# Running online experiments

Session 1: jsPsych
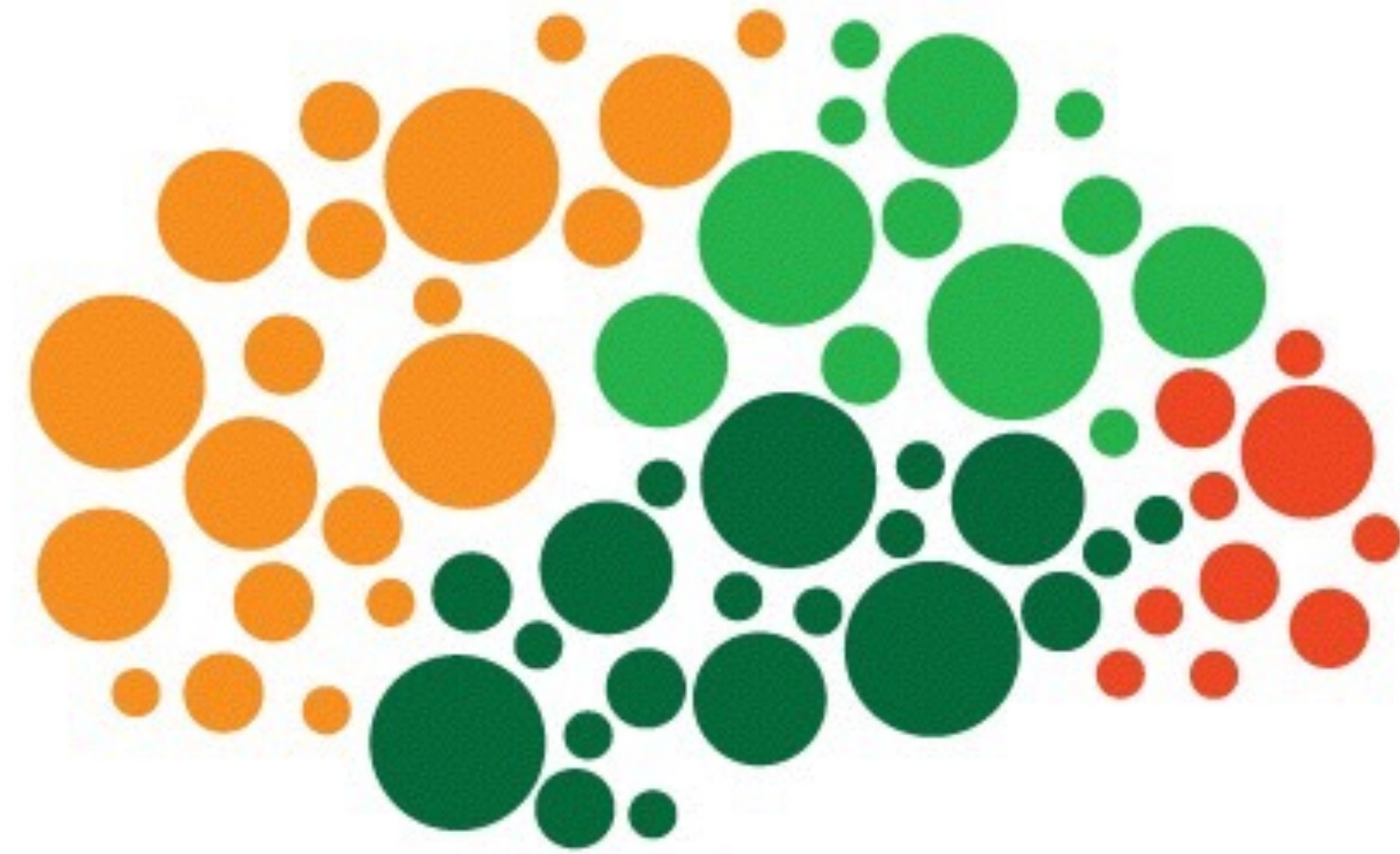
Justin Sulik justin.sulik@gmail.com

@justinsulik

# Outcomes: learn how

- jsPsych can make your experiments and data collection

  - customizable

  - open source

  - portable

- To set up an experimental timeline

- To add instructions and stimuli to the timeline

# Outcomes

- Learn where to begin:

  - What the basic elements of a web experiment are

  - What to google if (when) you get stuck

jspsych.org                                                    Josh de Leeuw

# JavaScript?

- It's the language that makes websites interactive

# Oh god do I have to learn another language?

- **No!**

# Oh god do I have to learn another language?

- **No!**

- You have to learn what some basic things are called

  - e.g., event handlers, console, selector

  - So that you can google them

- You have to learn to copy+paste from jsPsych.org, stackoverflow, etc.

- AND be patient, keep at it, problem solve, ask for help

# 3 basic elements of your experiment

- web page: html

- style sheet: css

- scripts that make it dynamic: js

# 3 basic elements of your experiment

- web page: html

- style sheet: css

- scripts that make it dynamic: js

- Before we get to jsPsych - go over the above 1 by 1

# html

- We're going to be working in the folder catsperiment

- Open basic.html in Atom (or text editor)

# html

- html uses tags <> to divide up the page
- <h1> big header </h1>
- <body> the content of the page </body>

# html

- Notice that it's just plonked everything on the left, starting from the top

- That's because we've given it some content, but not told it how to style our content

- There are ways to do this inside the html document, but the modern sensible way is to use CSS (Cascading Style Sheets)

# CSS

- The CSS file is a style sheet: it just tells the html how things should look

```
h1 {font-family: Helvetica; font-size: 90px;}
```

# CSS

- The CSS file is a style sheet: it just tells the html how things should look
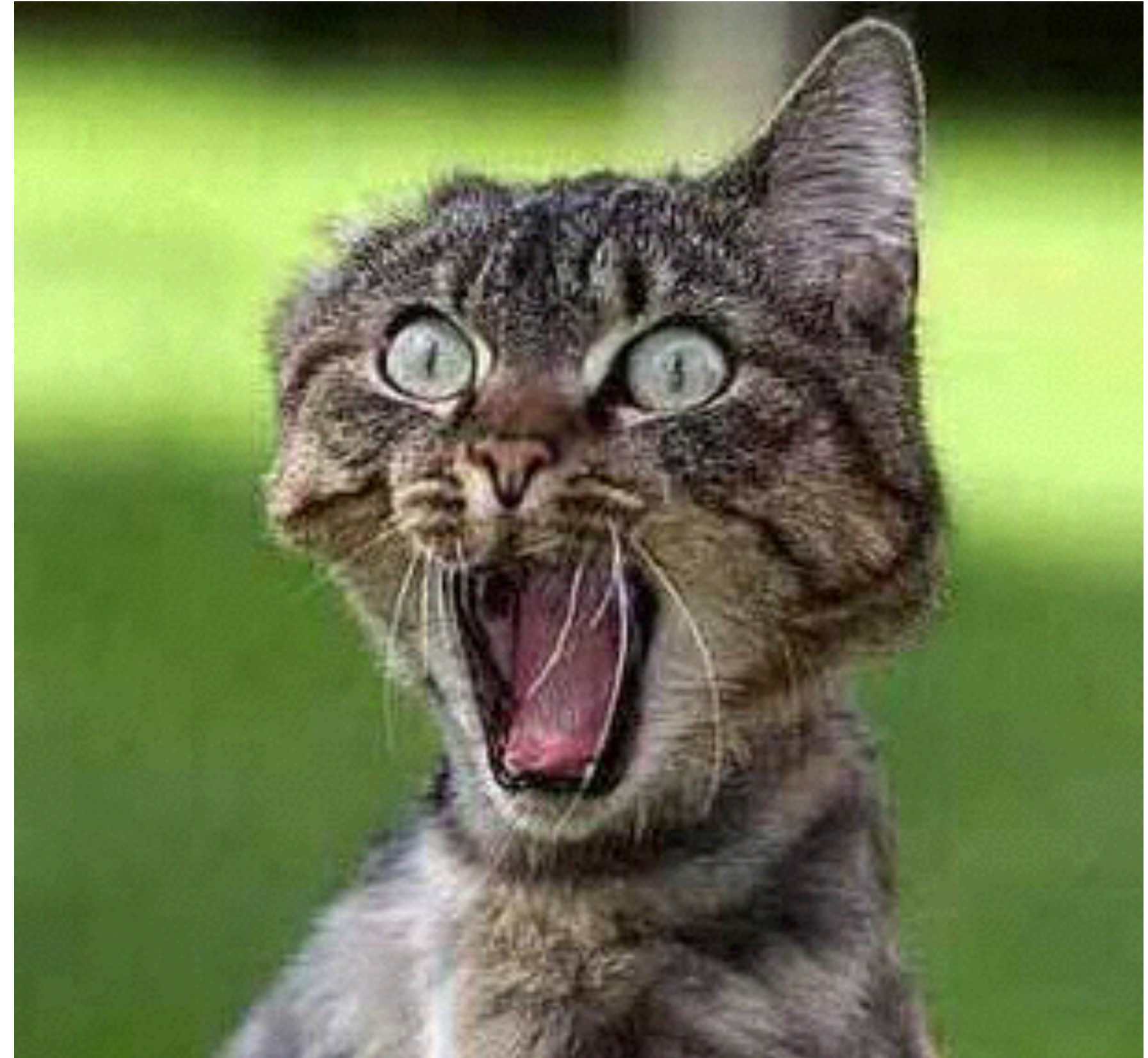```
h1 {
font-family: Helvetica;
font-size: 90px;
color: #acacac;
}
```

# CSS

- Why hasn't the header changed color?

- We need to tell the html where to look for its style sheet!

- Un-comment the <link> and reload the page

# CSS



- Still nothing? What's gone wrong? OMG PANIC IT'S NOT WORKING HTML IS STUPID

- (or just open the error console and see what's gone wrong)

# CSS

- "Failed to load resource: The requested URL was not found on this server."

- Why can't it find resource `basiccss`?

# CSS

- un-comment the p{} selector to change how all the <p> parts of the html are formatted
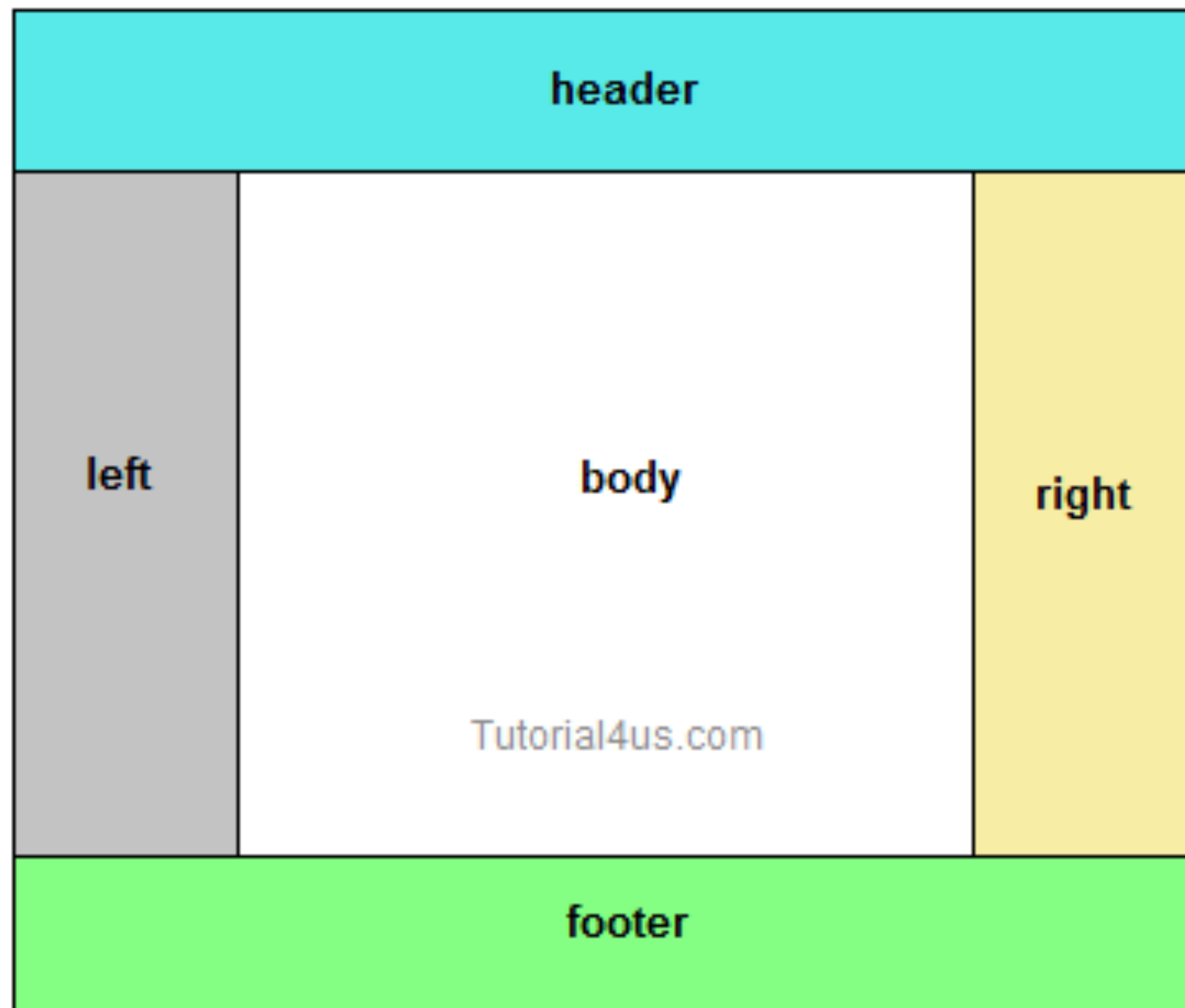
# CSS

- What if we want to format just **some** of the paragraphs?

- We can create custom classes by adding a `class` attribute to the html tag, e.g.

  `<p class="myDiv">`

# html+css

- We've changed how things look, but they're still just clumped at the top left

- To position your content, think of a web page as a series of nested boxes

| header | | |
| left | body | right |
| footer | | |

Tutorial4us.com

# html+css

- open basicWithDivs.html

- Look for these divs:

  - mainDisplay

  - left panel

  - right panel

# html+css

- It turns out left/right panel are inside mainDisplay

- The indentation shows this

- But it's probably easier to see on the page

- In basic.css, add a border to `.mainDiv`, `.left` and `.right`

  - e.g. `border: 1px solid red;`

# html+css

- Now our page has content+styling

- But it's static!

  - You can't interact with it

  - Everything it displays has been programmed in

- To make it dynamic, we need JS

  - People can interact with it

  - It will change the content/styling depending on what they do

# JS+jQuery

- JavaScript is what makes web pages interactive

- jQuery is a JavaScript library
  - Think of it as an extension of the language that simplifies certain tasks, rather than a whole extra language

- https://www.javascript.com

- http://try.jquery.com

- https://www.codecademy.com

# JS+jQuery

- We can include our scripts in our html

  - Inside a <script> tag

- Or we can out the JS in a separate file and just use the <script> tag to point the html to a script file with `src=""`

# JS+jQuery

- Open basicWithJS.html

- What scripts have been included in our html?

# JS+jQuery

- Notice that we have a local source for jQuery

- It's already in the catsperiment/scripts folder

- It can be downloaded from the jQuery website

- Alternatively, google hosts libraries so you needn't download anything

  - (https://developers.google.com/speed/libraries/)

# JS+jQuery

- Open the console to check that 2 things are being logged:

  - one from the html script itself

  - one from the basic.js script

# JS+jQuery

- A couple things about sytax:

- End commands with ';'

```
var x = 1;


function(args){doStuff;}
```

# JS+jQuery

- A couple things about syntax:

- End commands with ';'

```
var x = 1;


function(args){
doStuff;
}
```

# JS+jQuery

- $( selector ).method();

- $( 'p' ).css('color', 'red');
  - Notice - different syntax from css file

- $( '.myDiv' ).html('New text');

- in myScript, try change the content or style of one html element

# JS+jQuery

$( 'img' ).hover(doStuff1, doStuff2);

- doStuff1: function for when mouse enters img
- doStuff2: function for when mouse leaves img

# JS+jQuery

```
$( 'img' ).hover(
  doStuff1,
  doStuff2
);
```

# JS+jQuery

```
$( 'img' ).hover(

  function(){whatToDoOnEntering},

  function(){whatToDoOnLeaving}

);
```

# JS+jQuery

```
$( 'img' ).hover(

    function(){

    whatToDoOnEntering

    },

    function(){

    whatToDoOnLeaving

    }

);
```

# JS+jQuery

- Ok, so why is nothing happening when I mouse over the image?

- This is one of the times when (sadly) there are no error messages to help you

# JS+jQuery

- Learning moment: try be aware of the order in which things are built

- Look at the order in which the various 'Hi!' messages are logged

# JS+jQuery

- Currently, it runs basic.js as it's parsing the header

- At that point, it hasn't rendered the body yet!

- So there are no images yet

- So it has nothing to attach this event handler to

# JS+jQuery

- 2 solutions:
  - Change the order (move the basic.js <script> tag to the bottom of the html)
  - Put everything inside a function that gets run once the page has loaded

# JS+jQuery

- Try the simple way (moving the <script>)
- Then switch to file basic2.js which has the more complicated way

# JS+jQuery

`$( selector ).method(doStuff);`

# JS+jQuery

```
$( document ).ready(doStuff);
```

# JS+jQuery

```
$( document ).ready(function(){});
```

# JS+jQuery

```
$( document ).ready(

    function(){

});
```

# JS+jQuery

```
$( document ).ready(

    function(){

        [move our script here]

});
```

# JS+jQuery

- Ok, let's get some responsive text now

- Declare a variable hoverCounter

  `var hoverCounter = 0;`

- (just before your event handler)

- In the event handler, add 1 to `hoverCounter` every time the mouse enters the image (`+=1`)

# JS+jQuery

- Always work bit by bit, testing as you go

- (i.e. log `hoverCounter` to the console to check it's working before you try display it on the page)

# JS+jQuery

- Use a selector to change the content of `myDiv` to show how many times we've moused over the image

# JS+jQuery

- Make the image do something when you're hovering over it:

  - In the same function that increases hoverCount, use `$( this )` to select the current element (i.e. the image)

  - Try either:

  - give it a border `$( this ).css();` or

  - make it fade `$( this ).fadeTo('fast', 0.4);`

# Interim summary

- Ok, so currently we have 3 documents

  - html: our page with content

  - css: our style sheet

  - js: our JS/jQuery script to change things

- We could design the whole experiment based on this!

# Interim summary

- But we could also just use jsPsych

  - This is a JS library for running experiments

  - It has lots of plugins

    - These are basically neat little packages we can plug values into and it will sort out the rest

    - We just say we want a trial of type X with stimuli A and B, and it handles everything for us!

      - Display, responses, RT, etc.

# The basic structure of a jsPsych experiment

- html with the main experiment script

- css file with styling for your experiment

- jspsych.js with the jspych script

- one or more plugins for displaying particular kinds of stimuli (e.g. jspsych-survey-likert.js)

- a timeline

# jsPsych

- Open catsperiment.html
- Add some scripts in the header (follow comments)

# Main experiment script

- Uncomment the <script> underneath the body

- This sets up an experiment timeline

- This is a container where we're going to put all the pages that a participant will have to go through

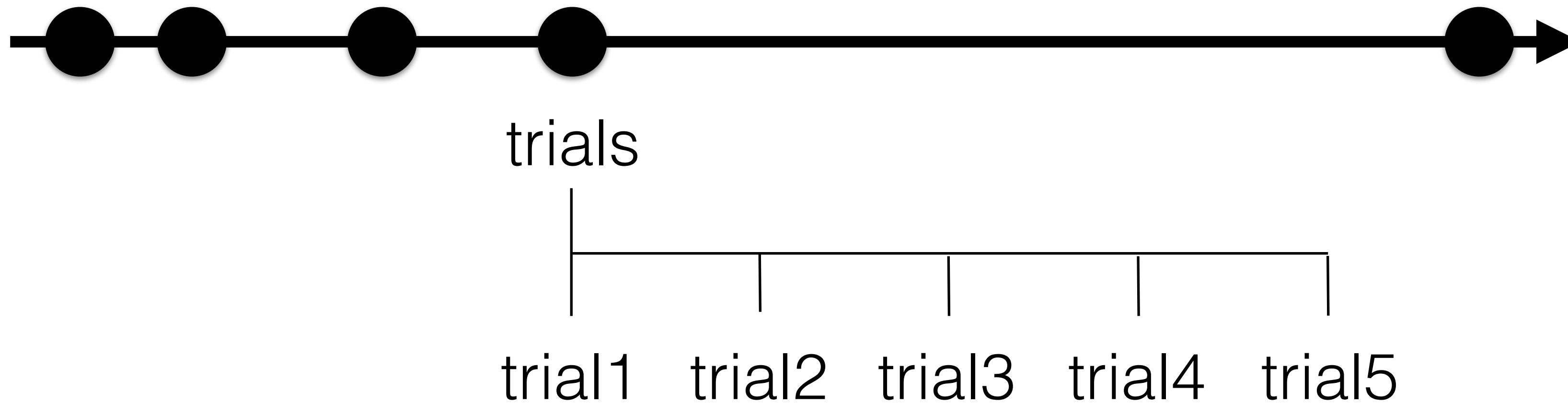jsPsych timeline

# jsPsych timeline



welcome

# jsPsych timeline



instructions

# jsPsych timeline



practice

# jsPsych timeline



trials

# jsPsych timeline

end message

# jsPsych timeline

trials

trial1 trial2 trial3 trial4 trial5

# jsPsych timeline

# jsPsych timeline

- The timeline is a list of lists:

```
[

[welcome],
[instruction1,instruction2,instruction3],
[practice],
[trial1,trial2,trial3,trial4,trial5,trial6,...
trialN]

]
```

# jsPsych timeline

- All we have to do is go through the things we need in our experiment, and add them to the timeline

- (`push` them to the `list`)

- Google how to do this!

# Instructions

- Here is where we meet our first jsPsych plugin: **jspsych-instructions.js**

- How do you use plugins?

- Look at the examples on jspsych.org

- Each plugin is designed for a specific kind of page (instructions, 2AFC, go/no-go, etc.)

- (It's not too hard to write your own plugin - e.g. for our cross modality experiment)

# Instructions

- Each object in the timeline is like a dictionary of values

```
var myName = {

attribute1: value1,

attribute2: value2,

};
```

# Instructions

- Uncomment the script at the bottom of the html file

- Why isn't it displaying anything?

- Find out what the problem is and fix it!

# Instructions

- Another problem?

SyntaxError: Unexpected identifier 'pages'. Expected '}' to end a object literal.

This one's a bit obscure. Compare carefully with my example a few slides ago - what's missing

# Instructions

- Another problem?

- We've created an object, but we haven't added it to our timeline

- (what was the word I used for adding to a list?)

# AAAAAGH, still nothing

# jsPsych timeline

- We've created a timeline

- We've added some experiment objects to it

- We still need to tell jsPsych to begin the experiment

- Go to jspsych.org and find a basic tutorial

- Look for whatever's needed to begin the experiment

  - (it may need some variable names changed)

# Lists of lists of lists

- If your list of instructions gets complicated, create it elsewhere and just pass the variable name to the trial object

# The images are all different sizes!

- How will we display them correctly?

# It just ends!

- google 'jspsych end message'

- let's provide them with a thank you

# Add feedback

- For now just logging to console - can add to display but a bit tricky for a short session

- Open feedback.js and follow instructions in comments

# Thinks we can't cover

- Conditional timelines

- Checking if correct

- Saving data

- Playing audio

- How to serve it online

- Check the docs! Search stackoverflow! Then ask me!

# Worth remembering

- Just because this **can** happen online doesn't mean it must

- You can use this setup for your in-lab experiments/fieldwork.