

Collector Documentation

Documentation by Anthony Haffey on behalf of [Some Open Solutions](#).

If you identify errors or have suggested clarifications, please go to <https://collectalk.com/categories/documentation>

A program for running experiments on the web Copyright 2012-2015 Mikey Garcia & Nate Kornell

Kitten-release (2019-2020) Anthony Haffey

The most up to date version of this documentation can be found at:
https://docs.google.com/document/d/1SKYIJF1dAjMDS6EHUlwZm2KQVOzx17S6LbU_oSGxdE/edit?usp=sharing

Collector Documentation	1
Using Collector online (i.e. without installing Collector)	3
Dropbox registration	3
Creating experiments	13
Conditions sheet	13
Trial type editor	14
How it looks	17
Timeline	18
Capturing response with keyboard	20
Capturing the response with the mouse	21
Changing the canvas size	22
Procedure sheet	22
Shuffle 1	23
Trial type	24
Item	24
Max time	24
Text	24
Stimuli sheet	25
List Stimuli button	25
Helper sidebar	26
Downloading and Uploading experiments	27
Running your study	27
Save!	27

Click the “run” button	27
Publishing your task	28
Decrypting your data (and understanding it)	30
Decrypting your data within the “Data” tab	32
Understanding your data	33
Surveys	33
FAQ/General questions	34
Can I use Collector if I am not a programmer?	34
What language is Collector written in?	34
Do I need to ask anyone if I want to use Collector?	35
Troubleshooting	35
Further notes	35
What happens with my password for encryption and decryption	35
What happens with my password for Collector	36
Original contributors	36
Tyson Kerr	36
Victor Sungkhasettee	37
Adam Blake	37
Nate Kornell	37

Using Collector online (i.e. without installing Collector)

To use without installation go to either:

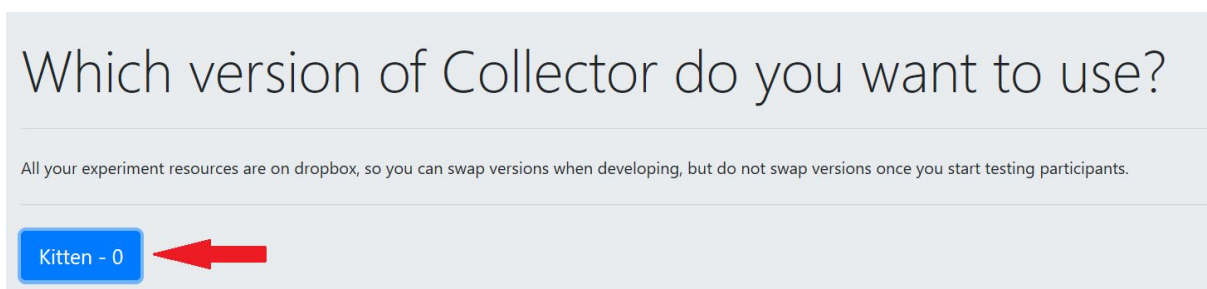
- <https://ocollector.org> (allows you to choose which release of Kitten you want)
- <https://some-open-solutions.github.io/collector> (will automatically take you to the most recent version of kitten that is stable)

To use without installing, you will need to register with Dropbox and Collector

Dropbox registration

Go to: <https://ocollector.org> once there select a version of Collector to use. As it says on the page, it is fine to swap between versions of Collector when creating experiments, but **DO NOT SWAP WHEN TESTING PARTICIPANTS!**

So let's go to ocollector.org:



Which should take you to the home page. Just click on "Kitten" to proceed to Collector:

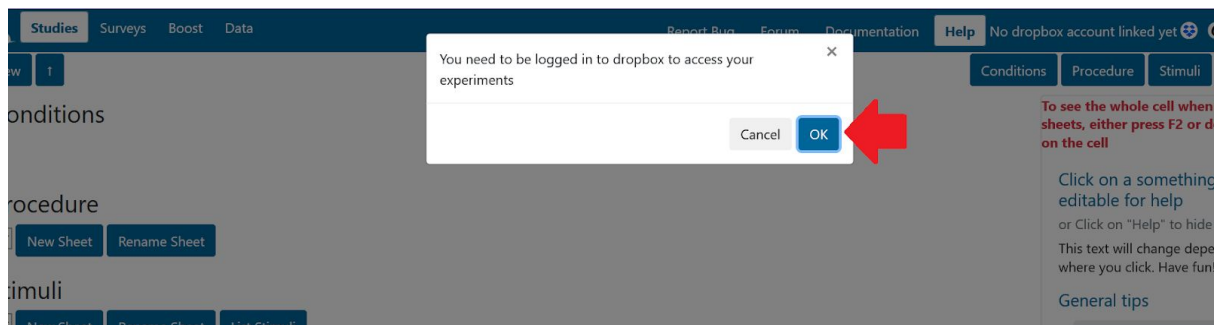
Collector

A free tool for online data collection

Select which version you would like to use:



You should then be asked to login in with dropbox. Just click on “OK” (you’ll have the chance to register an account later). If you accidentally click “Cancel” just refresh the page.

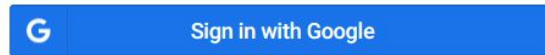


You will then be asked to sign in to dropbox. **I strongly recommend that you do this with a NEW GOOGLE ACCOUNT (which you’ll create next).** The reason for this is that I would suggest you keep your research resources separate from your personal or other accounts. So a new account does this, and google make it easy to do this by creating a free account with them.

To do this click on “Sign in with Google”:



Sign in to Dropbox to link to Collector-SOS



or

This page is protected by reCAPTCHA and is subject to the Google [Privacy Policy](#) and [Terms of Service](#).

[Forgotten your password?](#)



Now you need to create your new google account. Click on the “Create account” button:

 Sign in with Google



Sign in

Continue to **Dropbox**

Email or phone

|


[Forgot email?](#)


To continue, Google will share your name, email address, language preference and profile picture with Dropbox. Before using this app, you can review Dropbox's [privacy policy](#) and [terms of service](#).

[Create account](#)


[Next](#)

And then go through the process of creating an account. Once you have registered with google, you'll get to the following page, where you'll click "Allow":


 Sign in with Google




Dropbox wants to access your
Google Account

 anthonywoz2020@gmail.com

This will allow **Dropbox to:**

 See and download your contacts




Make sure that you trust Dropbox

You may be sharing sensitive info with this site or app. Find out how Dropbox will handle your data by reviewing its [terms of service](#) and [privacy policies](#). You can always see or remove access in your [Google Account](#).

[Find out about the risks](#)

Cancel

Allow



You should get to the following page, in which you'll be able to select a **free/basic** option:

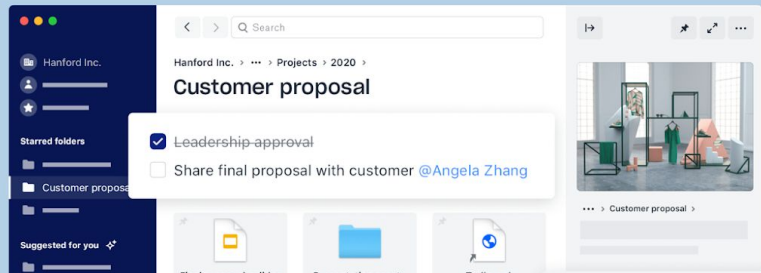
Work comes together in Dropbox Business

Organise all of your team's content, tune out distractions and get everyone coordinated with the world's first smart workspace.

Try free for 30 days

Purchase now

or get Dropbox Basic



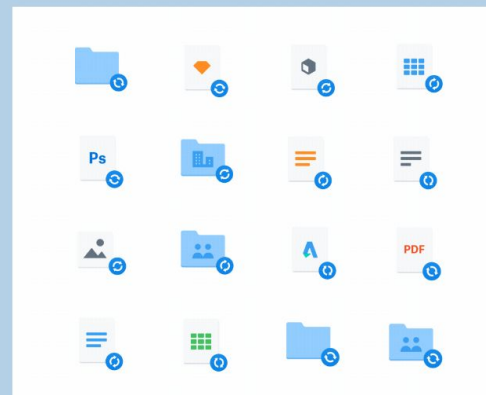
Then click on “Sign up for free”:

Dropbox Basic


Get a Dropbox free account

Sign up for 2 GB of storage, and start accessing and sharing photos, documents and other files from any device.

Sign up for free



And Now you can register with dropbox with your newly created google account by clicking “Sign up with Google”:



Create an account [or log in](#)

First name

Surname

Email


Password

This page is protected by reCAPTCHA and is subject to the Google [Privacy Policy](#) and [Terms of Service](#).

☐ I agree to the [Dropbox Terms](#).

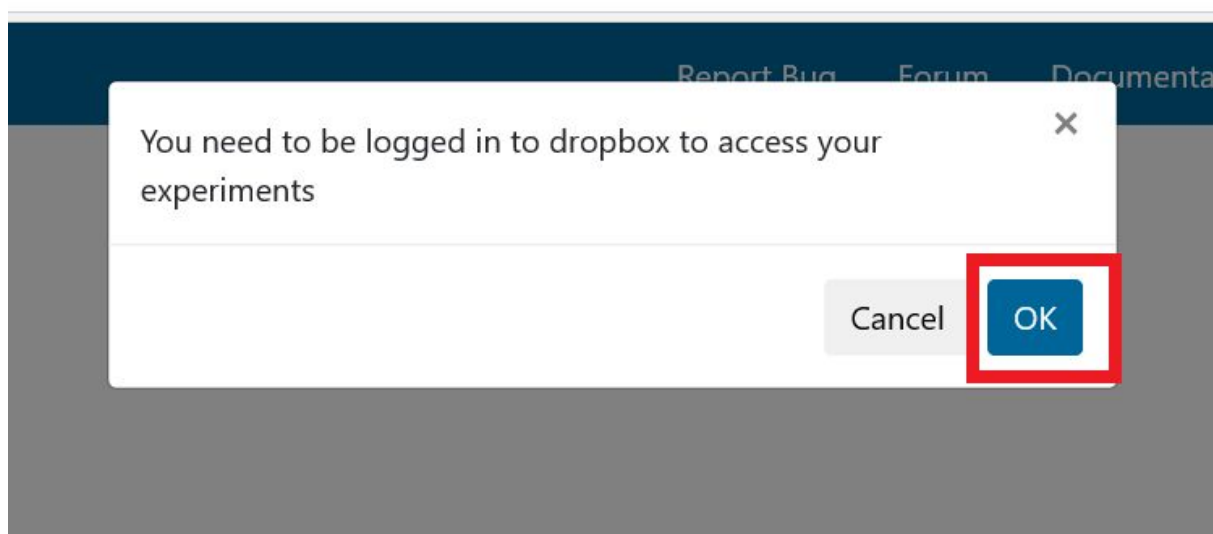
Create an account

or

 Sign up with Google

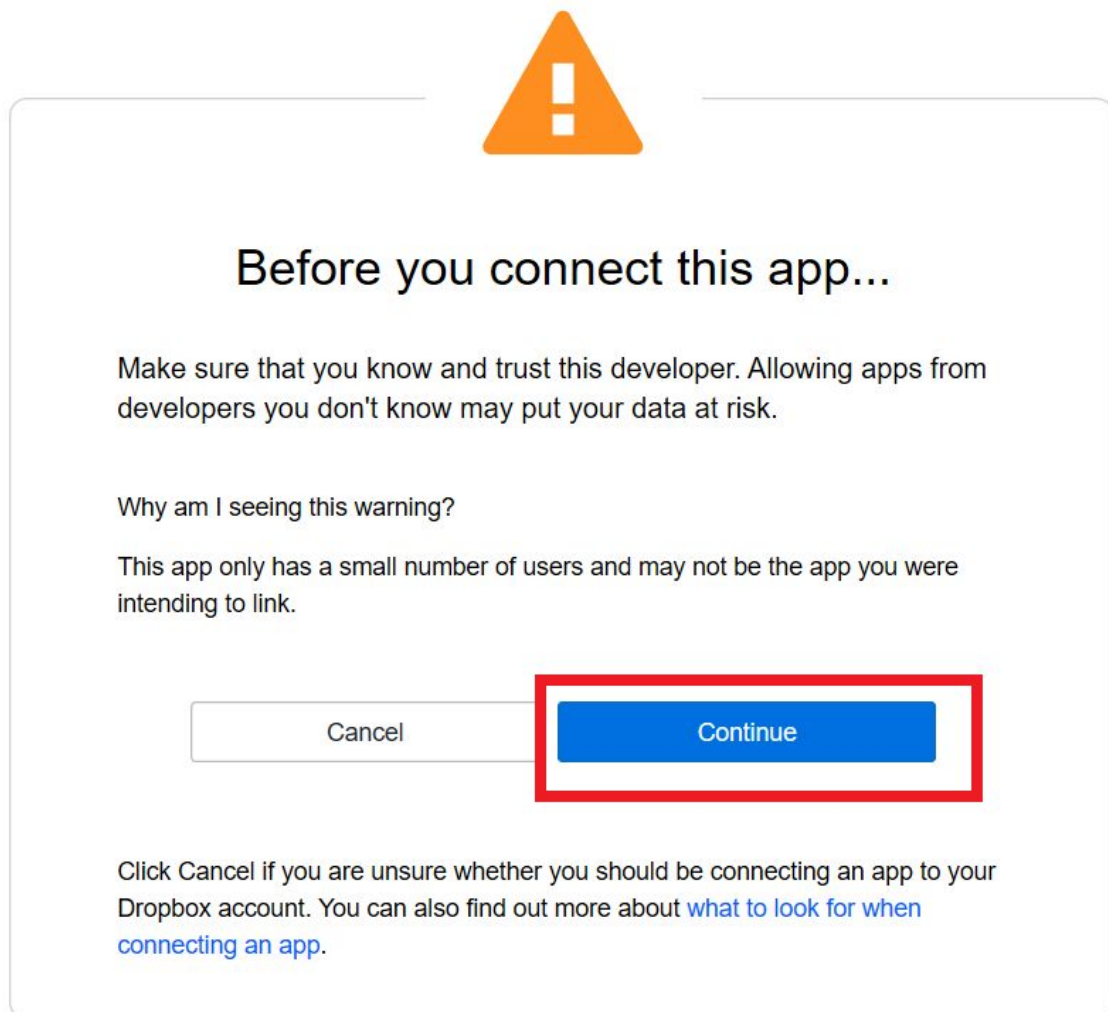
- Now choose the account you just created,
- Then click “Allow”.
- Then you should be asked to sign up. You just need to accept Dropbox terms and click “Sign up”.

You should arrive at your home page on dropbox. Let's go back to ocollector.org until you're back on the version you chose to use. Once you get back to something like [“https://ocollector.org/0/kitten/”](https://ocollector.org/0/kitten/) you should see the same request you had at the start of this process. Click on “OK” again:

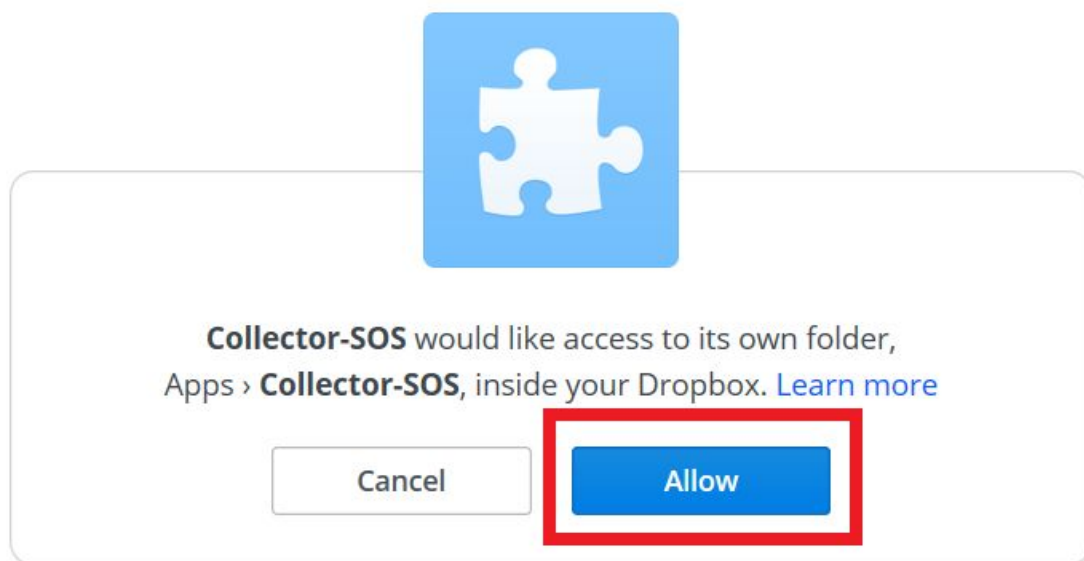


And this time you'll be asked if you want to connect Collector-SOS to your new dropbox account. Select "Continue":

(If you have been using your personal account so far, you might want to go back a few steps if you find the next message makes you uncomfortable with using personal accounts for Collector)



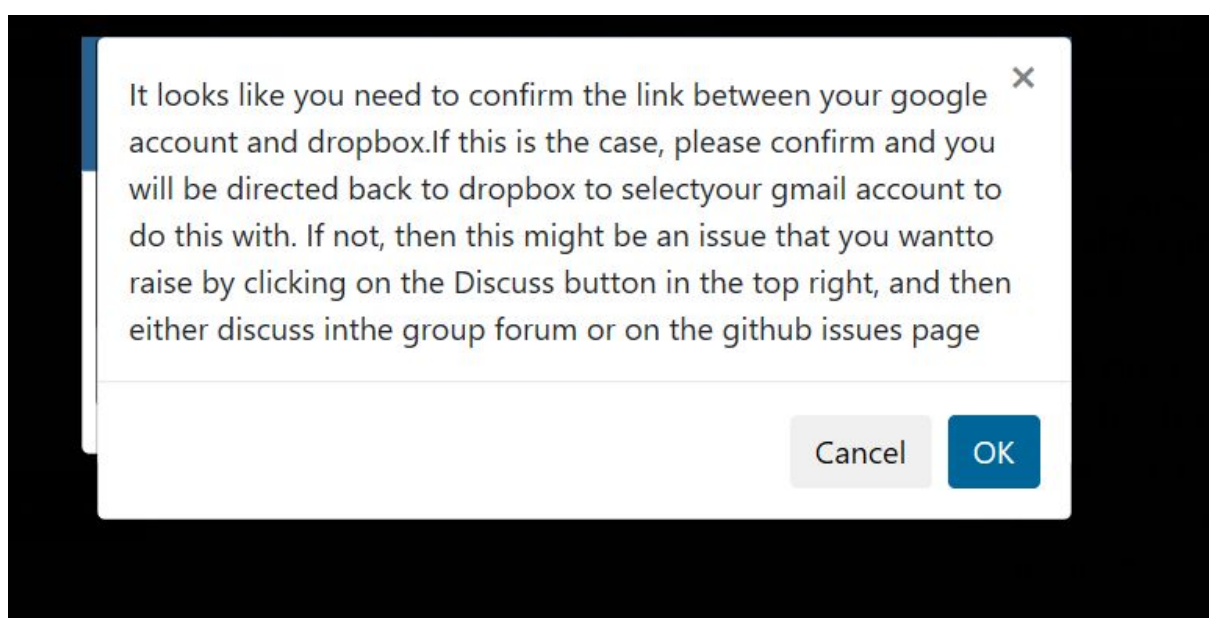
Now you'll be asked if you want Collector-SOS (some open solutions) to have it's own folder. This means that Collector will only ever edit files within this folder, and never anywhere else on dropbox. Select "Allow":



It is possible that you'll see quite a few messages, and need to close the front one by clicking on the X in the top right corner of it.

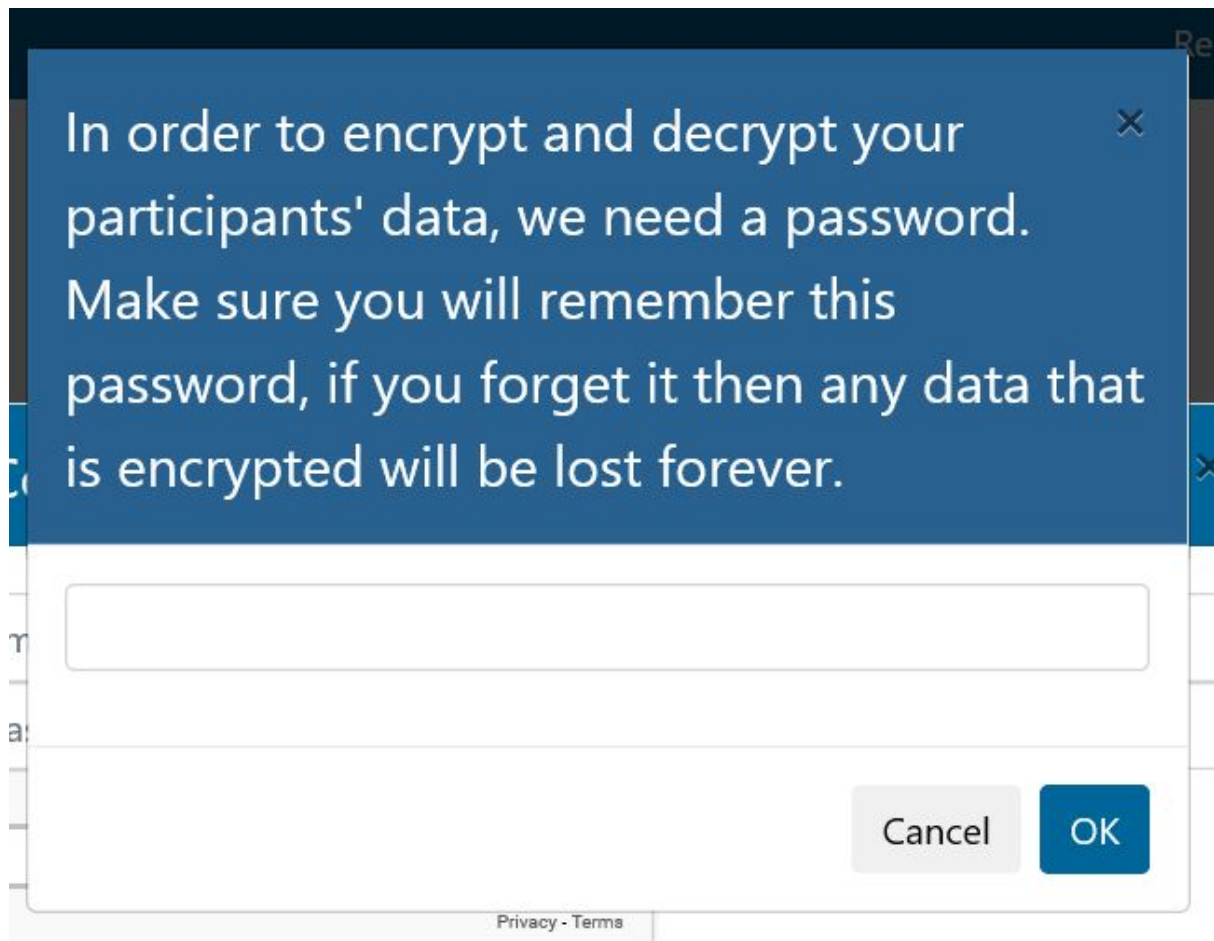


Once you've done that you might see the following error message:



If you click on “OK” and sign in again, this should fix the connection between dropbox and your google account.

Once that works, you should be able to start settings some of your settings, such as your password for data encryption and decryption. This should be asked for automatically:



Think carefully about what your password will be. **Do not use a password you use for anything else!** Here are some reasons why:

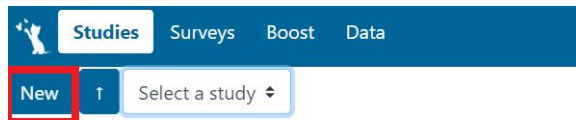
- You may want to share this dropbox account with a collaborator
- Whenever possible, you should not use the same password on different websites.

However, having said that, your password will never be stored (see [What happens with my password for encryption and decryption](#) for clarifications of how your password is used to decrypt data).

Creating experiments

For the purposes of this documentation, I'll be using the example of creating a stroop task called "stroop_example".

To create a new experiment just click on the “New” button at the top left of the screen:

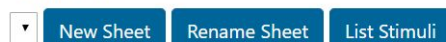


Conditions

Procedure



Stimuli



If you're developing your experiment online rather than with an installation, give it a few seconds for the experiment creation to finish. You'll know it's ready because the new name will be where "Select a study" was before:



Conditions sheet

The conditions sheet controls some of the settings of the experiment. For example, if you want the user to have a randomised id rather than what they typed, then change the cell just below “participant_id” from “on” to “random”:

Conditions

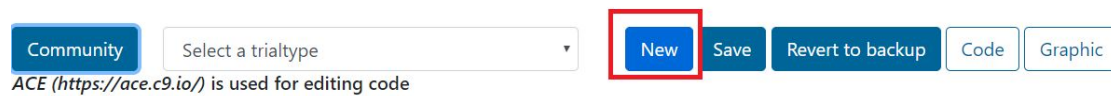
[illegible][illegible]

Have a look around at the different settings (notice how the “Help” sidebar tells you about each of them as you click across different columns).

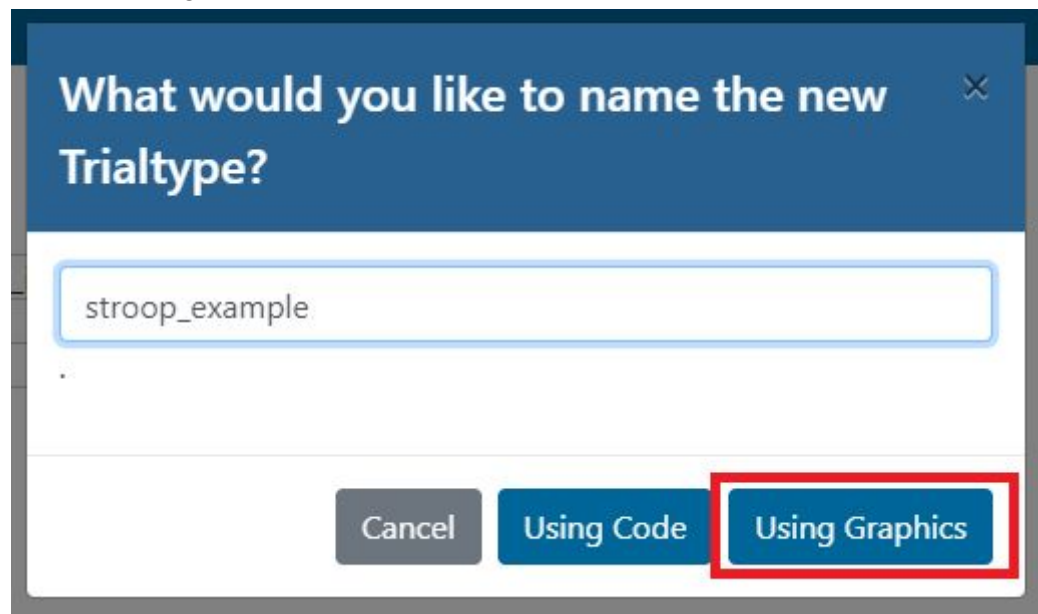
If you want to have different participants experience different conditions, then describe the differences in the different rows. This will result in you creating slightly different links for each version of the task.

Trial type editor

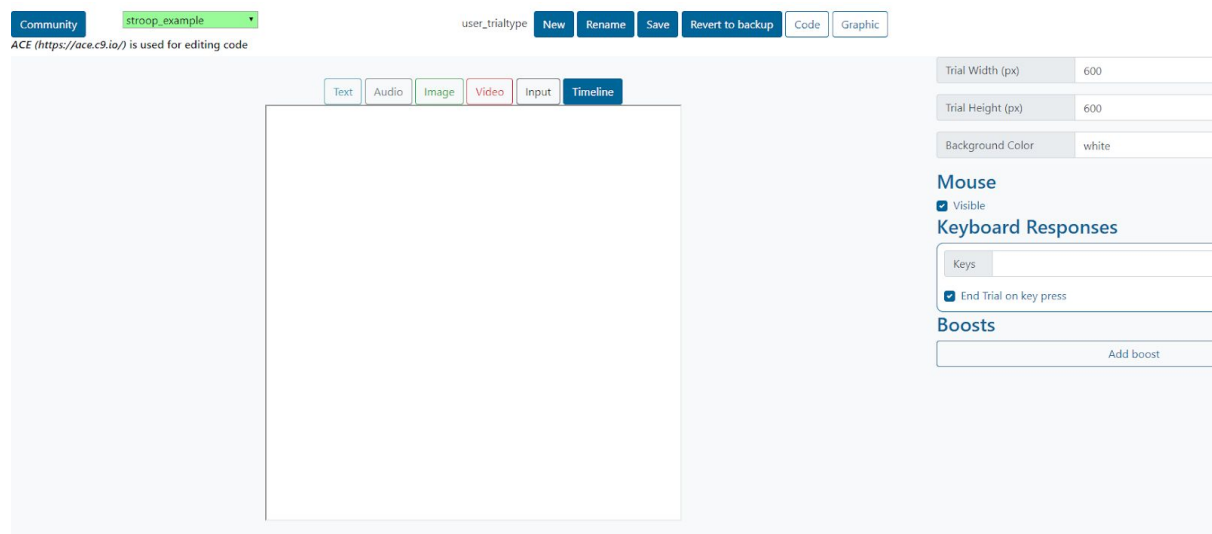
Trial types are types of trials the participant will experience. To create a new trialtype just click on the “New” button towards the bottom of the page:



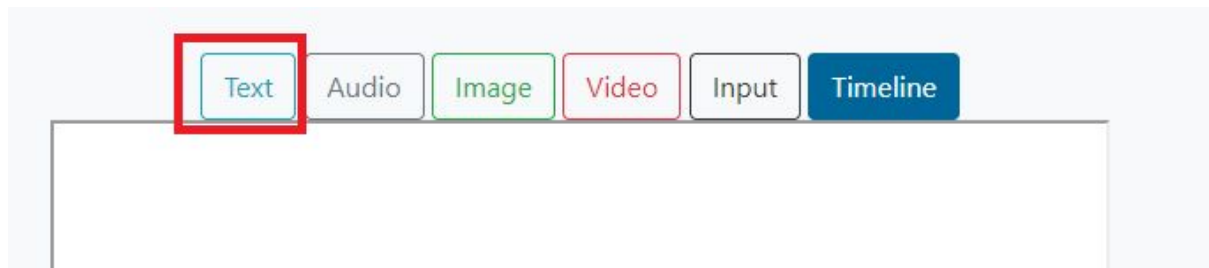
And then you’ll be asked for a name and whether you want to create the trialtype using code or the graphic editor. Let’s start by talking through how to use the graphic editor. So let’s click on “Using Graphics”:



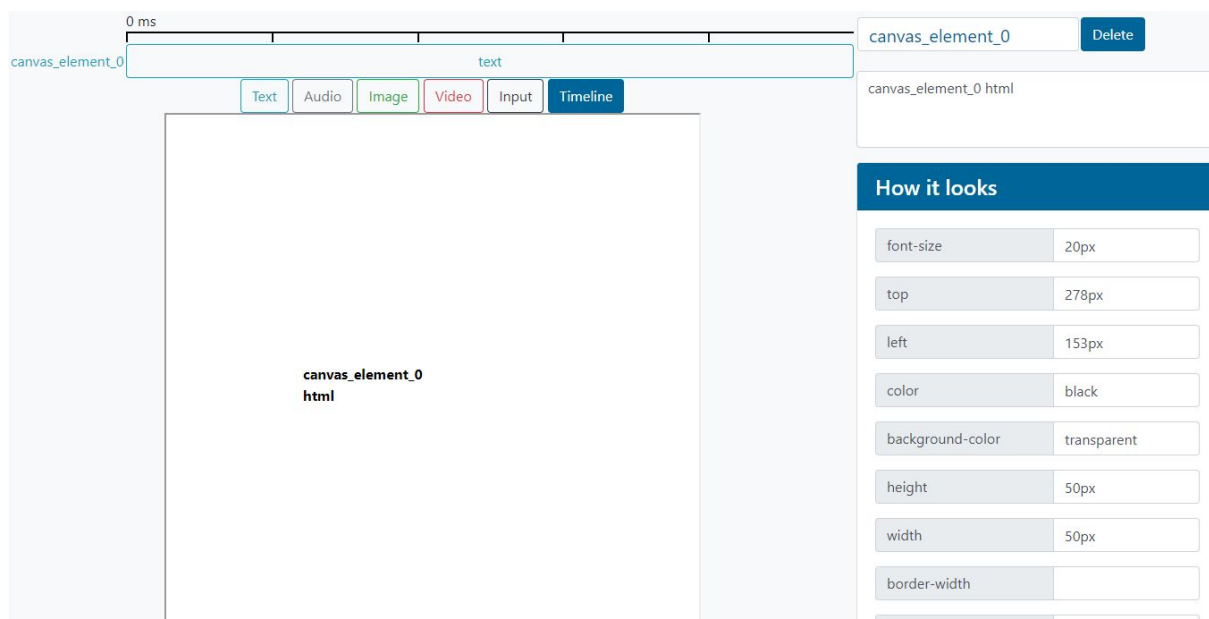
You may have noticed that I’ve given exactly the same name to the experiment and the trialtype “stroop_example”. I find this helpful as they can never get confused with each other. Once you’ve created the new trialtype you should see the the graphic editor:



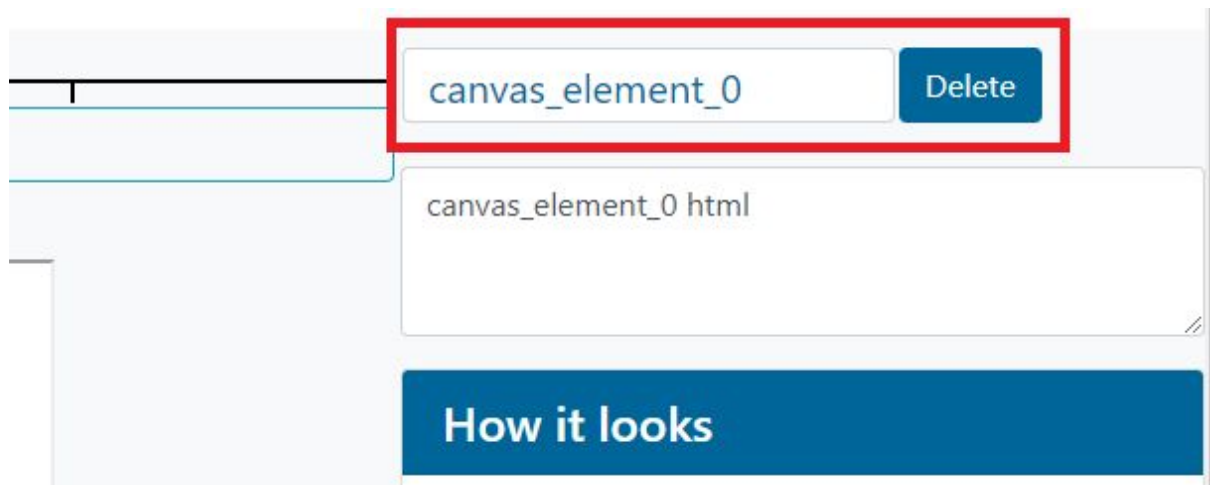
For a simple stroop experiment, we'll start by adding some text. Just click on the **Text** button:



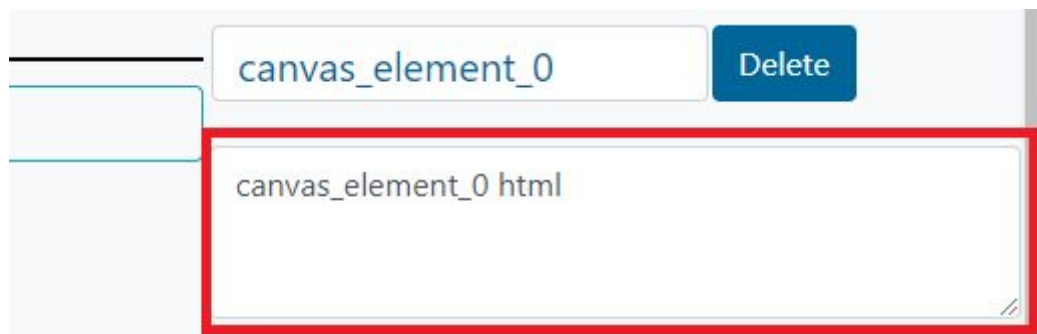
And then click where you want it. I'm going to try to put this pretty close to the center of the page:



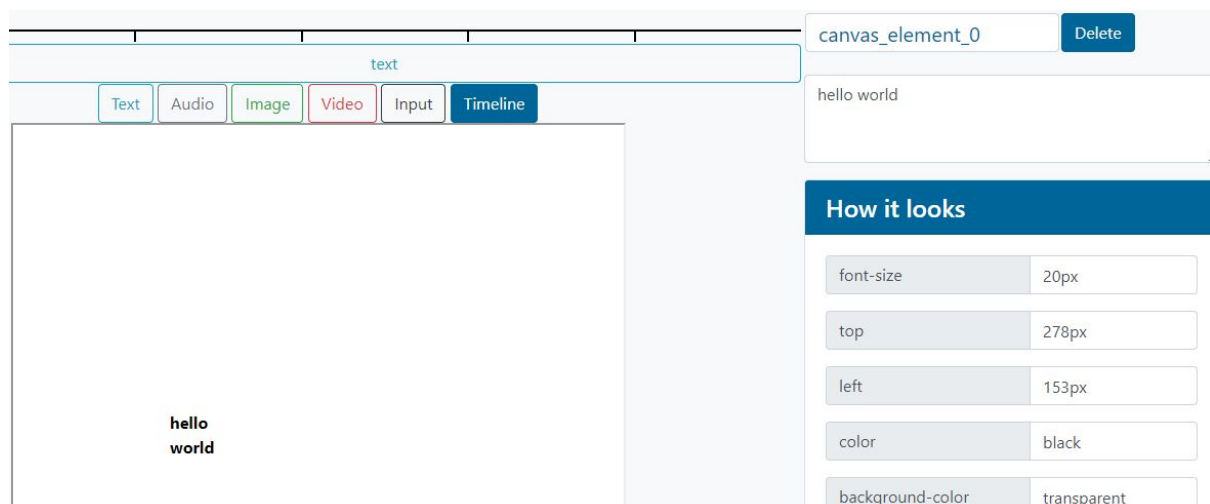
Let's unpack what we now have. In the top right corner we have the element_id "canvas_element_0":



This can be useful for making the experiment interactive, but we don't need to worry about that for now. Just below that we have the actual text in this element:



As it stands, every time you use the trial "stroop_example" you will get the text "canvas_element_0 html" - which is not particularly helpful. Let's change that to just say "hello world" to see exactly how this works:



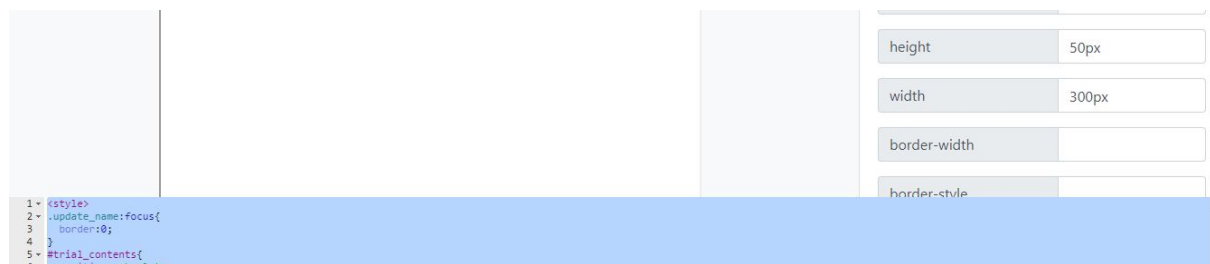
So you could theoretically create a separate trial with a different word or series of words for each trial. However, that would be very time consuming, so we're going to embed a **variable**

instead. **Variables** change for each trial depending on what you have in your procedure and stimuli spreadsheets. You just write the column title in the `{{}}` brackets. For a stroop task, we'll make use of the fact that **stimuli** sheets already have a **cue** column. So let's just refer to that column (we'll edit the **stimuli** sheet later). We'll replace "hello world" with `{{cue}}`.

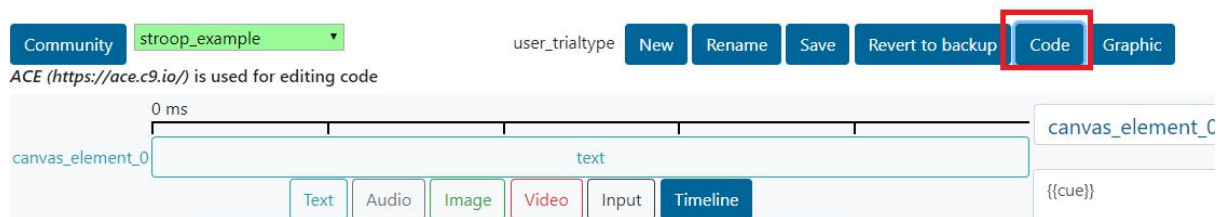
How it looks

Play around with the "How it Looks" panel to see the different sizes and colors you can have.

You might find that the **code** editor is in the way of the lower options:



You can hide the **code** editor by clicking on the "Code" button at the above the **graphic** editor:



You might need to click it a couple of times before the **code** editor disappears. You can click it again to make it come back. This can be a nice way to learn code - as you edit the experiment using the **graphic** editor, the code is written for the **trial type**. But let's get back to making our stroop task.

Being a stroop task, **color** is something that will also have to be a variable that changes from trial to trial. So I'm going to replace "black" with `{{letter_color}}` which is a column I'm going to need in my stimuli sheet later. I'm also going to make the font-size 30px; and the width 300px:

{{cue}}

How it looks	
font-size	30px
top	278px
left	153px
color	{{letter_color}}
background-color	transparent
height	50px
width	300px

It's important to have an appropriate width; if your width is too small your text will run over multiple lines. You might have noticed this in the earlier example with "hello world".

Timeline

At the top of the graphic editor, we get a sense of what happens when. In our case, let's:

- Add a fixation cross at the start for 500ms
- Add a 500ms gap following this by changing the start time of the word to 1000ms

To add a fixation cross, we just add another text element as described above. I'm going to make the **font-size** 100px and put it 250px from the top and left. This should put it in the center of a 600 x 600 pixel trial. To make it only be present for 500ms I'm going to scroll down to the **interactivity** panel, and change the **hide(ms)** value to 500ms:

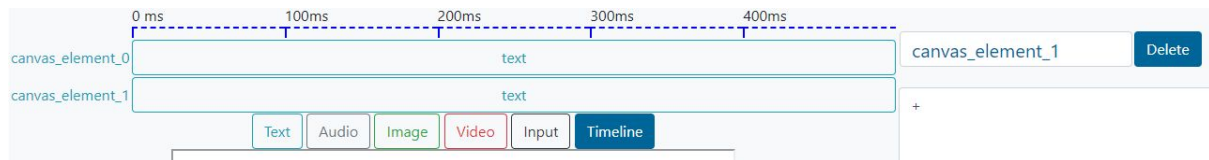
z-index	0
---------	---

Interactivity

show (ms)	0
hide (ms)	500
name	

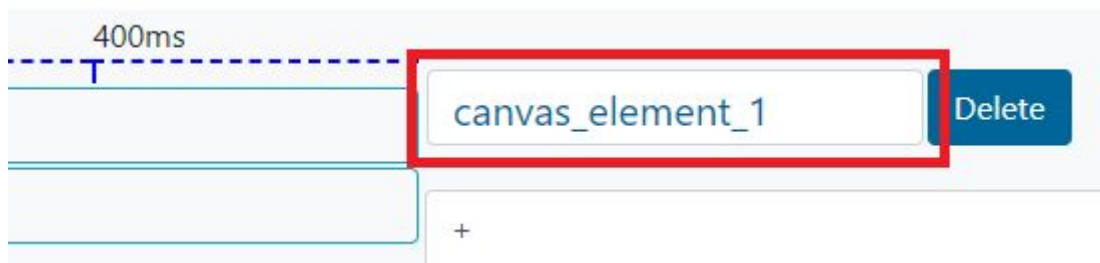
(Note that you can click on **how it looks** or **interactivity** headers to hide and show these interfaces).

Now that we've changed the **hide** time for the fixation, our timeline at the top looks like this:

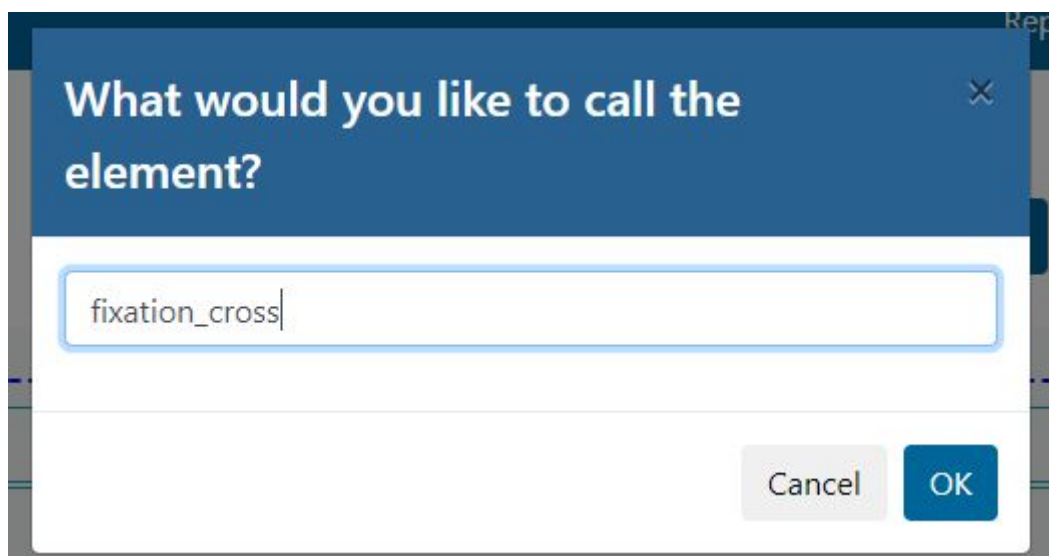


The **dotted blue line** means that at least one of the elements has no stop or hide time (i.e. will display until the end of the trial). Pay attention to how the timeline changes as we change the show and hide times to the stroop word.

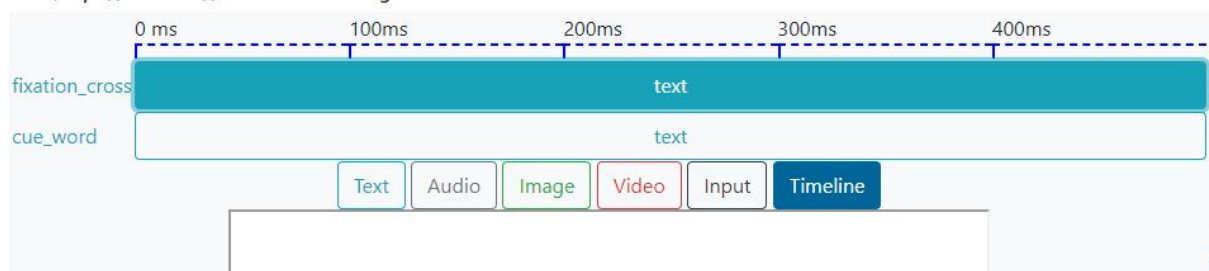
But first, let's change the element names so that the timeline makes more sense. Just click on the element name:



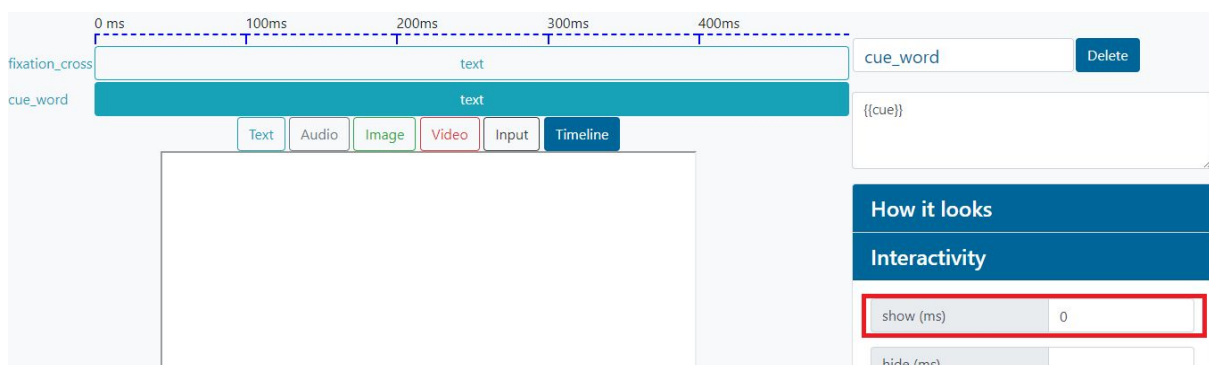
And you'll be asked for a new element id. I'm going to change the fixation cross element's name to "fixation_cross":



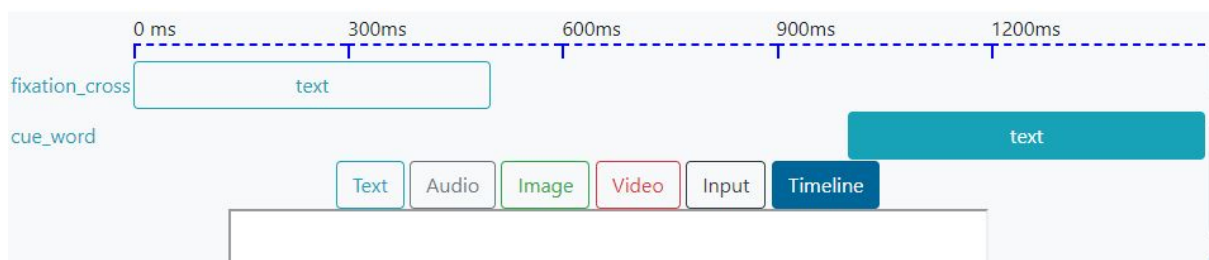
I'm also going to change the stroop word ("{{cue}}") element id to "cue_word". To switch between elements, either click on them in the timeline, or click on them on the **canvas**. The timeline is a bit easier to read now:



The next step is to have a 500ms gap between the fixation cross end and the start of the cue word. So let's just **show** the cue word at 1000(ms). This option is just above the **hide** input we just updated:



Let's change that to 1000 and see what it does to our timeline:



That seems sensible enough. Now we need to actually capture the participant's response!

Capturing response with keyboard

For stroop tasks, you typically need to capture a participant pressing on a key that represents a color. For simplicity's sake, I'm going to accept keyboard responses of "a" and "b". To be able to set the valid keyboard responses click on the canvas anywhere that there isn't an element. You should see the following:

I'm just going to add the letters "g" for green and "b" for blue to **keys**. Notice also the **End trial on key press** checkbox. This is what it says - if you have this ticked then as soon as the participant presses any of the keys you put in the **keys** input, the trial will end.

Capturing the response with the mouse

This is less relevant for a stroop task, but if you want to see what the participant clicked on in the trial, then select that element, go to the **interactivity pane** and change the **name** and **value**. The **name** will be related to the column header in your data sheet. So I'm going to make the name "clicked_element". The **value** will be what value you will put in that column if the participant clicks on that element. In this case I'm going to change the **value** to "fixation". I'm also going to tick the **End trial on click** option. If you do the same you should see the following:

How it looks

Interactivity

show (ms)	0
hide (ms)	500
name	clicked_element
value	fixation

☒ End trial on click?

Do the same for the cue_word element, but use the **value** “cue”.

Changing the canvas size

The canvas, i.e. the space that stores your trialtype, will by default be 600x600 pixels. If you want to change that, just click anywhere on the canvas where there isn't an element, and change the **trial width (px)** and **trial height (px)** settings.

900ms

1200ms

text

Input

Timeline

Trial Width (px)	600
Trial Height (px)	600
Background Color	white

Mouse

☒ Visible

Now we're ready to use this trialtype in our experiment.

Procedure sheet

The procedure sheet describes what order things will happen in your experiment. By default the first trials/events will be at the start of your spreadsheet, and the last will be at the bottom

of your spreadsheet. For example, in the procedure sheet below:

Procedure

procedure_1.csv ▾

New Sheet

Rename Sheet

1	item	trial type	max time	text	shuffle 1	
2	2	instruct	user	This is the start	off	
3	3	instruct	user	This is the end	off	
4						

The participant would be presented with an instruction of “This is the start”. When they moved onto the next trial they would get an instruction of “This is the end”.

Shuffle 1

You may want to randomise the order of some of the trials. If you want to do this then you just put the same word into all the rows you want to be randomised together. For example, in the following procedure I want to randomise the order of the middle three instructions:

Procedure

procedure_1.csv ▾

New Sheet

Rename Sheet

1	item	trial type	max time	text	shuffle 1	
2	2	instruct	user	This is the start	off	
3	3	instruct		Random 1	instruct_shuffle	
4	4	instruct		Random 2	instruct_shuffle	
5	5	instruct		Random 3	instruct_shuffle	
6	6	instruct	user	This is the end	off	
7						

You can see above that I set the value in the “shuffle 1” column to “instruct_shuffle” for rows 3,4,5. Each participant will start with the instruction “This is the start”, and end with “This is the end”, but the order of the middle instructions will be randomised for them. We’ll replace these later with the “stroop_example” **trial type** we’ll create later.

Trial type

The above examples are of the “instruct” **trial type** - which is just a simple way of giving the participants instructions that you write in the “text” column in your procedure sheet. You can create your own **trial types** using the trial type editor at the bottom of the interface.

Alternately, you can copy existing trial types that other users have made from <https://collectalk.com/>.

However, if you followed the above instructions, we now have a “stroop_example” **trial type** we can use. So let’s replace “instruct” with “stroop example”:

Procedure

procedure_1.csv ▾	New Sheet	Rename Sheet				
1	item	trial type	max time	text	Shuffle 1	
2	2	instruct	user	This is the start	off	
3	3	stroop_example	user	Random 1	instruct_shuffle	
4	4	stroop_example	user	Random 2	instruct_shuffle	
5	5	stroop_example	user	Random 3	instruct_shuffle	
6	6	instruct	user	This is the end	off	
7						

The text “Random 1” etc. won’t actually do anything in this case, so feel free to delete it.

Item

In the above example, you might have noticed the “item” column has numbers 2,3,4,5,6 in it. This is because your procedure sheet can refer to stimuli you describe in your **stimuli** sheet. These numbers refer to **rows** 2,3,4,5,6 in the **stimuli**. We start at 2 because when you edit the stimuli sheet here (or in Excel) the first stimuli will be on row 2. We’ll change what’s in the **stimuli** sheet later to refer to **variables** needed for a stroop task.

Max time

This is the maximum time in milliseconds from the start of the trial. You might find that you want your timings to be dependent on events within the trial. In which case you’ll want to use the **code** editor for **trial types**, or perhaps a **boost** that you apply to a **trial type** that controls timings in this way.

Text

This is only relevant for trial types that include a “{{text}}” **variable** in their code. The instruct **trial type** does, so it’s an easy way to add instructions. In the **procedure** spreadsheet.

Stimuli sheet

This is where you define the stimuli or variables for each trial. In our stroop task, we need to update the “cue” column to reflect the words used in a stroop task, and add a column “letter_color” which defines the colors of the word. So let’s just have 4 trials for each combination of blue and green:

Stimuli

stimuli_1.csv ▾					New Sheet	Rename Sheet	List Stimuli
1	cue	answer	letter_color				
2	blue	Apple	blue				
3	green	Banana	green				
4	blue		green				
5	green		blue				
6							

You may notice there’s a redundant “answer” column - that’s fine, it’ll just be ignored.

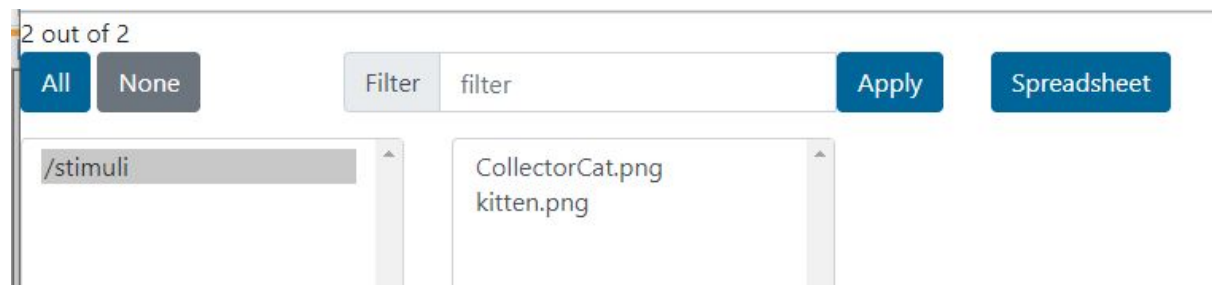
List Stimuli button

If you are running an experiment with pictures, videos and/or audio files then there’s an easy way to generate a list of all the stimuli if you’ve uploaded them to the **stimuli** folder on dropbox. Just click on the **list stimuli** button below the **Stimuli** title:

Stimuli



You should get an interface like this:



It might take a minute or two, depending on how many stimuli files you have.

Spreadsheet button - If you click on the “Spreadsheet” button, you should be able to download a list of the files (and **very importantly** their dropbox locations). This can save time massively if you have a lot of stimuli.

Filter input - if you have lots of stimuli, just type in letters for what you want to keep. For example, if I only wanted “CollectorCat.png”, I could type in “Cat” (**This IS case sensitive**) and then click **apply**. I would then see:

2 out of 2

AllNone

FilterCat

Apply

Spreadsheet

/stimuli

CollectorCat.png

Importantly, if I clicked on the **spreadsheet** button now I will only get information for the filtered list.

Helper sidebar

On the right hand side there is a “helper” sidebar. You’ll notice that the advice it gives you changes depending on what you’re focusing on. For example, when I’m looking at the “participant_id” column in the **conditions** sheet I get the following help:

participant_id	end_message	start_message	buffer	
random			5	

To see the whole cell when editing the sheets, either press F2 or double click on the cell

participant_id

Participants give their ID

By default, participants will give a **participant id** (e.g. a Prolific Academic or MTurk ID) at the start of the experiment. Change this setting to either **random** (recommended) to generate a unique and randomised id for the user, or **off** (not recommended) for the participant to have the id 'notUnique'. The participant_id influences the name of their data file. If the participant does not put in an ID, they will skip the **start_message**.

General tips

Downloading and Uploading experiments

On either side of the experiment name, there's an up arrow for uploading an experiment and a down arrow for downloading the selected experiment:



This can be an easy way to duplicate an experiment. Just download it, and then upload it and give it a different name!

If your experiment included trialtypes that were written using the **Code** editor (as opposed to the **Graphic** editor), Collector should ask you if you want to upload those trialtypes also. Similarly, any Surveys that were included in the original experiment should be downloaded and uploaded accordingly.

Running your study

Save!

Make sure you've saved! You usually can just press CTRL-S to save. The best way to save all your progress is click on the save button in the top left of the **studies** interface:



Conditions

1	name	notes	stimuli	procedure	fullscreen	welcome
2	condition_1	You can put mo	stimuli_1.csv	procedure_1.cs	off	
3						

Click the “run” button

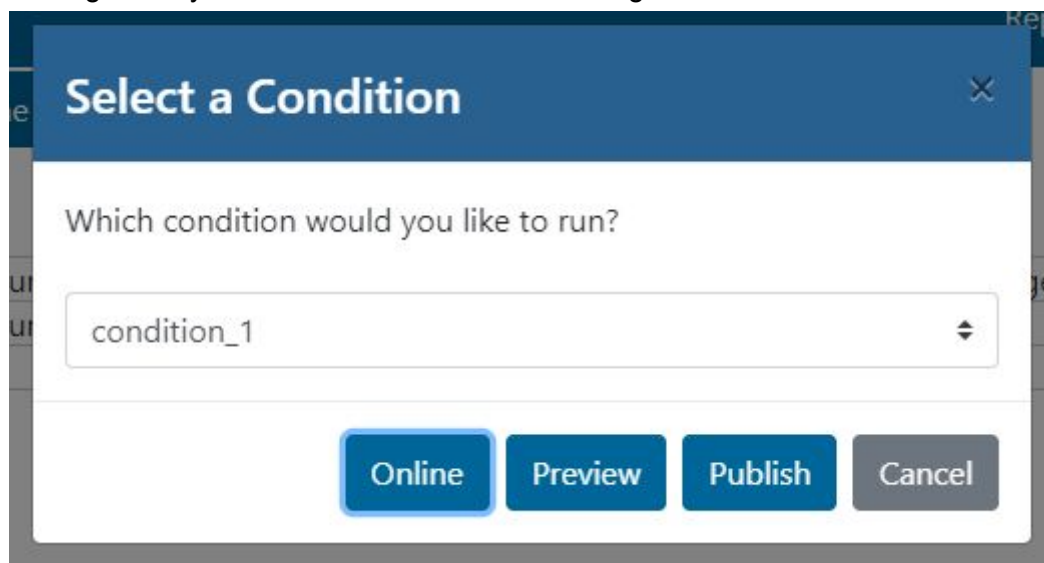
When you click on the **run** button:



Conditions

1	name	notes	stimuli	procedure	fullscreen	welcome	participant_i
2	condition_1	You can put me	stimuli_1.csv	procedure_1.cs	off		on
3							

You'll generally see an interface like the following:



Online - will take you to where you would want to send participants. But see **publish** before using this

Preview - this should run the task without saving data


Publish - this will allow you to publish your task onto a server that will handle either saving

of your data. You can also get to the **publish** interface by clicking on the **C** icon in the top right of the screen.

Publishing your task

After clicking on **run** → **publish** or just the **C** icon on the top right of the screen, you should see something like:

Register

Server	Register User	Register Experiment
ocollector.org	Create Account	Publish
	Delete Account	Unpublish
	Forgot password	
open-collector.org	Create Account	Publish
	Delete Account	Unpublish
	Forgot password	
Add server		
<div>email</div> <div>password</div> <div> <input type="checkbox"/> I'm not a robot <div>  <div>reCAPTCHA</div> <div>Privacy • Terms</div> </div> </div>		

If you are a member of one of the following universities, you are automatically able to have your data e-mailed to you:

University of Reading (i.e. have an @reading.ac.uk e-mail address)
 University of Reading Malaysia (i.e. have an @reading.edu.my e-mail address)
 University of California, Los Angeles (i.e. have an @g.ucla.edu e-mail address)
 University of DeMontford (i.e. have an @dmu.ac.uk e-mail address)

To get your data e-mailed to you and backed up, put in your e-mail and password and then click on **create account** for the server you want to register with. Once you have successfully registered with that account, you can then **Publish** your experiments to that server. Once you've successfully published your experiment, you're ready to start data collection!

Note, if you publish your experiment with multiple servers Collector will use some of the servers as back-ups in case the first server fails.

Once you've reached here with the stroop task (and have successfully registered your experiment), close this interface and click on the **run** button as described above. You can now click on the **online** button to see how your task looks (and allow us to look at your data in a bit).

Decrypting your data (and understanding it)

I'm going to include some clarifications about actually running your task in this section, so feel free to skip to [decrypting your data within the "Data" tab](#) if you already have been e-mailed or downloaded data.

If you're working through the stroop task exercise, you'll see something like the following when you run your experiment:

Collector

It's very important to read the following before starting!

If you complete multiple Collector experiments at the same time, your completion codes may be messed up. Please do not do this!

If you participate in this experiment, your progress in it will be stored on your local machine to avoid you losing your progress if the window or tab closes or freezes. This data will be cleared from your computer once you have completed the task. **However, if you do not want this website to store your progress on your computer, DO NOT PROCEED.**

If the experiment freezes, try pressing **CTRL-S** to save your data so far.

I'm just going to run the experiment as "example_p".

Assuming everything went okay, you'll get through the task until you reach:

Please wait while we confirm that all your data has been saved

After a while, the data should be saved, and you'll see:

Thank you for participating. If you'd like to download your raw data [click here](#)

This is totally optional, but please select the country you are completing this from and then click on "Submit"

Submit



It seems that the lag between completing an experiment and the researcher's e-mail acknowledging it vary quite a lot. However, if your participant sees the second screen with the map the data is on the way. If they get stuck on:

Please wait while we confirm that all your data has been saved

...for 5 or more minutes then something has gone wrong. You may need to ask them to press CTRL-S to save the data so that they can email it to you directly if this occurs.

Decrypting your data within the "Data" tab

Once your data has arrived in your e-mail, download it, and then go to the **data** tab at the top of the Collector interface:



Once there you can click on the "Decrypt files" button, select your file in your downloads folder (or wherever you saved it).

You'll need the password you created originally to decrypt it. Once you've typed it in you should be presented with your file!

Understanding your data

If you've been working through the stroop example, your data should look something like this:

	P	Q	R	S	T	U
	cue	answer	letter_color	keyboard_response	clicked_element	
	blue	Apple	blue			
	blue		green	g		
	green		blue	b		
	green	Banana	green	g		

You can see in column **s** “keyboard_response” that I pressed g then b then g. This seems to be correct as “b” represents me thinking the letters were blue, and “g” represents me thinking the letters were green (which matches column **R** “letter_color”). Had I clicked on the fixation cross or the word, then column **t** “clicked_element” would have included the words “fixation” or “cue”.

Surveys

To create and edit surveys, go to the Survey tab at the top:



You cannot create a survey from scratch, you need to select a previously existing survey to base the new survey on. I'm going to use demographics just because it has a lot of different types of questions. Just select “demographics.csv” and then click on **new survey**:



The **helper bar** on the right should tell you a lot about each column depending on where you click in the survey. So I'm just going to highlight the fact that after each change to the survey you can preview you changes by clicking on the **preview** button:

New Survey
exemplar_demographics.csv
Save
Rename
Delete
Spreadsheet
Preview

Gender
Male
Female
Other

Age

Education (which level are you currently in, or is the highest you've completed)
-- no option selected --

Do you speak English fluently?
Yes
No

At what age did you start learning English? (write "0" if from birth)

What is your country of residence?

What is your preferred hand?
Left
Right
Both

If you would like to please state your ethnicity

If you want to go back to editing, just click on the Spreadsheet button to the left of the preview button:

New Survey
exemplar_demographics.csv
Save
Rename
Delete
Spreadsheet
Preview

FAQ/General questions

Can I use Collector if I am not a programmer?

YES! This program was designed to allow researchers to work with a format they're very familiar with (spreadsheets) to create interactive experiments. As you will see in the tutorial Collector is almost a completely programming free solution. Of course there are times when being able to code will make your life easier but we have tried to minimize user coding wherever possible.

What language is Collector written in?

Collector is written mostly in javascript/Jquery. Formatting of the presentations is controlled mostly by HTML5 and CSS2.

Do I need to ask anyone if I want to use Collector?

No. Collector is distributed under the GNU GPLv3 license (full text can be found in the **Web/Admin** folder). You are free to use, modify, and distribute this program as you wish. If you distribute a modified version of this program you are required to make your modified version available to the public under the same GNU GPLv3 license of the original program. If you are publishing papers that have used Collector experiments we ask that you acknowledge its use somewhere. This isn't a requirement of using this software but we'd like to get the word out to as many people as possible about its availability.

Collector will be citeable in a prepublication journal later this year.

If you build trial types that you think others would find useful please go to <https://www.collectalk.com> to share.

Troubleshooting

My changes aren't being reflected in the experiment. Why?

1. Save all your files before running the experiment. 90% of the time when you are having this problem it is because you forgot to save the changes you've made. Seems silly but I run into this all the time.
2. Clear your cache. There are extensions for chrome (<http://goo.gl/r961i>) and firefox (<http://goo.gl/r3Zky>) that make this process very quick
3. Are you sure you're editing the right experiment/folder? (this is for if you edit the experiment files directly, not within the Collector.py interface)

Further notes

What happens with my password for encryption and decryption

Upon creating a Collector folder in your dropbox account, a public and private key are generated using tweet nacl library by dchest available at <https://cdnjs.cloudflare.com/ajax/libs/tweetnacl/1.0.1/nacl.min.js> . The public key is used for encrypting data, the private key is used for decrypting data. Your private key is encrypted using Crypto-Js available at <https://cdnjs.cloudflare.com/ajax/libs/crypto-js/3.1.9-1/crypto-js.min.js>

This means that your private key can only be used when you type in your password, and thus the data can only be decrypted by someone who knows your password.

What happens with my password for Collector

When creating an account on Collector you are asked for a password. This password is hashed using a randomly generated salt and pepper at the start and end of your password using PHP:

```
$salt = create_random_code(20);
$pepper = create_random_code(20);
$prehashed_code = create_random_code(20);
$hashed_password = password_hash($salt.$user_password.$pepper,
PASSWORD_BCRYPT);
$hashed_code = password_hash($prehashed_code, PASSWORD_BCRYPT);
```

This means that we don't store your password in bare form on the server. Next time you log in your password is compared to the hashed version, and if they match then we accept it is you, and keep you logged in for 30 minutes. If you don't do anything for 30 minutes then we'll ask you to log in again.

In case you're curious - here's how we generate random codes:

```
function create_random_code($length){
    $characters =
'0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ';
    $charactersLength = strlen($characters);
    $new_code= '';
    for ($i = 0; $i < $length; $i++) {
        $new_code .= $characters[rand(0, $charactersLength - 1)];
    }
    return $new_code;
}
```

Original contributors

Collector was initially created by Mikey Garcia as software that should allow researchers to conduct online studies for free. The below is taken from the [original repository](#) he created:

Tyson Kerr

"Many of the best ideas/solutions in Collector have come from the mind of Tyson. Tyson's contributions are so wide that it is hard to think of a piece of the code he hasn't been involved with at this point. Despite his broad contributions, I think Tyson would agree with

me that his real baby is the getdata functionality in /Data/. Every time you use those slick menus to check participant completion, exclude flagged users, or download all your precious data into one clean sheet you have Tyson to thank."

Victor Sungkhasettee

Figured out how to implement the Audio trial type

Adam Blake

"Completely reorganized the collector.js code to be object oriented. Is responsible for the current look of Collector because he redid nearly all of the CSS to make it much prettier than I initially could."

Nate Kornell

"Without Nate there would be no Collector. Many years ago Nate taught me how to use the tool he had created for himself, Generic, and that code inspired me to write Collector. Most of the core ideas and design decision at the heart of this project are either directly lifted from Nate's program or were based on adaptations of what he had created."