useful for batch processing of results files. Possibly of interest if you are familiar with matplotlib
  - `log` – chronological record of everything BUT depends on what settings you use.
- These are all generated automatically.
- Filename based on Session/Participant and date number

- It is possible to **add extra data** to results files - ***Ask for help***

An overview of the interface

Menus (key items)



Preferences

Monitor settings

Experiment settings

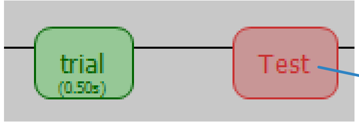
Run experiment

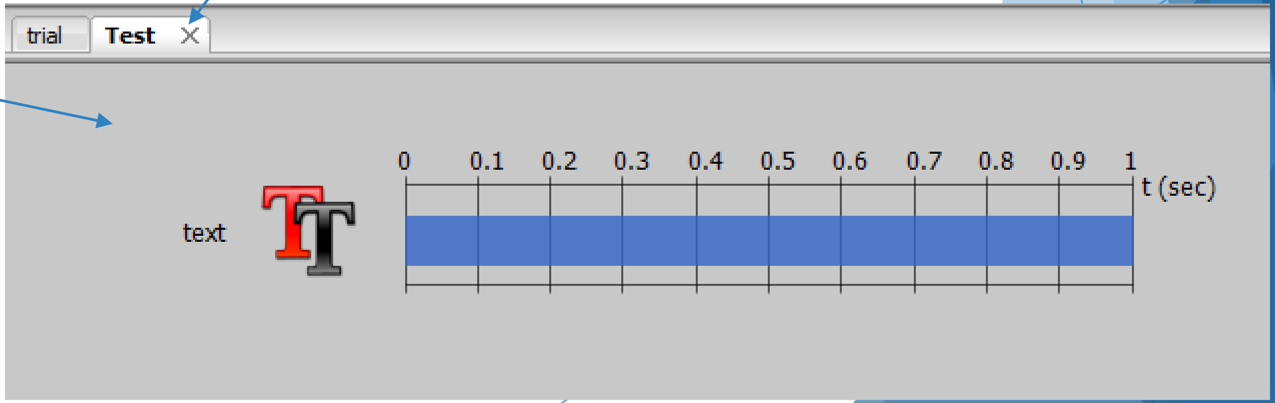
Stop experiment

Routines

One tab per routine

- components in this routine
- WARNING: closing a TAB deletes the routine!

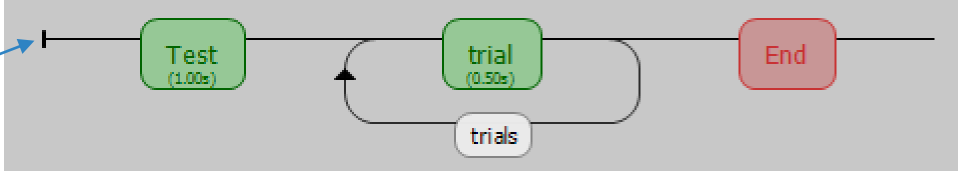




Continued…

Experiment flow

- sequential order of execution



Experiment starts here

And ends here

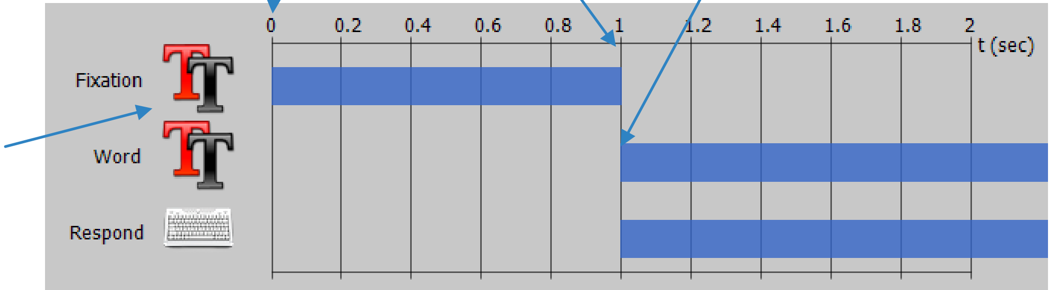
Routine timelines

- Concurrent objects/components controlled by start/duration

Timeline start

Fixation finishes

Components set to start after 1 second



Components on the timeline

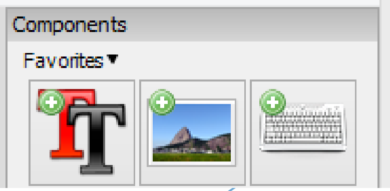
Drawing order is top first, when items drawn at same time!

Both these components end times are “open-ended”

Finishes, when user responds!

Components

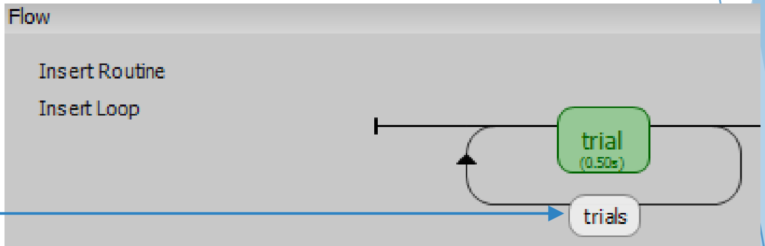
- 5 sections - drop-down menus
- More details coming up…



Key components

A quick look at a few of the key and basic components you can use:

- **Text box** (display component)
- **Keyboard** (response component)
  - Can record data automatically in results file
- **Mouse** (response component)
  - Can record data automatically in results file
- **Loops** (flow component)
  - Data feeds and repetitions



Loop created called "Trials" which repeats the routine called "trial"

When you add a component, you must give it a sensible name (No spaces, something meaningful so you remember what it is!)

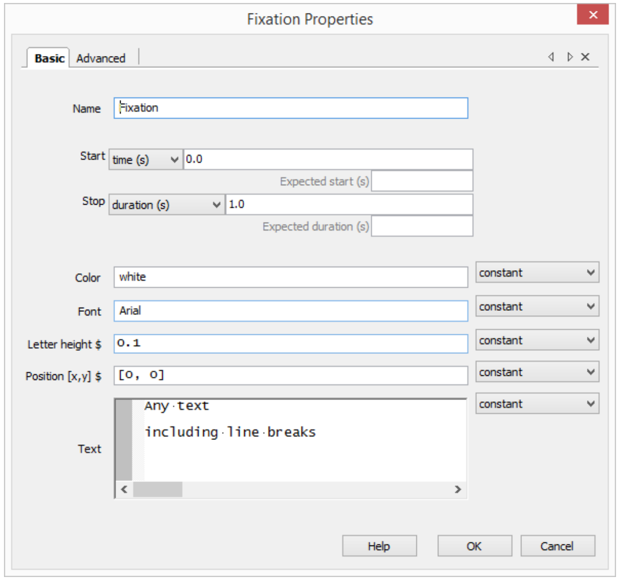
Each component comes with its own set of properties and attributes These allow us to make them do different things.

Key component properties

When adding a new component or double clicking a pre-existing one to edit on your timeline, you're presented with a properties box.

Including properties such as:

- Position
- Color
- Text (where applicable)
- Start time
- Duration
- ….



Fixation Properties

Name: Fixation

Start: time (s) 0.0

Expected start (s)

Stop: duration (s) 1.0

Expected duration (s)

Color: white (constant)

Font: Arial (constant)

Letter height $: 0.1 (constant)

Position [x,y] $: [0, 0] (constant)

Text: Any text
including line breaks
(constant)

…. And so on. Depending on the type of component.

These affect the **behaviour** of the component within our experiment!

Exercise 1.3

Exercise 1.3

- Study the Stroop experiment to observe some of what we've just discussed and
  - try changing the colour of the instructions
  - Increase the font size of the target word stimuli

More on “loops”

Loops are where we want to **repeat** something a number of times.
You may hear the word “Iteration” used to describe a repetition.

In PsychoPy we want to present stimuli or trials

- Often routines are repeated using a Loop
- This usually to represent our **trials**
- Text, Image or other components are repeated within our routine/s
  - Including their timelines and settings.
- But the actual text/image may need to change each time to represent trial stimuli.

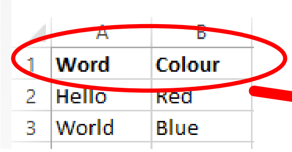
But how?...

By using a data source!

Created in Excel or package that can create CSV files

Headings refer to “attributes” that are created for us that we can use as our trial data

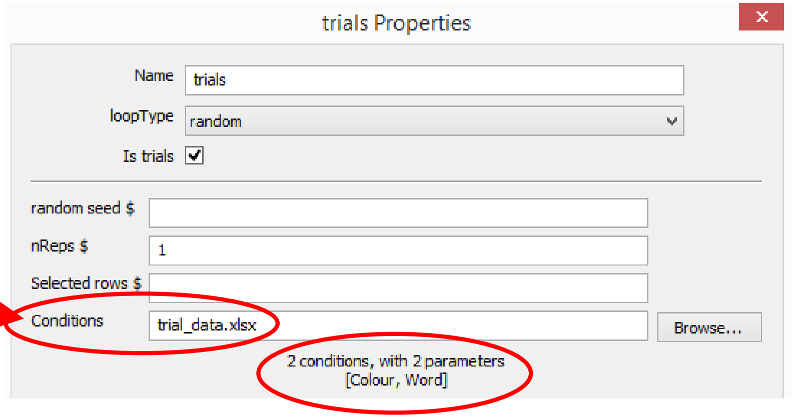
Excel/csv data file



	A	B
1	Word	Colour
2	Hello	Red
3	World	Blue

Attached to loop

Via “**Conditions**” property



trials Properties

Name: trials

loopType: random

Is trials ☑

random seed $:

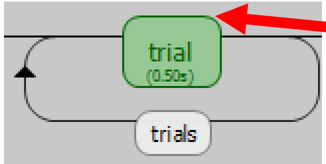
nReps $: 1

Selected rows $:

Conditions: trial_data.xlsx

2 conditions, with 2 parameters
[Colour, Word]

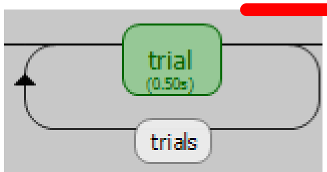
PsychoPy automatically picks up the headers and treats them as “**Attributes**”

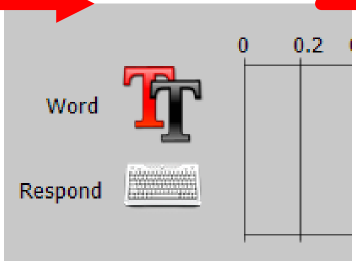


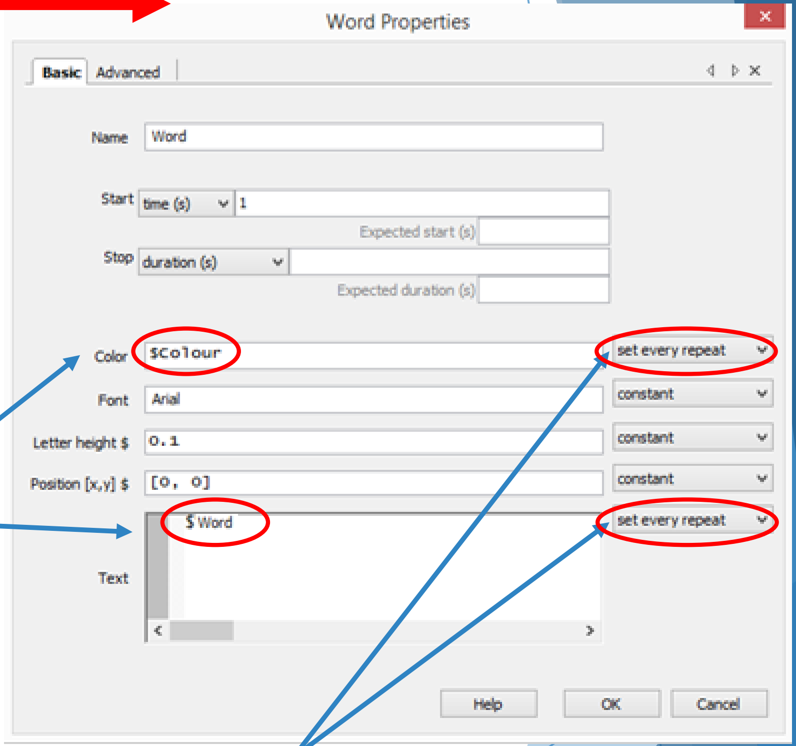
These are now available to the “**trial**” routine and the **components** within.

Accessing those attributes

“**Word**” and “**Colour**” now become available to use in the Text component







Name: Word

Start: time (s) 1

Expected start (s)

Stop: duration (s)

Expected duration (s)

Color: $Colour — set every repeat

Font: Arial — constant

Letter height $: 0.1 — constant

Position [x,y] $: [0, 0] — constant

Text: $ Word — set every repeat

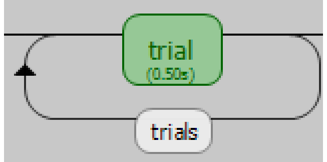
We use the dollar sign (**$**) to tell PsychoPy that we are using an attribute, which will contain data for our current trial

Crucially, we MUST change the update method from “**Constant**” to “**Set every repeat**”.

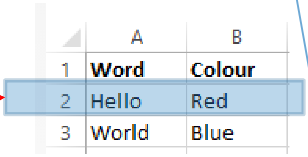
So we are saying – “**This value will change on every repetition/loop/trial”**

So when the trials run…

1) We come into the trial routine…



2) And fetch a row



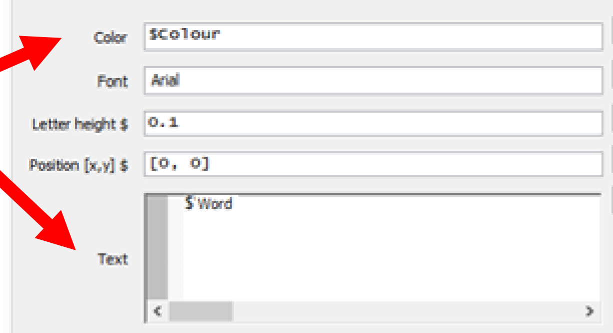
	A	B
1	Word	Colour
2	Hello	Red
3	World	Blue

3) **“Hello”** and **“Red”** are passed into their attributes.

Done automatically by PsychoPy

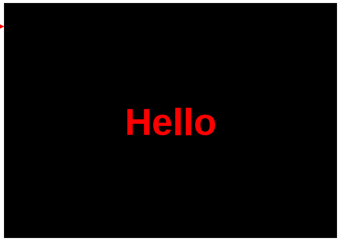
```
Color = “Red”
Word = “Hello”
```

4) These are passed to the Text component in the trial routine.

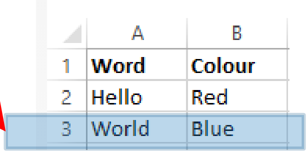


Color	$Colour
Font	Arial
Letter height $	0.1
Position [x,y] $	[0, 0]
Text	$ Word

5) This enables display when we run the experiment!



6) When the trial ends, we then fetch our next trial data…



	A	B
1	Word	Colour
2	Hello	Red
3	World	Blue

7) And the cycle repeats until we have no data left to use!

Exercise 2: further modifications of the Stroop demo

1. Open up `/stroop/stroop.psyexp` from Exercise 1
2. Change the intro text and the first line “**OK. Ready for the real thing?**” to “**Instructions**”
3. Update keyboard input from “**left**”, “**down**”, “**right**” to “**a**”, “**s**”, “**d**”
   - Ensure you update the instructions
4. Add another 5 trials to the Stroop
   - Edit the `trialTypes.xlsx` is where the source trial data is!
5. Try adding a new routine which will act as a title page for the experiment
   - HINT: the “instruct” routine should give you some clues
   - Look at the properties of the Text and Keyboard component and replicate
6. Re-Run and see what you get
   - Look at the result file, has your new data column come over?

For info only

Miscellaneous extras

Some “Gotchas”

1) Update `properties` setting

It is key to set the properties where you have an attribute coming in from “`Constant`” (never changes) to “`Set every repeat`” = (update and change on every repetition/trial/loop). Keeping as Constant will cause an error!

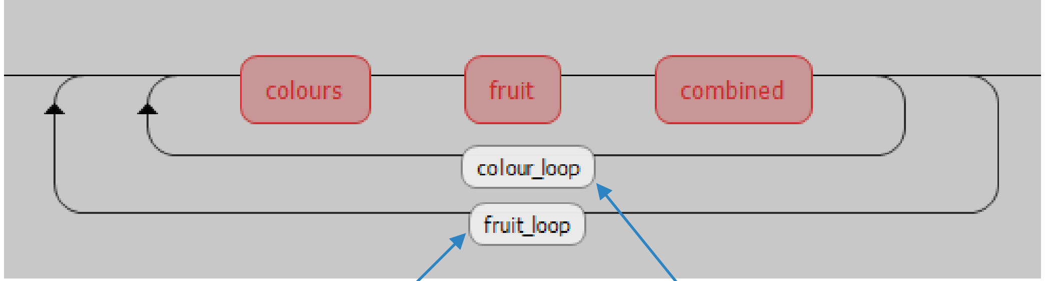
NOTE: The **“Set every frame” option updates** component at every single screen refresh **= Overkill!**

2) Making updates to the data file (i.e new columns)

When a trial list is updated with new columns, you **must re-attach the CSV file.** This is so PsychoPy will pick up the new attributes you add!

Nested loops and lists

- Useful for randomising order of blocks (outer level)



Outer loop
(Starts first)

Inner loop
(Starts second, runs and
Completes each time)

Components - Part 2

Overview of remaining components in brief.

Stimuli:

- **Sound** – Playback of sound
- **Image** – Display of image
- **Aperture** – Add a circular effect onto image component
- **Grating** – Wrapped texture that can be cycled in 2 dimensions
- **Movie** – Playback of movie files
- **Dots** – Presentation of Random Dot Kinematogram to participants
- **Polygon** – Shape presentation of different sides (square, rectangle, octagon)

Responses:

- **Mouse** – Take responses from the mouse
- **Mic** – Only records sound, doesn't register response to sound
- **Scale** – Mouse friendly scale to choose a value
- **ioBox, Cedrus** – Input options for external hardware devices and button boxes

Other:

- **Parallel icon** – Send signals down a cable (EEG)
- **Static** – A static period to allow for pre-loading images or other operations

More details can be found at
http://www.psychopy.org/builder/components.html

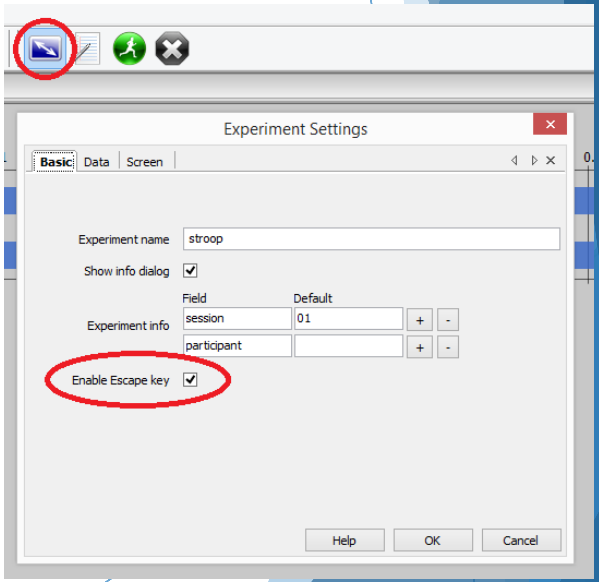
File Preferences (menu option)& Experiment Settings (icon)

Escaping from your experiment

- One of the key settings you may want to consider changing!
- By default, when running an experiment you can press ESC
- This will halt the experiment
- And save any data recorded so far.

This can be disabled:

- E.g.



Experiment Settings

Basic | Data | Screen

Experiment name: stroop

Show info dialog ☑

Experiment info

Field	Default
session	01
participant	

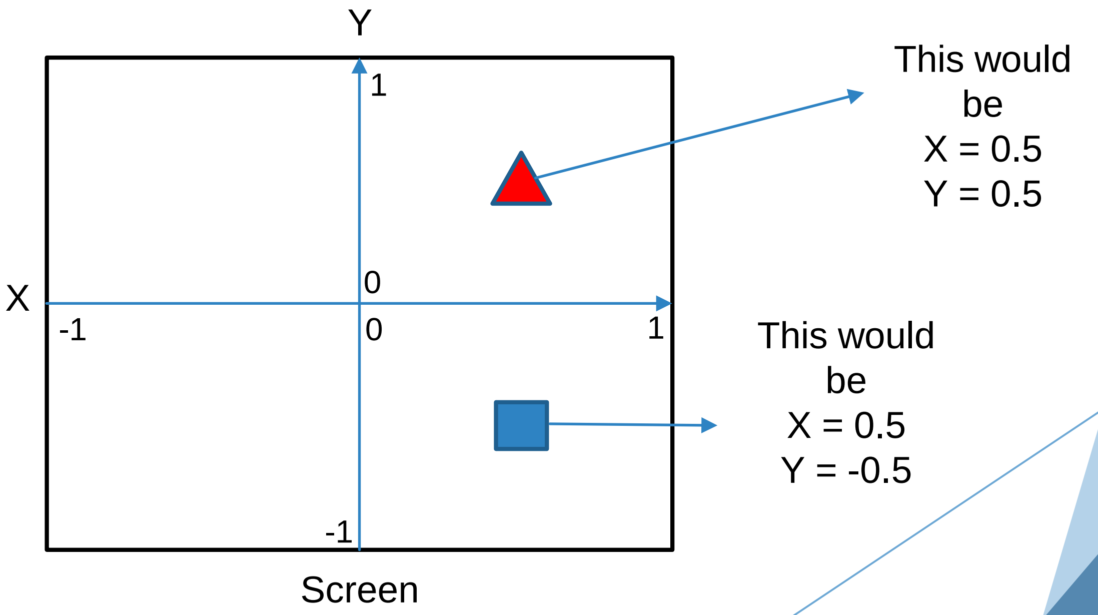
Enable Escape key ☑

Monitor settings

- PsychoPy has concept of a ‘**Monitor**’
  - Allows you to:
    - store information about multiple monitors
    - keep track of multiple calibrations for the same monitor.
- `TOOLS | MONITOR CENTER`
- Means you can:
  - specify the size and location of stimuli in units that are independent of your particular setup.e.g.
    - **pixels**
    - **cm of screen**
    - **degrees of visual angle**
  - easy to port programs to different setups as PsychoPy calculate appropriate pixel size for you

Positioning components

- Positioning of components within a routine
- Stacking order is important!
  - Refers to order they are drawn
  - Components rendered at the same time
  - Those at same position will overlap!
- Screen divided by a coordinate system
  - Other options available too which you can read about here:
  - http://www.psychopy.org/general/units.html



This would be
X = 0.5
Y = 0.5

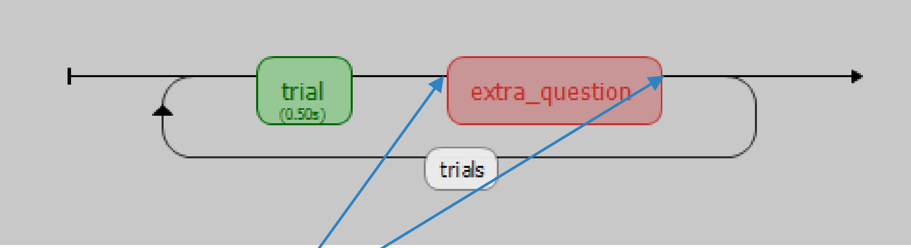
This would be
X = 0.5
Y = -0.5

Branching in PsychoPy

- Sometimes we may want to skip a particular routine
- Maybe the user should only be prompted for a particular question if they answered previously with a particular answer or you might want to let them have a rest in the middle of a long experiment?

- Conditionals in a Code block come in handy for this!

Example:



So how do we bypass this routine based on an answer under "trial"?

General best practices

- When lab testing, ensure you test your experiment in there 24 hours before

- Label your routines/loops and components clearly

- Make regular backups!!

  - We cannot stress this enough!
  - Don’t just rely on USB sticks (can be unreliable)

- Check your data output and ensure its solid
  - Make sure you are saving ALL the data you need for your analysis

Where can I get extra help?

- **PsychoPy:**
  - PsychoPy built-in HELP
    - Remember, there are separate help sections for Builder and Coder!
  - PsychoPy website – http://www.psychopy.org/index.html
  - **PsychoPy discourse** (Main support forum) - http://discourse.psychopy.org/
  - Google groups forum (mostly replaced by Discourse site) – https://groups.google.com/forum/#!forum/psychopy-users
  - Pre-made scripts by us - http://www.kent.ac.uk/psychology/technical/experiments.html
- **Python:**
  - http://www.python.org/doc/
  - Google is your friend!
- **Anything else:**
  - Psychology Technical Team (A1.2 or A1.6)
  - Check out FAQ handout in the directory (will grow over time)

Exercise 3
Turn stroop into a Picture based stroop

Simply add images to the stroop trials as a distractor.

- Open up the picture-stroop folder in your directory
  - `/picture-stroop/images` contains 6 images for you to use and link to each of the 6 trials.
- **Hints:**
  - You will need to enter a **"relative"** path for each image
    - E.g. "`images/1.jpg`" (without the quotes)
    - **Hint** - Create a new attribute called "image_path"
  - That path then needs to be passed to an **image component**
  - **Remember:** when trial data is updated, Excel link needs refreshing.
  - Attributes are accessed with **$** and **name of excel header**
- We will come round and help
- **If you don't get chance to finish, have a go at home.**

Feedback

Please take a moment to leave any feedback for future workshops at:

https://goo.gl/lBg0Em