# Learning Objectives

1. To learn how to include participant feedback to an experiment using code
2. Using conds files to present word stimuli
3. To work on your assessment with in-class support

# Activity 7.0

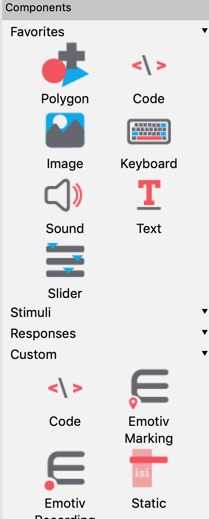
## Participant feedback

In some circumstances you might want to provide feedback for your participants. Feedback is particularly useful with children as it can elicit a reward type response, but equally it can be used to keep children on task during an experiment.

It can also help adult participants understand if they have performed a task correctly (or not as the case may be) so with adults we typically only use feedback in practice trials. A third way that feedback can be used is to screen out participants who have ‘failed’ the practice tasks.

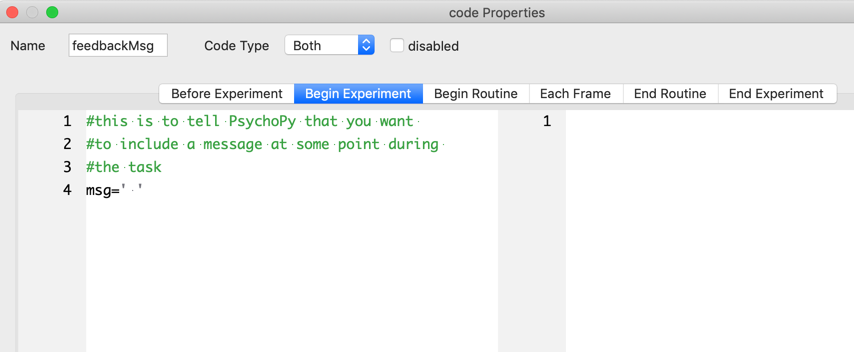
**To add feedback, you will need to use a small amount of code**.

1. Using the copy of the **CRT.psyexp** that you added a practice block to in **Seminar 5**, you should now save it in a new folder called **EDP7**
2. Also copy the stims folder and all conds files to this new folder
3. In order to provide feedback, we must ensure that the **pracStimResp** is set to collect data. On the **data** tab of the **pracStimResp** set the **Correct answer** box to reference the **corrAns** field in the **prac** conditions file
4. Create a **new routine**, call it **pracFeedback** and place it after the **prac** routine in the CRT experiment
5. Add a **code component** to the **feedback** routine using the code icon that is circled in Figure 1



Figure

1. This will automatically open the code properties dialog box. In the **Begin Experiment** tab you should type in the text as per Figure 2

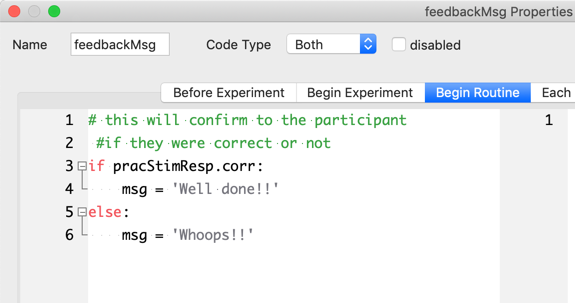


Figure

**The # (hash key) is used to make comments in code in PsychoPy and they will not be seen by the participant. If you make a comment on a new line, you need to use a new # to identify that it is not a part of the code that the program will execute.**

1. In the **Begin Routine** tab, type in the code **EXACTLY** as per Figure 3
2. This bit of code is telling the program that if the response to the keyboard press was correct (as stored on the last trial that ran), show the message that the participant got that response correct, and if it was wrong display the error message.

Figure

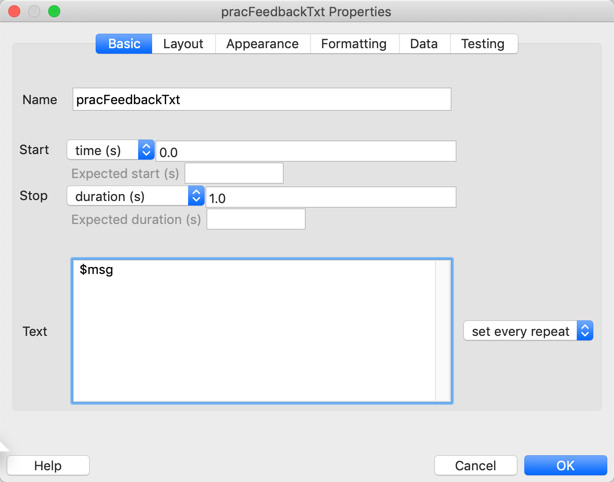


**NOTE: if you named your pracStimResp differently you should type in the exact name from your keyboard response component**

1. Add a **text component** to the **feedback routine**, call it **pracFeedbackTxt** and setthe text duration to **1000ms**
2. On the **Basic** tab from **pracFeedbackTxt** properties to include the message in the text component by typing **$msg** and ensure that you **set to every repeat** as per Figure 4.

Setting the dialog box to **every repeat** means that PsychoPy will check the last response every time a new stimulus is shown. Setting the text to **$msg** means that PsychoPy is referencing the code that you wrote in the **Begin Routine** tab of the **feedbackMsg** code

Figure



1. Change the text colour to **black**
2. Change the font size to **24**
3. Run the experiment to check that the feedback works

# Activity 7.1

## Presenting words using conditions files

1. Create a new blank experiment and save it in a new root folder called wordPres
2. Create a single routine with 3 text components, all to begin at the start of the routine and end with a keyboard response
3. Create a conds file in Excel. Apply the headings leftword1, midword2, rightword3 and condition, corrAns
4. Populate leftword1 and rightword3 columns with 10 words that relate to food, and 10 words that denote animals
5. Populate word2 (the central stimulus) with the words animal and food (10 of each)
6. Using the positioning information from last week position the words in left, middle and right positions
7. Run the experiment

# Activity 7.2

## Work on your assessment in class

If you have completed this activity you should work on your assessment in-class where you can ask for support from your tutor.