Psychopy: a Python library for psychological experiments. Session 3 – Routines and Flows.

Jaime A. Riascos Salas

Computational Neurosciences and Cybernetics Systems (CONCYS)

Institute of Informatics

Federal University Rio Grande do Sul (UFRGS)

Porto Alegre – Brazil.

.inf INSTITUTO DE INFORMÁTICA UFRGS

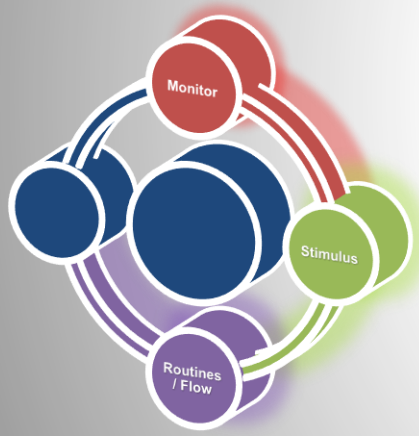
1. Routines
2. Flow
3. Loops
4. Code Component
5. Hands-on!!!

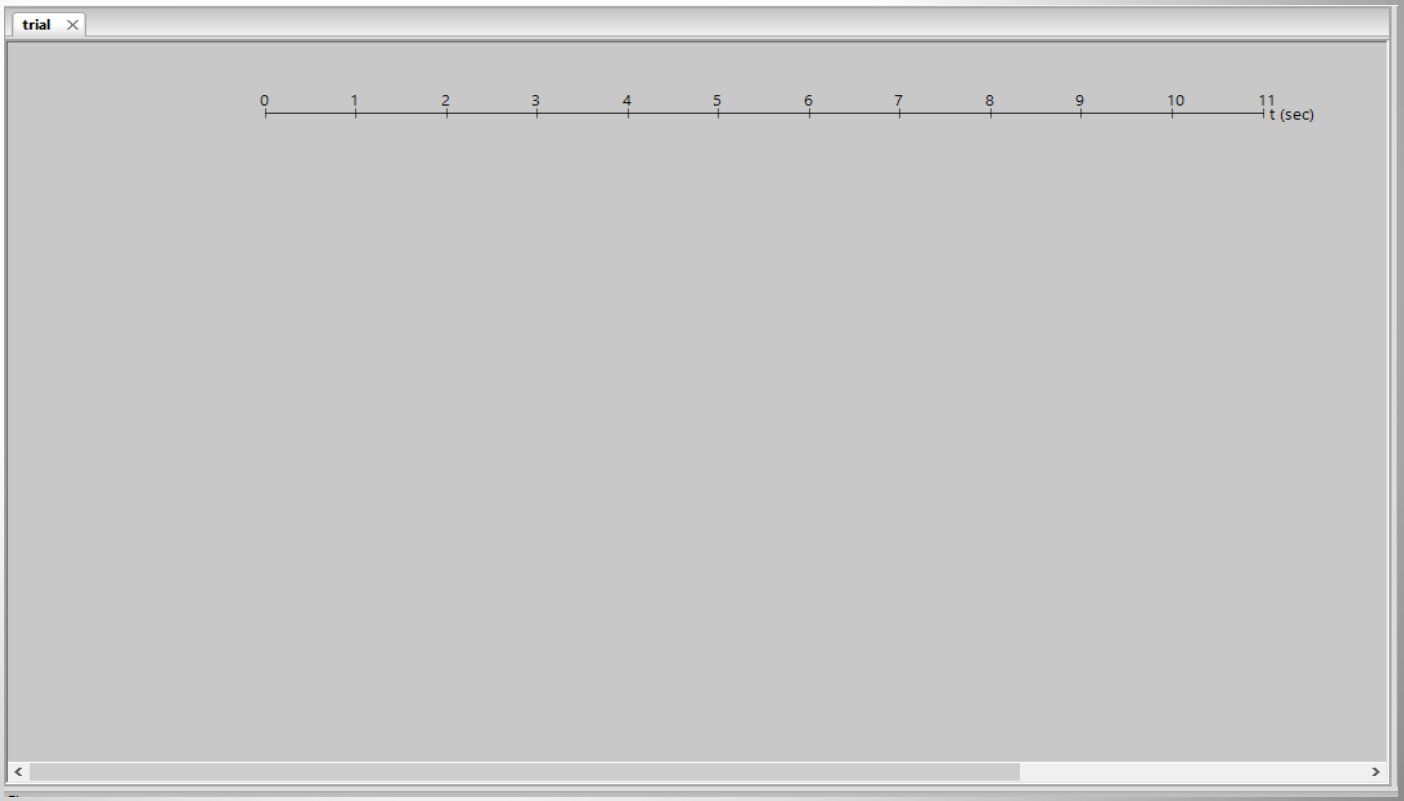
INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Routines

Did you remember this window?





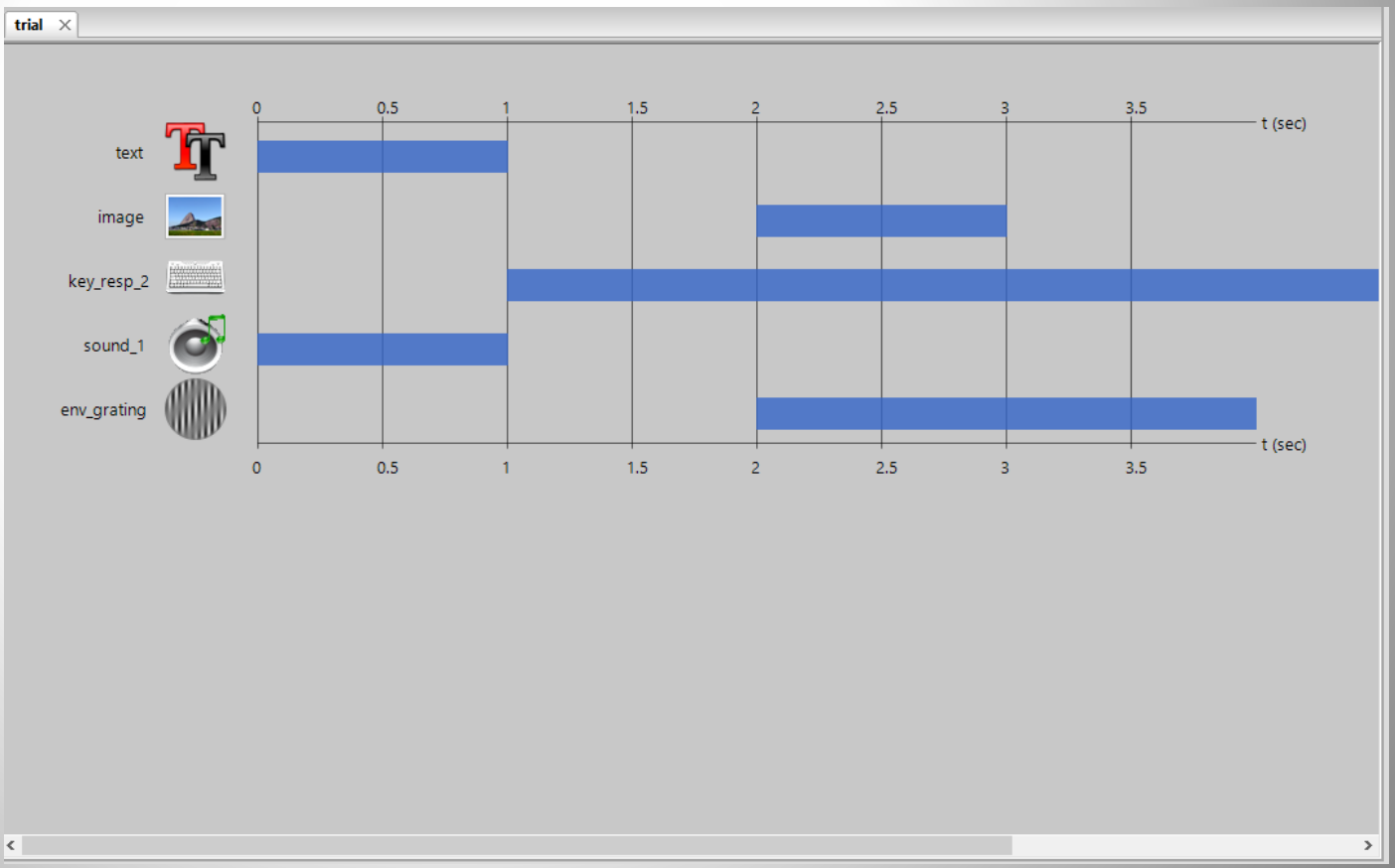
INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Routines

Did you remember this window?

A Routine might specify the timing of events within a trial or the presentation of instructions or feedback.

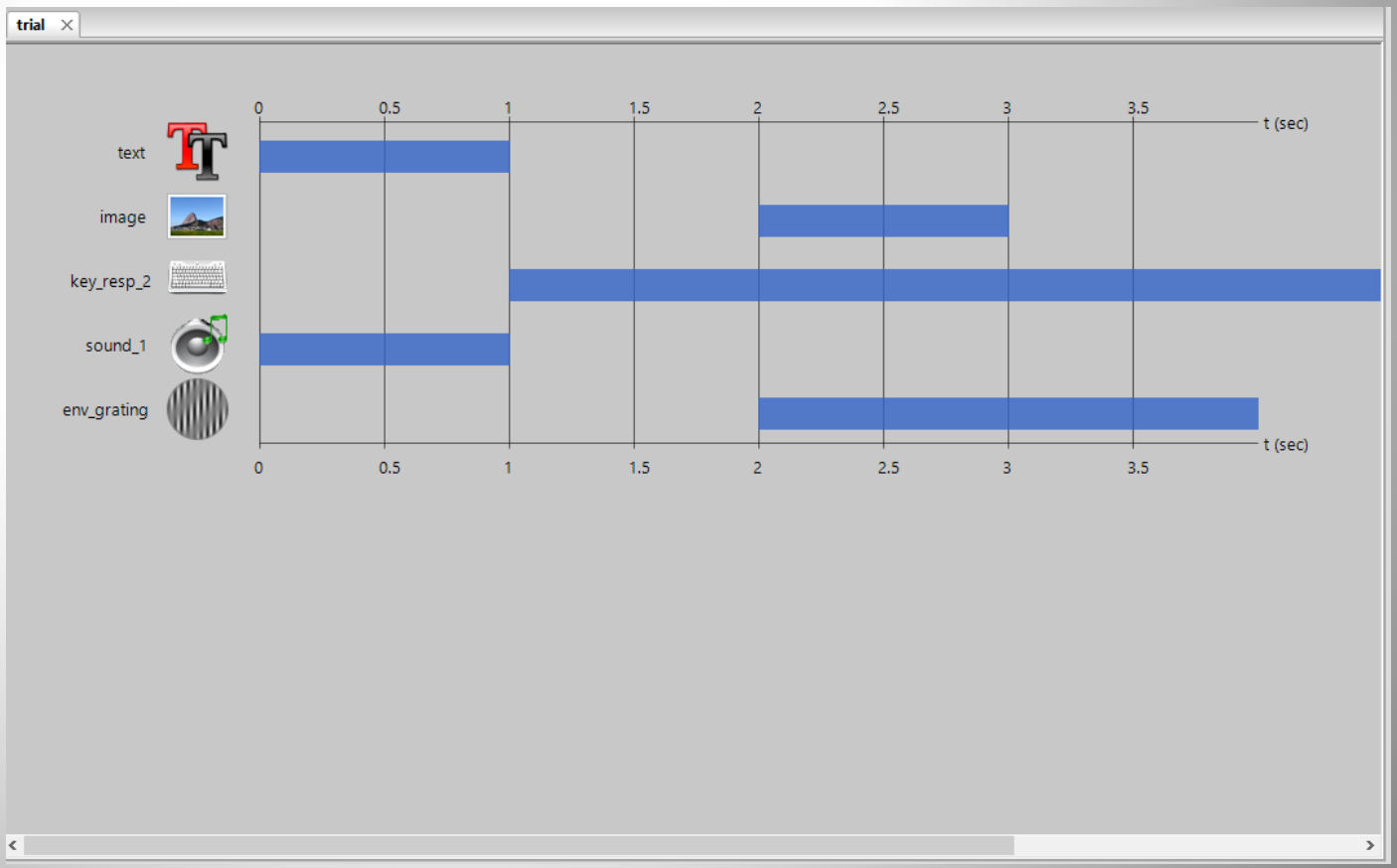


INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Routines

Did you remember this window?



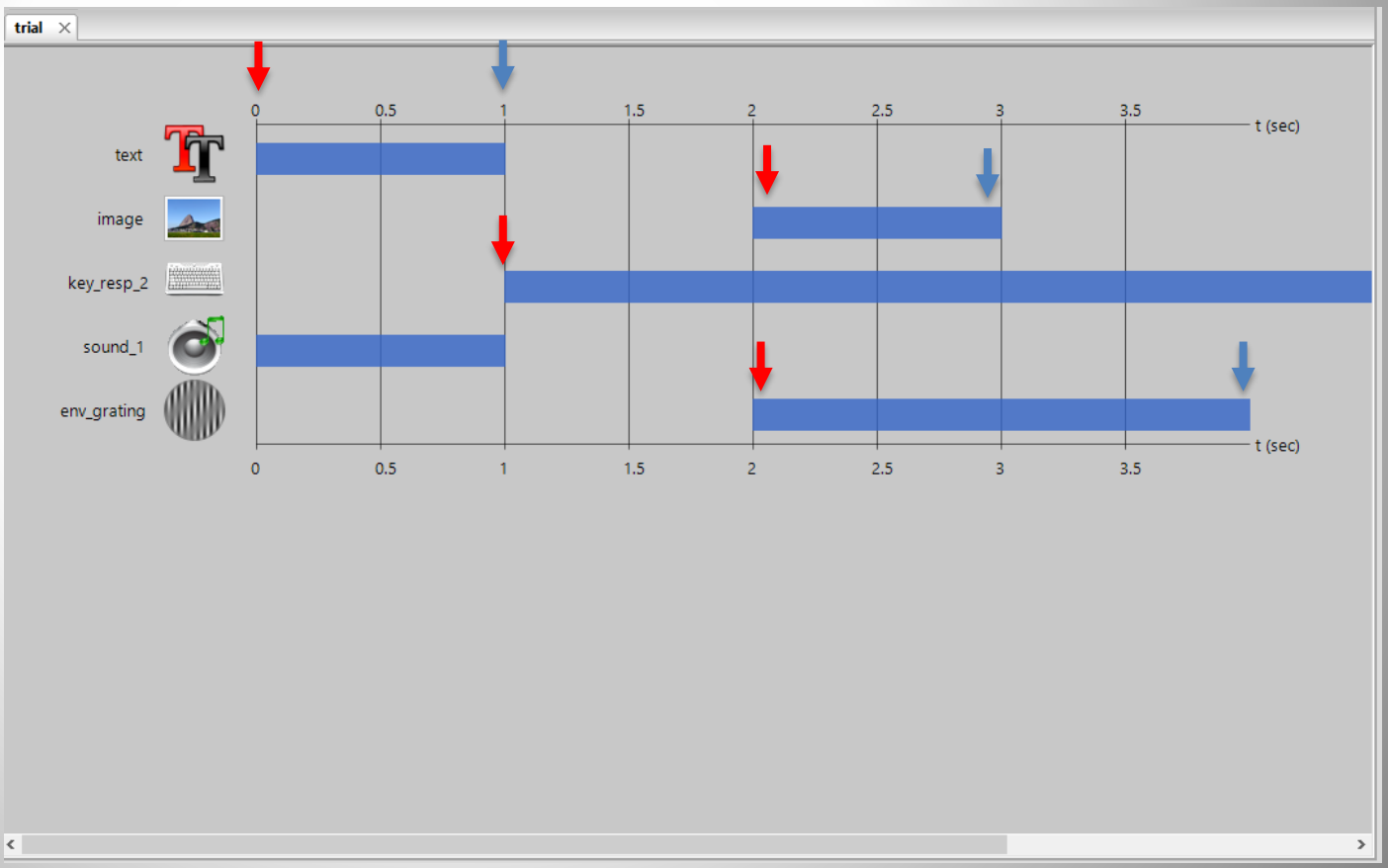
Within a Routine there are a number of components. These components determine the occurrence of a stimulus, or the recording of a response.

INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Routines

Any number of components can be added to a Routine. Each has a duration (**start** and **stop** time) which is represented by a line in the routine panel.



These object can overlap and there is also objects with a infinite duration (whole experiment);

INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Routines - Times

But… We need to set when each routine starts and finishes. This can be done in several ways:

- Use ***time(s)*** or ***frameN*** and simply enter numeric values into the start and duration boxes (or in the core.wait();
- Use a numerical variable (with $, see later in loops) into the start time;
- Use condition to cause the stimulus to start immediately after a component object, by entering *$myObjectName.status==FINISHED* into the *start* time.
- Duration can also be varied using a **Code Component** (see later).

Measuring duration in seconds (or milliseconds) is not very precise because it doesn't take into account the fact that your monitor has a fixed frame rate.
For example if the screen has a refresh rate of 60Hz you cannot present your stimulus for 120ms; the frame rate would limit you to 116.7ms (7 frames) or 133.3ms (8 frames). The duration of a frame (in seconds) is simply 1/refresh rate in Hz.

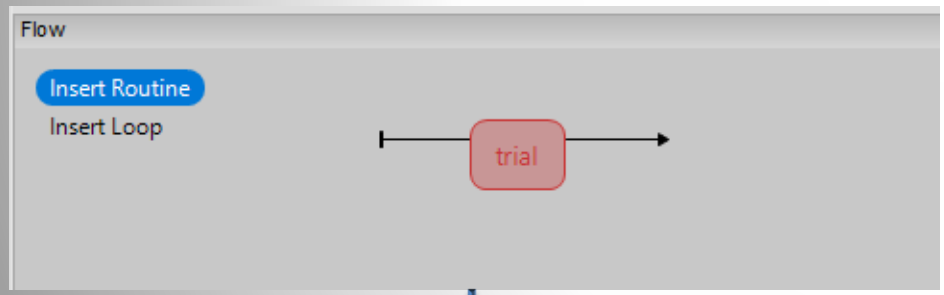
If you need very precise timing (particularly for very brief stimuli for instance) then it is best to control your onset/duration by specifying the number of frames the stimulus will be presented for.

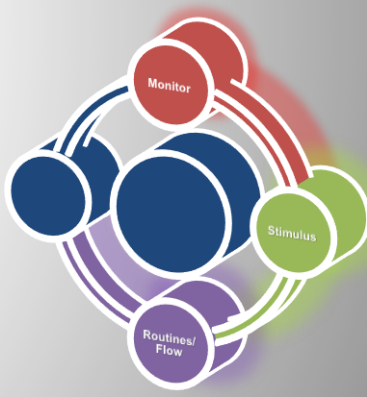
.inf INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

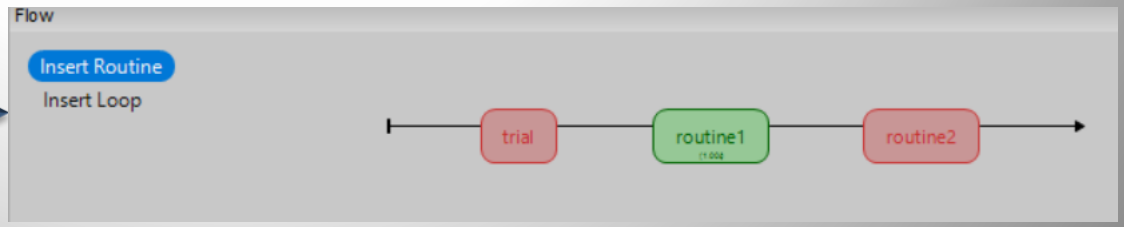
Flow

You can add routines as many as you need. Multiple Routines can then be combined in the Flow, which controls the order in which these occur and the way in which they repeat. For instance, your study may have a Routine that presented initial instructions and waited for a key to be pressed, followed by a Routine that presented one trial which should be repeated 5 times with various different parameters set. All of this is achieved in the Flow panel.





Green: Routine with a set duration.
Red: Routine without a set duration.

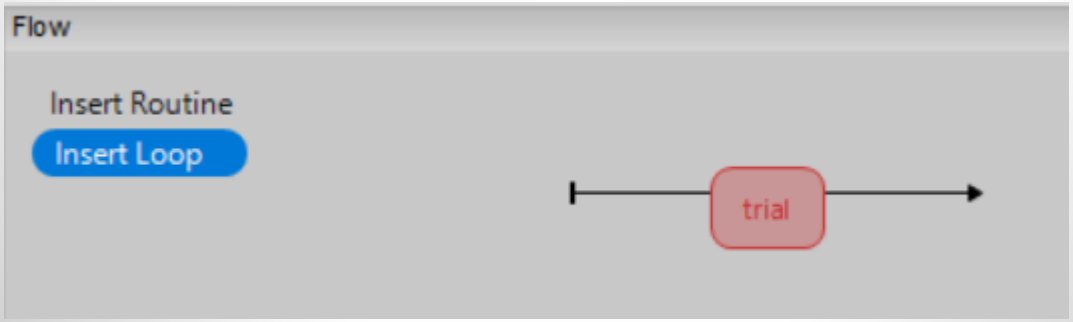


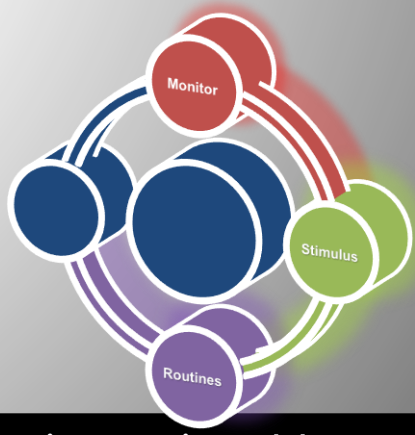
INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Loops

Loops control the repetition of Routines and the choice of stimulus parameters for each. PsychoPy can generate the next trial based on the method of constants or using an adaptive staircase. To insert a loop use the button on the left of the Flow panel, or the item in the Experiment menu of the BuilderLoops can encompass one or more Routines and other loops (i.e. they can be nested) as was seen in the Python lecture.

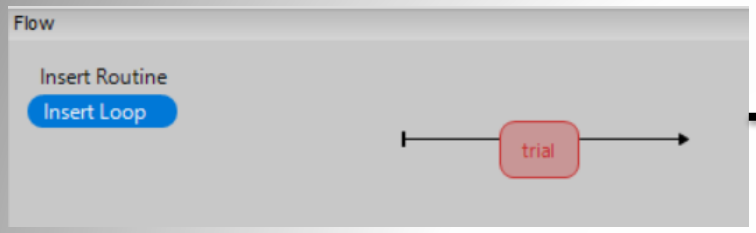


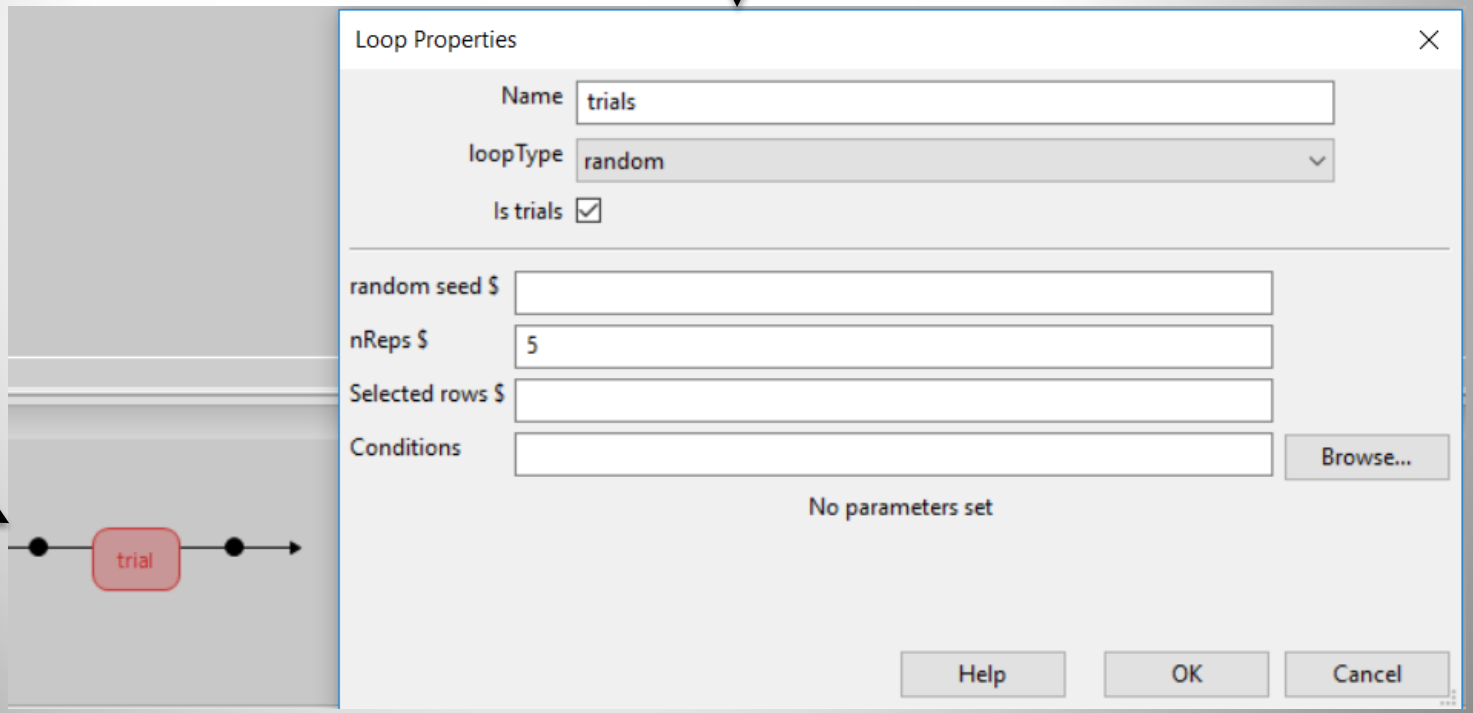


INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Loops





You must choose where the loop starts and finishes.

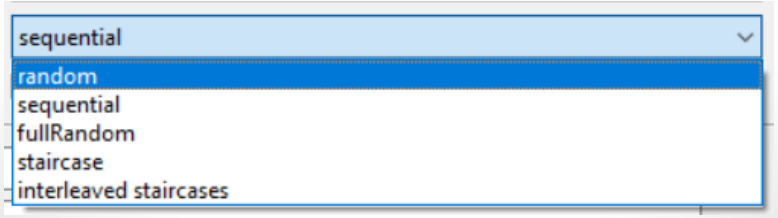
INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Loops - Types

Sequential, Random and Full Random

If the loop type is *sequential* then, on each iteration through the Routines, the next item will be selected in the order listed in the file. Under a *random* order, the next item will be selected at random (without replacement); it can only be selected again after all the other items have also been selected. *nReps* determines how many repeats will be performed (for all conditions).
With the *fullRandom* option, the entire list of trials including repetitions is used in random order, allowing the same item to appear potentially many times, and to repeat without necessarily having done all of the other trials.



INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Loops - Types

Sequential, Random and Full Random

For example, in a list [a,b,c], with three repetitions…

- ***sequential*** could only ever give one sequence with this order: [a b c a b c a b c];
- ***random*** will give one of 216 different orders (= 3! * 3! * 3! = nReps * (nTrials!) for example: [b a c a b c c a b]. Here the letters are effectively in sets of (abc) (abc) (abc), and randomization is only done within each set;
- ***fullRandom*** will return one of 362,880 different orders (= 9! = (nReps * nTrials)!), such as [b b c a a c c a b], which *random* never would, indeed, it is possible to obtain the unrandom-looking sequence like [a a a b b b c c c];

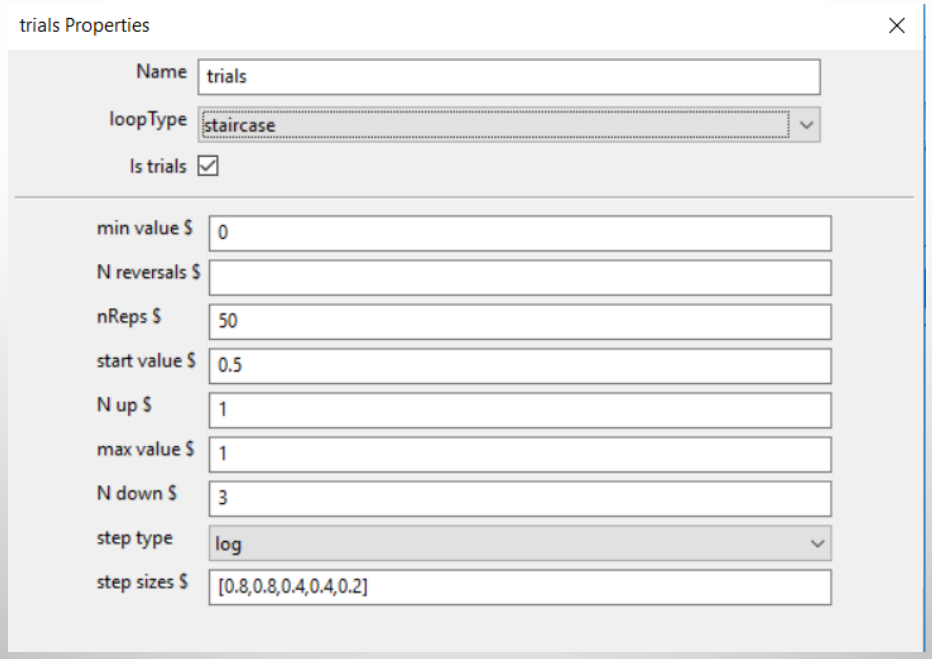
INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Loops - Types

Staircase methods

The loop type *staircase* allows the implementation of adaptive methods. That is, aspects of a trial can depend on (or “adapt to”) how a subject has responded earlier in the study. This could be, for example, simple up-down staircases where an intensity value is varied trial-by-trial according to certain parameters, or a stop-signal paradigm to assess impulsivity. For this type of loop a ‘correct answer’ must be provided from something like a Keyboard Component.



See later in Hands-on section

.inf INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Loops - Conditions

Constants methods

Selecting a loop type of random, sequential, or fullRandom will result in a method of constants experiment, whereby the types of trials that can occur are predetermined. That is, the trials cannot vary depending on how the subject has responded on a previous trial.

In this case, a file must be provided that describes the parameters for the repeats. This should be either Excel 2007 (xlsx), comma-separated-value (csv) or pick-up files in which **columns** refer to parameters that are needed to describe stimuli etc. and **rows** one for each type of trial. The top row should be a row of headers: text labels describing the contents of the respective columns.

.inf INSTITUTO DE INFORMÁTICA UFRGS

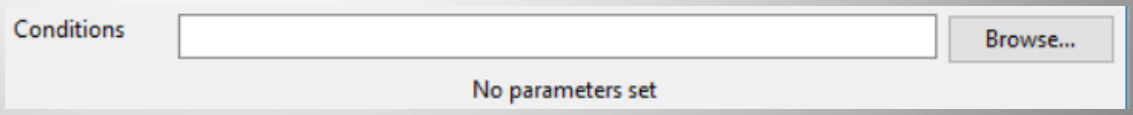
PsychoPy
Psychology software in Python

Loops - Conditions

Constants methods

For example, the next table has four different conditions (or trial types, one per line). The header line describes the parameters in the three columns: ori, text and corrAns. It's really useful to include a column called corrAns that shows what the correct key press is going to be for this trial (if there is one).

Ori	Text	corrAns
0	aaaa	left
90	aaaa	left
0	bbbb	right
90	bbb	right



INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Loops - Slicing

Selecting a subset of conditions

In the standard Method of Constants you would use all the rows/conditions within your conditions file. However there are often times when you want to select a subset of your trials before randomizing and repeating.

The parameter *Select rows* allows this. You can specify which rows you want to use by inserting values here:

- *0,2,5* gives the 1st, 3rd and 5th entry of a list - Python starts with index zero)
- *random(4)*10* gives 4 indices from 0 to 10 (so selects 4 out of 11 conditions)
- *5:10* selects the 6th to 9th rows
- *$myIndices* uses a variable that you've already created

.inf INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Loops - Slicing

Selecting a subset of conditions

In the standard Method of Constants you would use all the rows/conditions within your conditions file. However there are often times when you want to select a subset of your trials before randomizing and repeating.

The parameter *Select rows* allows this. You can specify which rows you want to use by inserting values here:

- *0,2,5* gives the 1st, 3rd and 5th entry of a list - Python starts with index zero)
- *random(4)*10* gives 4 indices from 0 to 10 (so selects 4 out of 11 conditions)
- *5:10* selects the 6th to 9th rows
- *$myIndices* uses a variable that you've already created

.inf INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Loops - variables

Accessing loop parameters from components

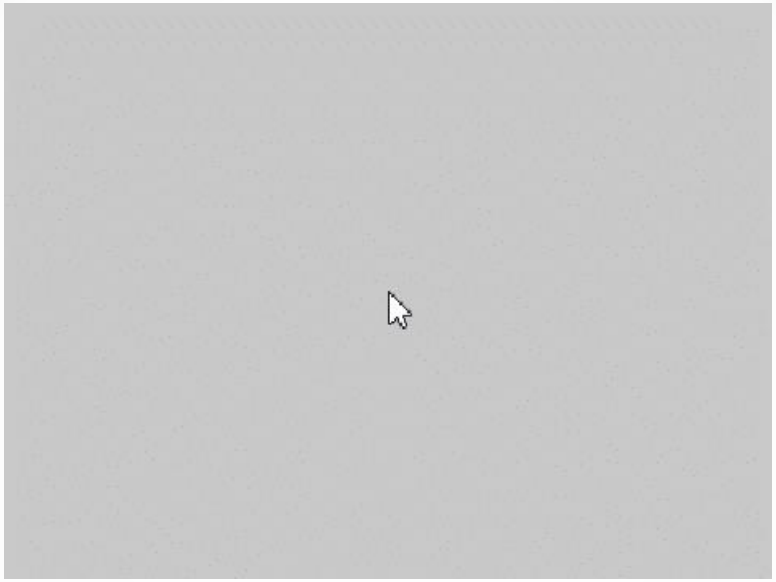
The parameters from your loops are accessible to any component enclosed within that loop. The simplest (and default) way to address these variables is simply to call them by the name of the parameter, prepended with **$** to indicate that this is the name of a variable. For example, if your Flow contains a loop with the above table as its input trial types file then you could give one of your stimuli an orientation **$ori** which would depend on the current trial type being presented. Example scenarios:

- You want to loop randomly over some conditions in a loop called trials (table above). You can then access these values from any component using $ori, $text, and $corrAns
- You create a random loop called blocks and give it a file with a single column called movieName listing filenames to be played. On each repeat you can access this with $movieName
- You create a staircase loop called stairs. On each trial you can access the current value in the staircase with $thisStair

.inf INSTITUTO DE INFORMÁTICA UFRGS

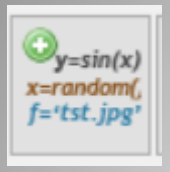
PsychoPy
Psychology software in Python

Loops



Note

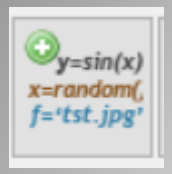
When you set a component to use a parameter that will change (e.g on each repeat through the loop) you should **remember to change the component parameter from `constant` to `set every repeat` or `set every frame`** or it won't have any effect!



Adding coding objects in the builder

The *Code Component* can be used to insert short pieces of python code into your experiments. This might be create a variable that you want for another Component, to manipulate images before displaying them, to interact with hardware for which there isn't yet a pre-packaged component in *PsychoPy* (e.g. writing code to interact with the serial/parallel ports).

Be aware that the code for each of the components in your Routine are executed in the order they appear on the Routine display (from top to bottom**). If you want your *Code Component* to alter a variable to be used by another component immediately, then it needs to be above that component in the view.** You may want the code not to take effect until next frame however, in which case put it at the bottom of the Routine.



Adding coding objects in the builder

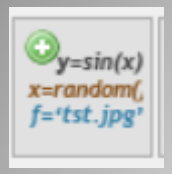
What can we do with coding objects?

Within your code you can use other variables and modules from the script. For example, all routines have a stopwatch-style Clock associated with them, which gets reset at the beginning of that repeat of the routine. So if you have a Routine called trial, there will be a Clock called trialClock and so you can get the time (in sec) from the beginning of the trial by using:

currentT = trialClock.getTime()

Some examples:

- **Set a random location for your target stimulus**
- **Create a patch of noise**
- **Send a feedback message at the end of the experiment**
- **End a loop early.**



Adding coding objects in the builder - Parameters

The parameters of the Code Component simply specify the code that will get executed at 5 different points within the experiment. You can use as many or as few of these as you need for any Code Component:

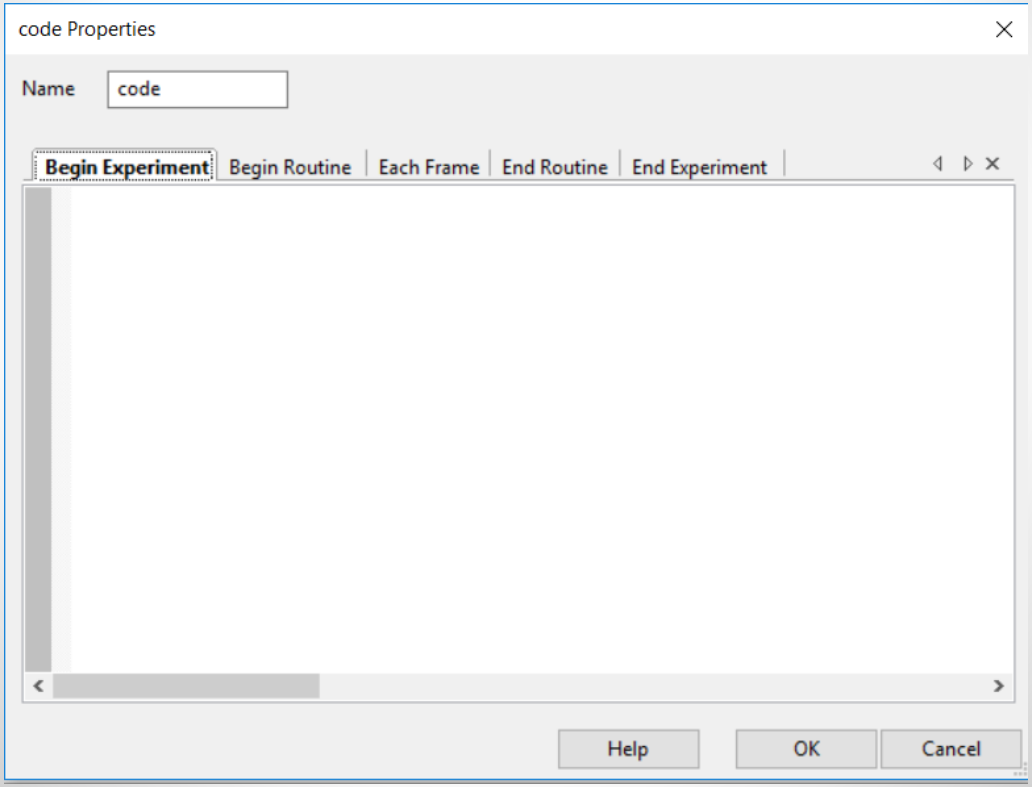
- **Begin Experiment:** Things that need to be done just once, like importing a supporting module, initialising a variable for later use.
- **Begin Routine:** Certain things might need to be done just once at the start of a Routine e.g. at the beginning of each trial you might decide which side a stimulus will appear
- **Each Frame:** Things that need to updated constantly, throughout the experiment. Note that these will be executed exactly once per video frame (on the order of every 10ms), to give dynamic displays. Static displays do not need to be updated every frame.
- **End Routine**: At the end of the Routine (e.g. the trial) you may need to do additional things, like checking if the participant got the right answer
- **End Experiment:** Use this for things like saving data to disk, presenting a graph(?), or resetting hardware to its original state.

.inf INSTITUTO DE INFORMÁTICA UFRGS

http://psychopy.org/builder/components/code.html#code-component

PsychoPy
Psychology software in Python

Adding coding objects in the builder



INSTITUTO DE INFORMÁTICA UFRGS

PsychoPy
Psychology software in Python

Please open the next demos:

-Stroop
-Stroop extended
-key_Name_Finder
-psychopysicsStaircase

INSTITUTO DE INFORMÁTICA UFRGS

Any questions… ?