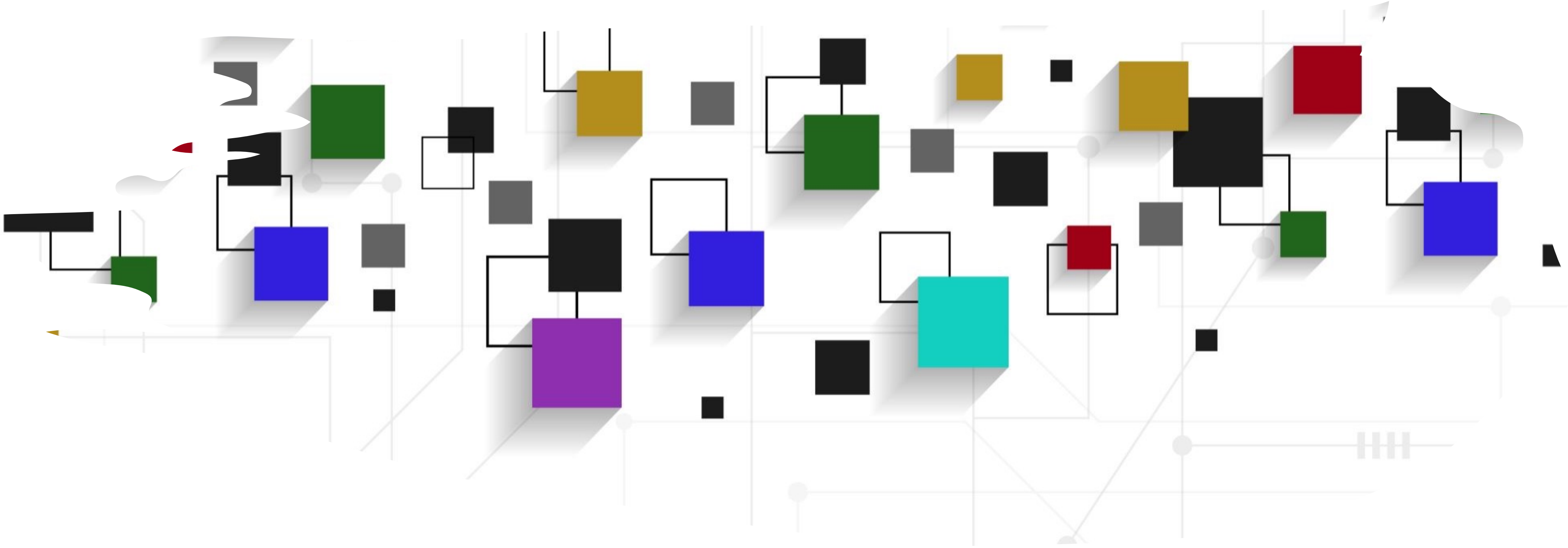


CogLab: jsPsych timeline

WEEK 5

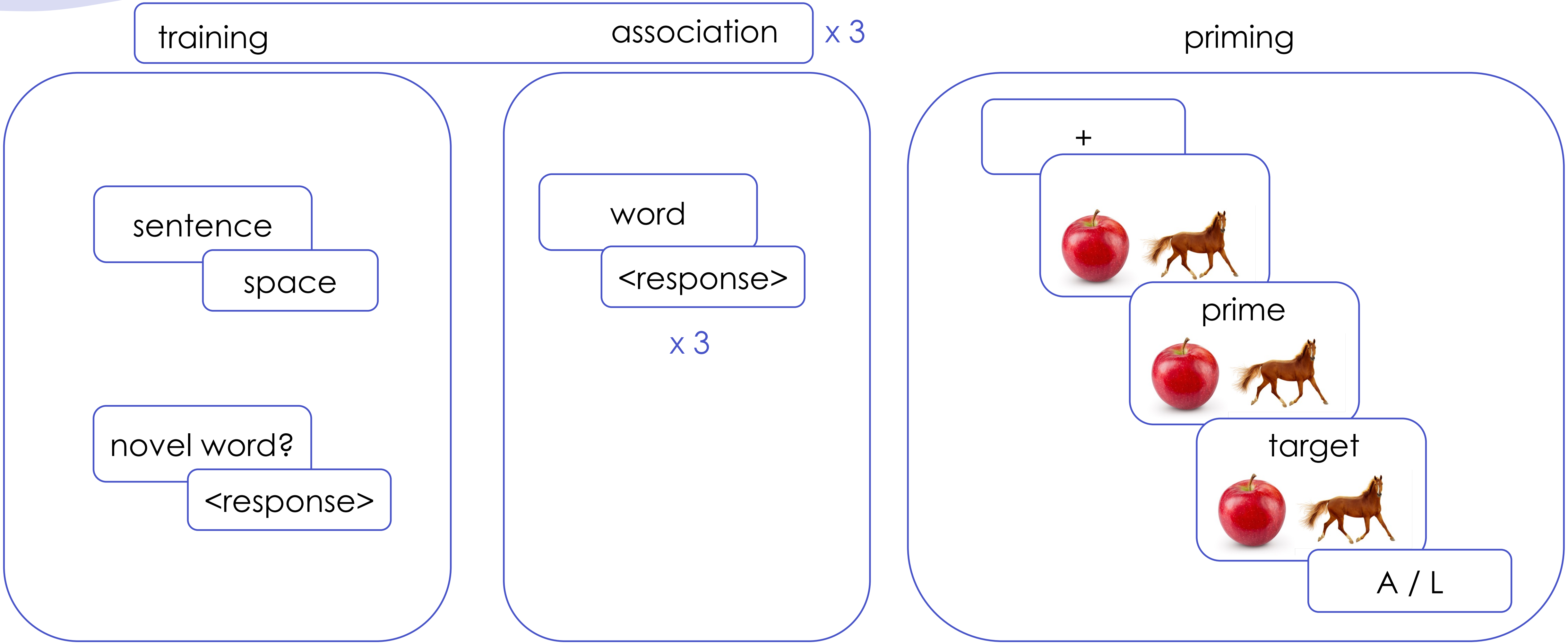
recap: Sep 21, 2023

- what we covered:
  - jsPsych 101
  - building a sequence of events/screens
- your to-dos were:
  - *prep:* https://www.jspsych.org/7.0/overview/timeline/
  - *try*: Week 4 quiz
  - *apply*: project milestone #3 (proposal)

going back to our experiment

- open Visual Studio Xcode and open the jsPsych experiment you created last week
- also open the index.html file in your browser to remind yourself of what we did!

experiment recap



training

association

x 3

priming

sentence

space

novel word?

<response>

word

<response>

x 3

+

prime

target

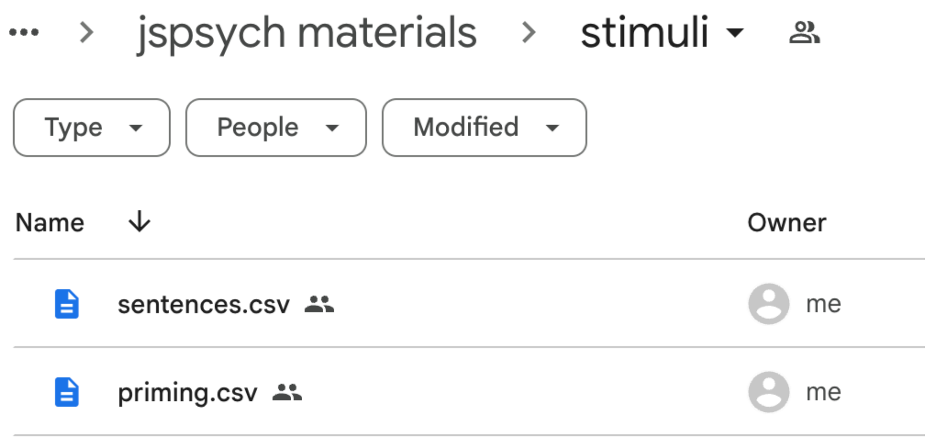
A / L

today’s agenda

- importing stimuli to jsPsych
- repeating sequence of events for different items

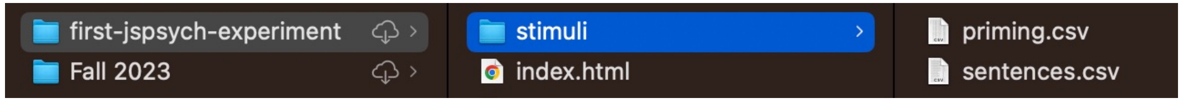
building stimuli files

- what do we need for this experiment?
  - a list of sentences
  - a list of prime-target pairs
  - images
- download the stimuli folder into the same directory as your index.html file



… > jspsych materials > stimuli

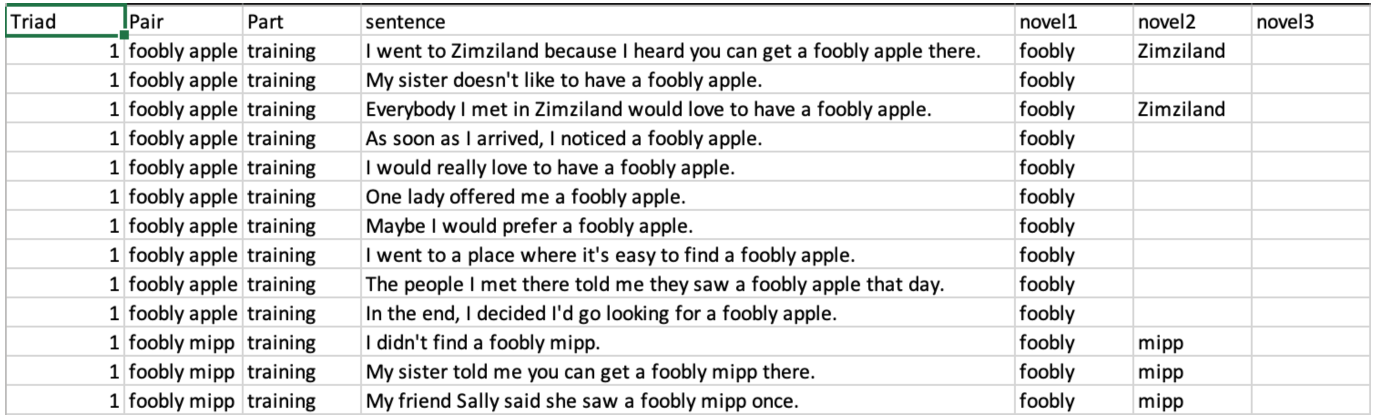
Name	Owner
sentences.csv	me
priming.csv	me



```
first-jspsych-experiment    stimuli       priming.csv
Fall 2023                   index.html    sentences.csv
```

sentences.csv

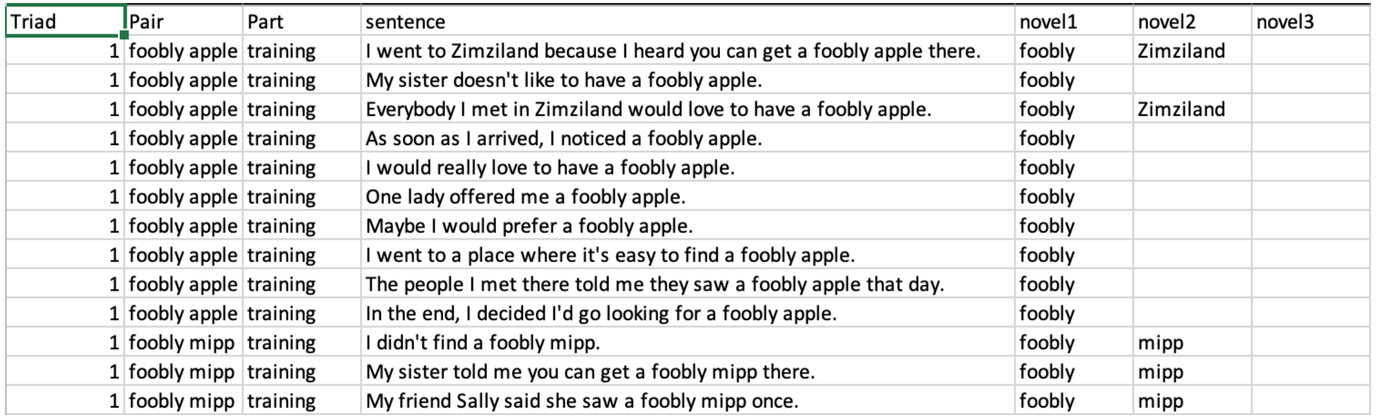
- sentences.csv contains the specific sentences that participants saw during the training phase organized row-wise
- what do you understand about the design from this file?



Triad	Pair	Part	sentence	novel1	novel2	novel3
1	foobly apple	training	I went to Zimziland because I heard you can get a foobly apple there.	foobly	Zimziland	
1	foobly apple	training	My sister doesn't like to have a foobly apple.	foobly		
1	foobly apple	training	Everybody I met in Zimziland would love to have a foobly apple.	foobly	Zimziland	
1	foobly apple	training	As soon as I arrived, I noticed a foobly apple.	foobly		
1	foobly apple	training	I would really love to have a foobly apple.	foobly		
1	foobly apple	training	One lady offered me a foobly apple.	foobly		
1	foobly apple	training	Maybe I would prefer a foobly apple.	foobly		
1	foobly apple	training	I went to a place where it's easy to find a foobly apple.	foobly		
1	foobly apple	training	The people I met there told me they saw a foobly apple that day.	foobly		
1	foobly apple	training	In the end, I decided I'd go looking for a foobly apple.	foobly		
1	foobly mipp	training	I didn't find a foobly mipp.	foobly	mipp	
1	foobly mipp	training	My sister told me you can get a foobly mipp there.	foobly	mipp	
1	foobly mipp	training	My friend Sally said she saw a foobly mipp once.	foobly	mipp	

importing sentences into index.html

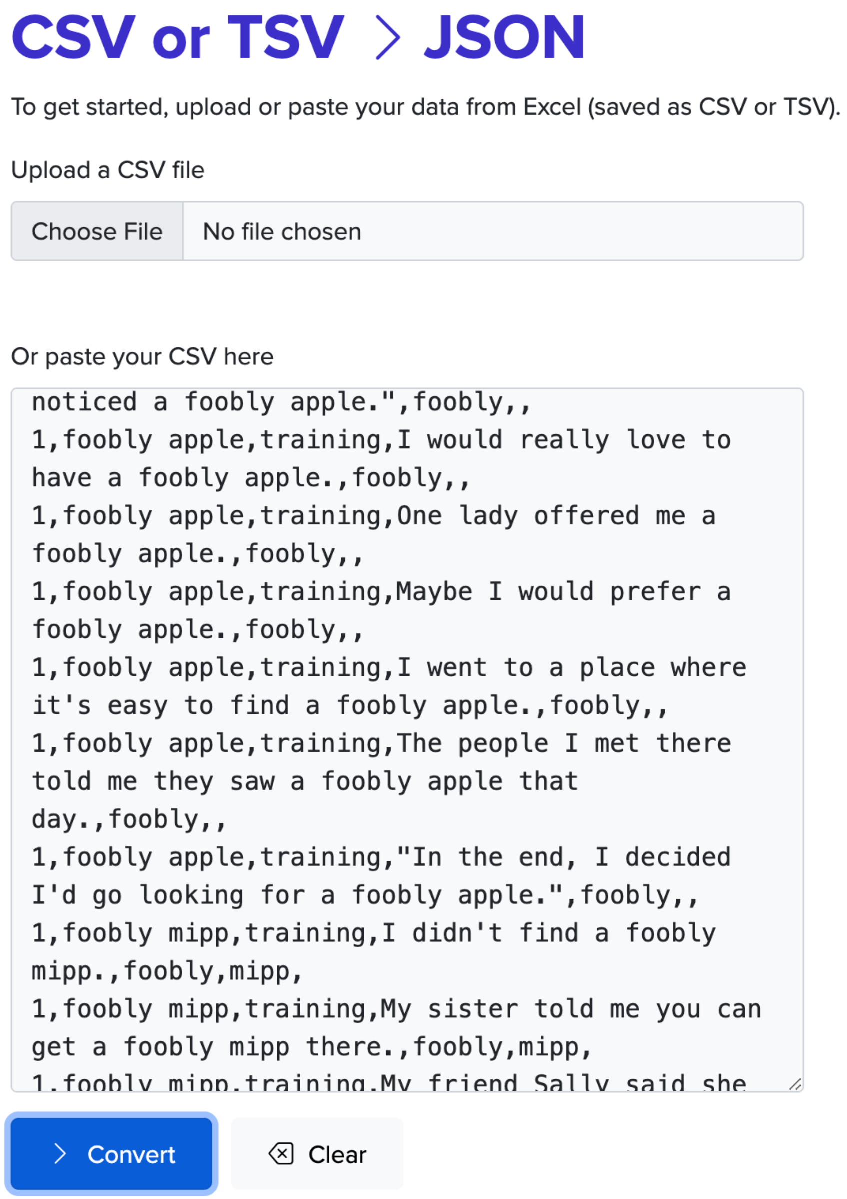
- CSV is not a format that is easily understood by jsPsych
- solution: convert each list to a format that jsPsych understands
    - JSON is a file format that jsPsych understands
    - excel/CSV to JSON conversion can be done online pretty easily!



Triad	Pair	Part	sentence	novel1	novel2	novel3
1	foobly apple	training	I went to Zimziland because I heard you can get a foobly apple there.	foobly	Zimziland	
1	foobly apple	training	My sister doesn't like to have a foobly apple.	foobly		
1	foobly apple	training	Everybody I met in Zimziland would love to have a foobly apple.	foobly	Zimziland	
1	foobly apple	training	As soon as I arrived, I noticed a foobly apple.	foobly		
1	foobly apple	training	I would really love to have a foobly apple.	foobly		
1	foobly apple	training	One lady offered me a foobly apple.	foobly		
1	foobly apple	training	Maybe I would prefer a foobly apple.	foobly		
1	foobly apple	training	I went to a place where it's easy to find a foobly apple.	foobly		
1	foobly apple	training	The people I met there told me they saw a foobly apple that day.	foobly		
1	foobly apple	training	In the end, I decided I'd go looking for a foobly apple.	foobly		
1	foobly mipp	training	I didn't find a foobly mipp.	foobly	mipp	
1	foobly mipp	training	My sister told me you can get a foobly mipp there.	foobly	mipp	
1	foobly mipp	training	My friend Sally said she saw a foobly mipp once.	foobly	mipp	

converting excel to JSON

- go to https://csvjson.com/csv2json
- upload sentences.csv
- click convert



CSV or TSV > JSON

To get started, upload or paste your data from Excel (saved as CSV or TSV).

Upload a CSV file

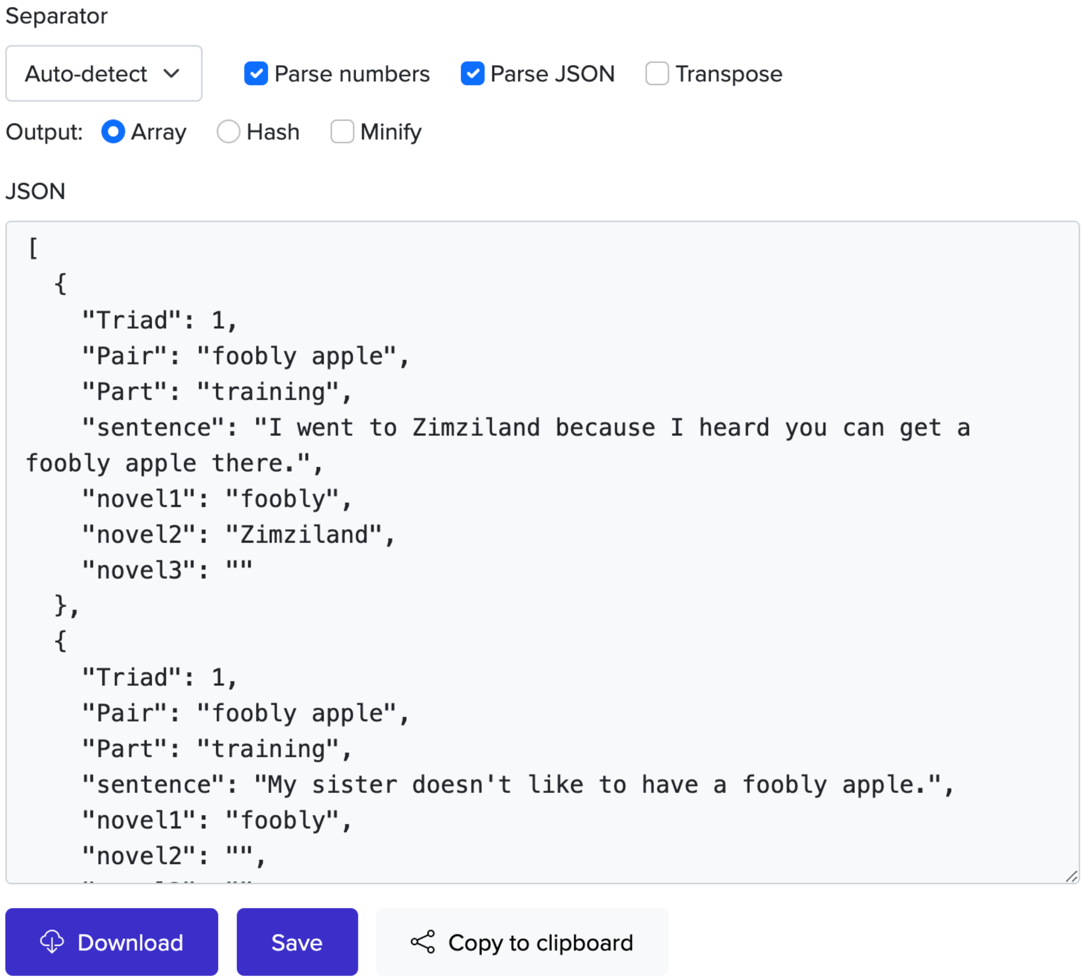
Choose File | No file chosen

Or paste your CSV here

```
noticed a foobly apple.",foobly,,
1,foobly apple,training,I would really love to
have a foobly apple.,foobly,,
1,foobly apple,training,One lady offered me a
foobly apple.,foobly,,
1,foobly apple,training,Maybe I would prefer a
foobly apple.,foobly,,
1,foobly apple,training,I went to a place where
it's easy to find a foobly apple.,foobly,,
1,foobly apple,training,The people I met there
told me they saw a foobly apple that
day.,foobly,,
1,foobly apple,training,"In the end, I decided
I'd go looking for a foobly apple.",foobly,,
1,foobly mipp,training,I didn't find a foobly
mipp.,foobly,mipp,
1,foobly mipp,training,My sister told me you can
get a foobly mipp there.,foobly,mipp,
1,foobly mipp,training,My friend Sally said she
```

saving a .js file

- download and save to the same folder as index.html as sentences.js
- open the file in VS Code



Separator

Auto-detect

☑ Parse numbers ☑ Parse JSON ☐ Transpose

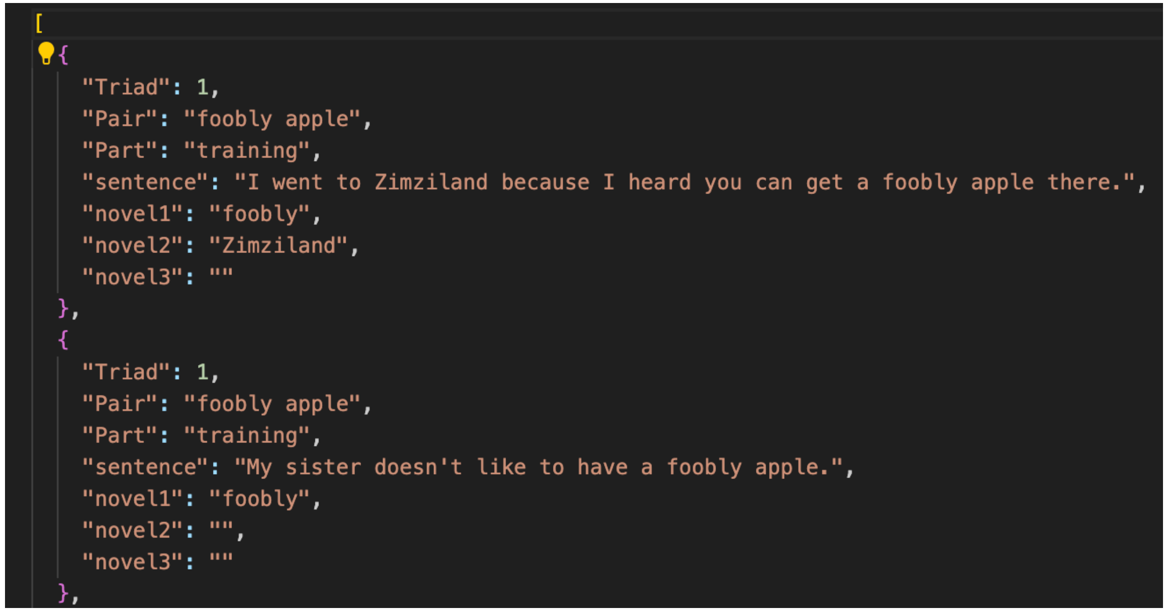
Output: ● Array ○ Hash ☐ Minify

JSON

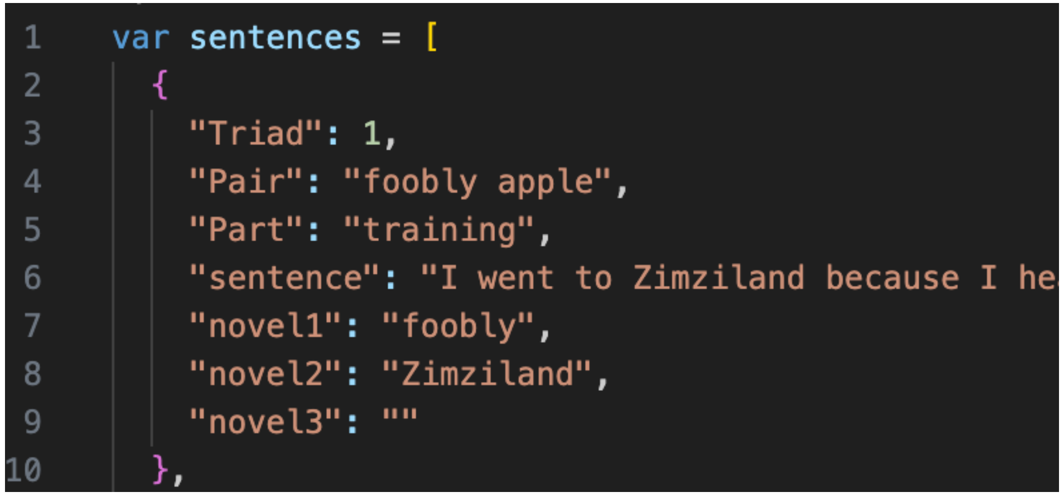
```
[
  {
    "Triad": 1,
    "Pair": "foobly apple",
    "Part": "training",
    "sentence": "I went to Zimziland because I heard you can get a
foobly apple there.",
    "novel1": "foobly",
    "novel2": "Zimziland",
    "novel3": ""
  },
  {
    "Triad": 1,
    "Pair": "foobly apple",
    "Part": "training",
    "sentence": "My sister doesn't like to have a foobly apple.",
    "novel1": "foobly",
    "novel2": "",
```

examining sentences.js

- notice that this file starts with a square bracket: this is a Javascript **object**
- each row in your CSV is defined by one {}, and each column and its value is denoted by column-name : value
- what is the name of the column that contains the sentence?
- to use it in your experiment you need to assign a name to this array
- add var sentences = before the square bracket
- save the file



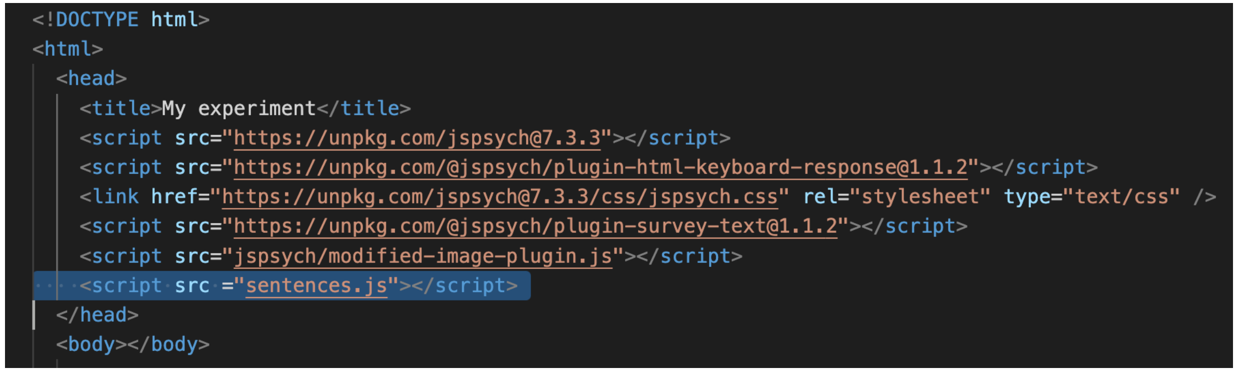
```
[
{
  "Triad": 1,
  "Pair": "foobly apple",
  "Part": "training",
  "sentence": "I went to Zimziland because I heard you can get a foobly apple there.",
  "novel1": "foobly",
  "novel2": "Zimziland",
  "novel3": ""
},
{
  "Triad": 1,
  "Pair": "foobly apple",
  "Part": "training",
  "sentence": "My sister doesn't like to have a foobly apple.",
  "novel1": "foobly",
  "novel2": "",
  "novel3": ""
},
```



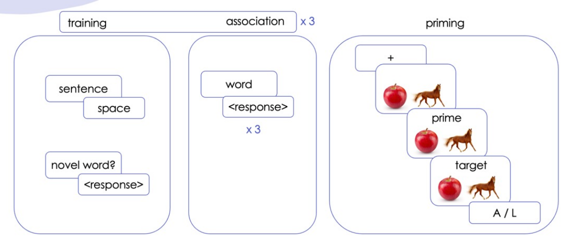
```
var sentences = [
  {
    "Triad": 1,
    "Pair": "foobly apple",
    "Part": "training",
    "sentence": "I went to Zimziland because I he
    "novel1": "foobly",
    "novel2": "Zimziland",
    "novel3": ""
  },
```

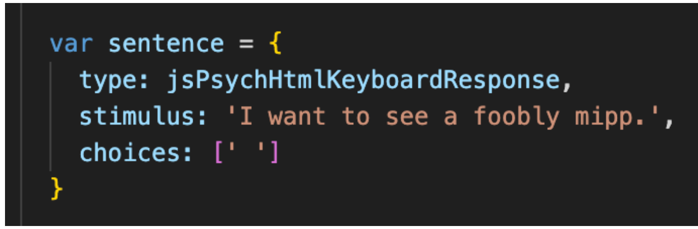
loading sentences.js into index.html

- import the .js file using the <script> tag inside the <head> tag as before
- now think about which trials in the experiment need to have these sentences, and which parameter will we need to modify for this



```
<!DOCTYPE html>
<html>
  <head>
    <title>My experiment</title>
    <script src="https://unpkg.com/jspsych@7.3.3"></script>
    <script src="https://unpkg.com/@jspsych/plugin-html-keyboard-response@1.1.2"></script>
    <link href="https://unpkg.com/jspsych@7.3.3/css/jspsych.css" rel="stylesheet" type="text/css" />
    <script src="https://unpkg.com/@jspsych/plugin-survey-text@1.1.2"></script>
    <script src="jspsych/modified-image-plugin.js"></script>
    <script src ="sentences.js"></script>
  </head>
  <body></body>
```

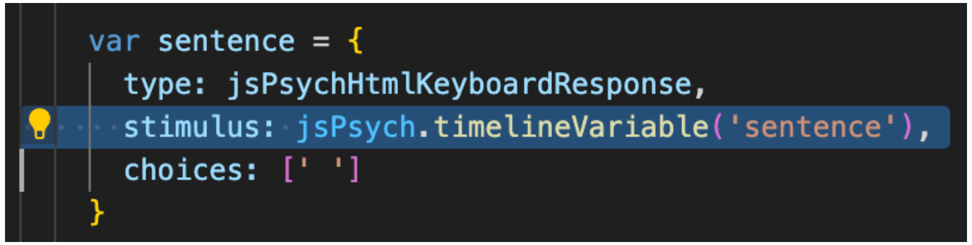




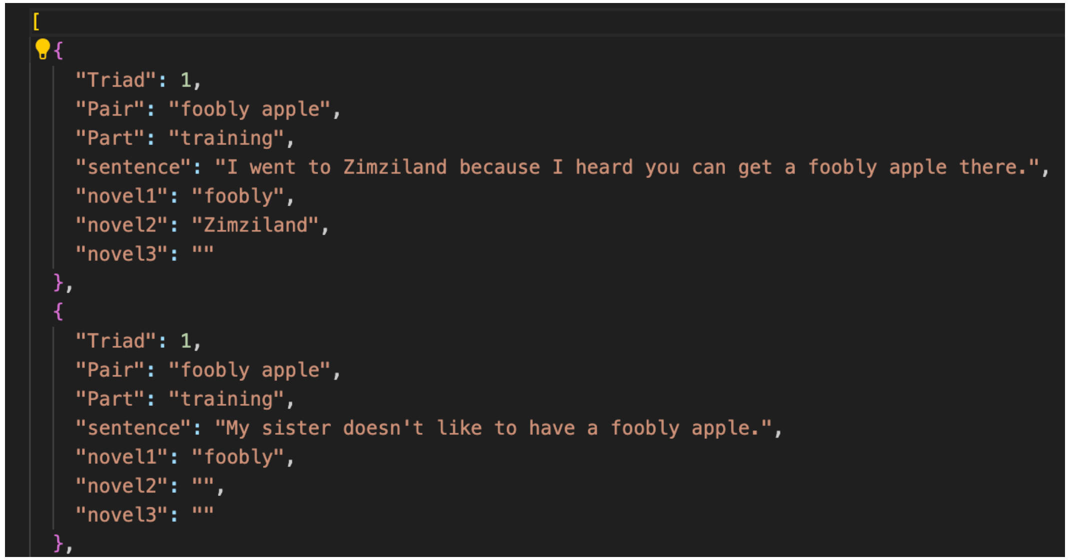
```
var sentence = {
  type: jsPsychHtmlKeyboardResponse,
  stimulus: 'I want to see a foobly mipp.',
  choices: [' ']
}
```

modifying our plugin

- we need to modify the sentence plugin (stimulus parameter) to import the list of sentences instead of only displaying one sentence
- we use a specific function called jsPsych.timelineVariable() with the argument being the name of the column that contains our sentences
- what if I wanted to instead display the word pair?



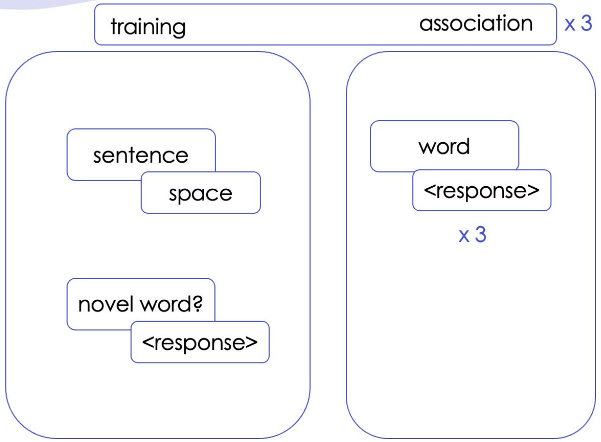
```
var sentence = {
    type: jsPsychHtmlKeyboardResponse,
    stimulus: jsPsych.timelineVariable('sentence'),
    choices: [' ']
}
```

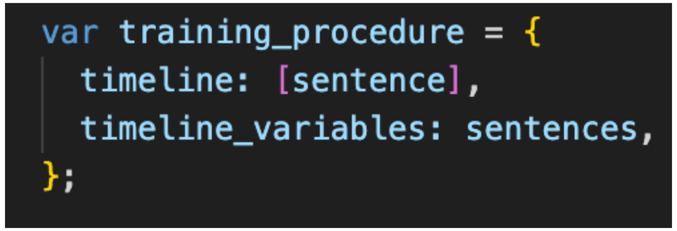


```
[
{
    "Triad": 1,
    "Pair": "foobly apple",
    "Part": "training",
    "sentence": "I went to Zimziland because I heard you can get a foobly apple there.",
    "novel1": "foobly",
    "novel2": "Zimziland",
    "novel3": ""
  },
  {
    "Triad": 1,
    "Pair": "foobly apple",
    "Part": "training",
    "sentence": "My sister doesn't like to have a foobly apple.",
    "novel1": "foobly",
    "novel2": "",
    "novel3": ""
  },
```

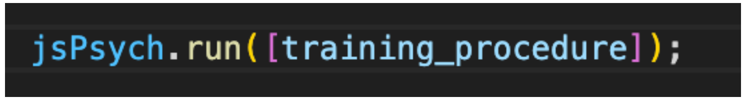
creating a timeline

- next, we need to connect the jsPsych.timelineVariable() function inside the plugin trial to the .js file we created using a timeline
- here, we are defining a timeline that consists of the sentence plugin trial, uses the sentences variable we defined inside sentences.js
- try running just the training_procedure
- save and reload index.html
- randomize the order of sentences, reload

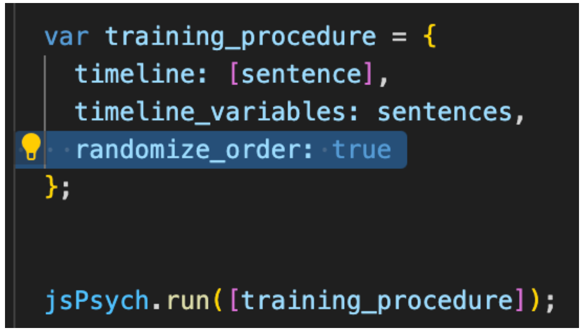




```
var training_procedure = {
  timeline: [sentence],
  timeline_variables: sentences,
};
```



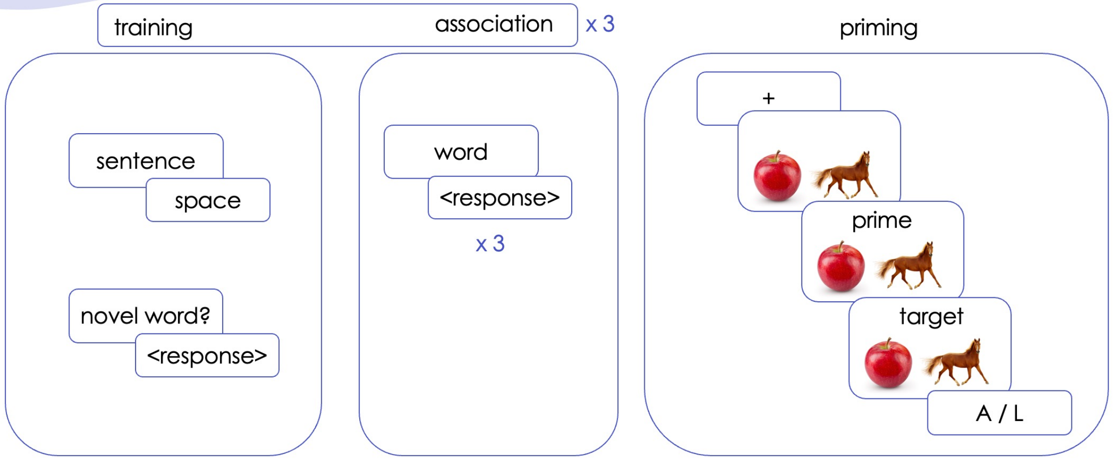
```
jsPsych.run([training_procedure]);
```



```
var training_procedure = {
  timeline: [sentence],
  timeline_variables: sentences,
  randomize_order: true
};


jsPsych.run([training_procedure]);
```

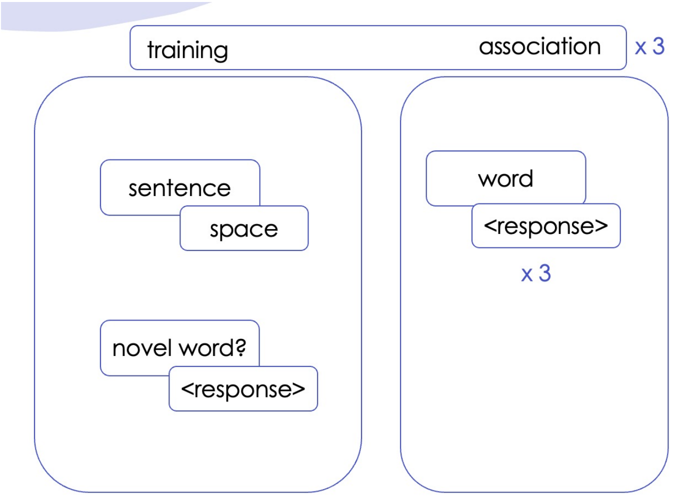
adding to the timeline

- we need a way to display an attention check every few trials
- we also need to add the association trials after the sentences



creating association stimuli

- what words were presented as cues for the association task?
- create a new file called association.js inside the same folder as index.html
- using the same format as sentences.js, create 4 rows with the column name as cue and the value inside the column being the novel words

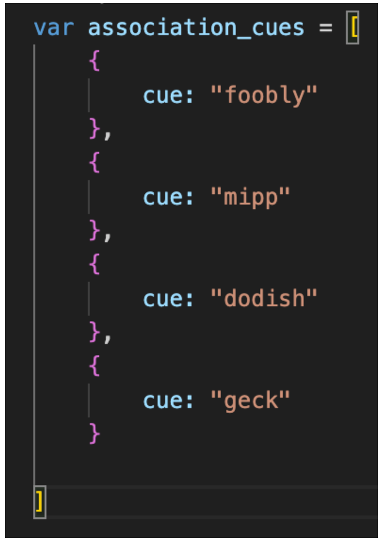


training association x 3

sentence
space

novel word?
<response>

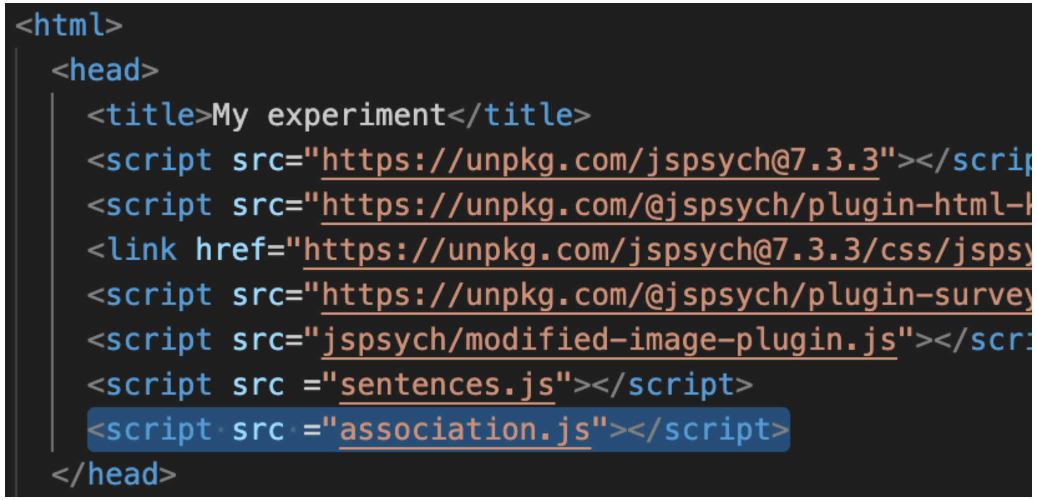
word
<response>
x 3



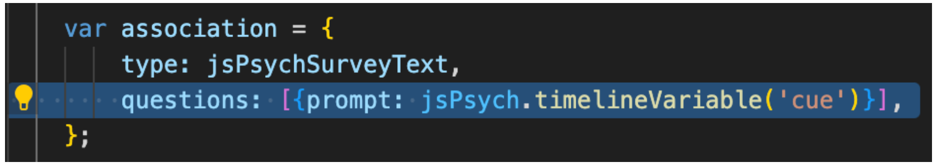
```
var association_cues = [
    {
        cue: "foobly"
    },
    {
        cue: "mipp"
    },
    {
        cue: "dodish"
    },
    {
        cue: "geck"
    }

]
```

importing associations.js into index.html

- add a <script> tag to include the .js file into your experiment
- now modify your association plugin trial to import the column that contains the novel words inside the association.js file



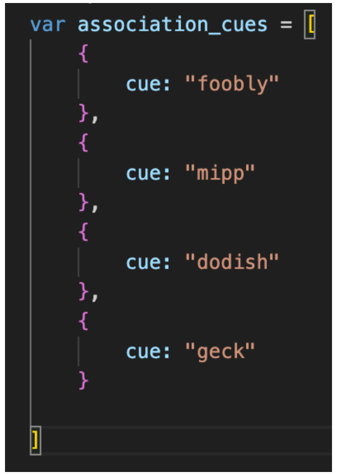
```
<html>
  <head>
    <title>My experiment</title>
    <script src="https://unpkg.com/jspsych@7.3.3"></scrip
    <script src="https://unpkg.com/@jspsych/plugin-html-
    <link href="https://unpkg.com/jspsych@7.3.3/css/jspsy
    <script src="https://unpkg.com/@jspsych/plugin-survey
    <script src="jspsych/modified-image-plugin.js"></scr
    <script src ="sentences.js"></script>
    <script src ="association.js"></script>
  </head>
```



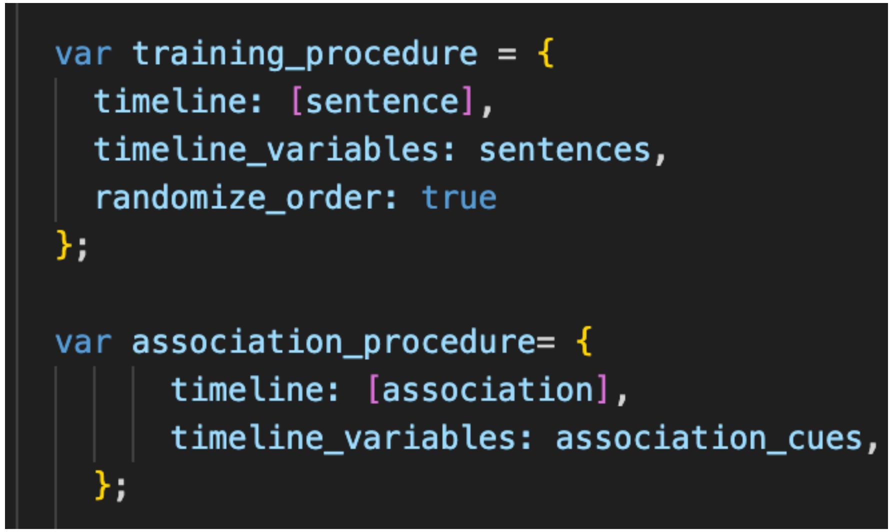
```
var association = {
    type: jsPsychSurveyText,
    questions: [{prompt: jsPsych.timelineVariable('cue')}],
};
```

creating a procedure for association

- we define association_procedure that pulls the variable from associations.js



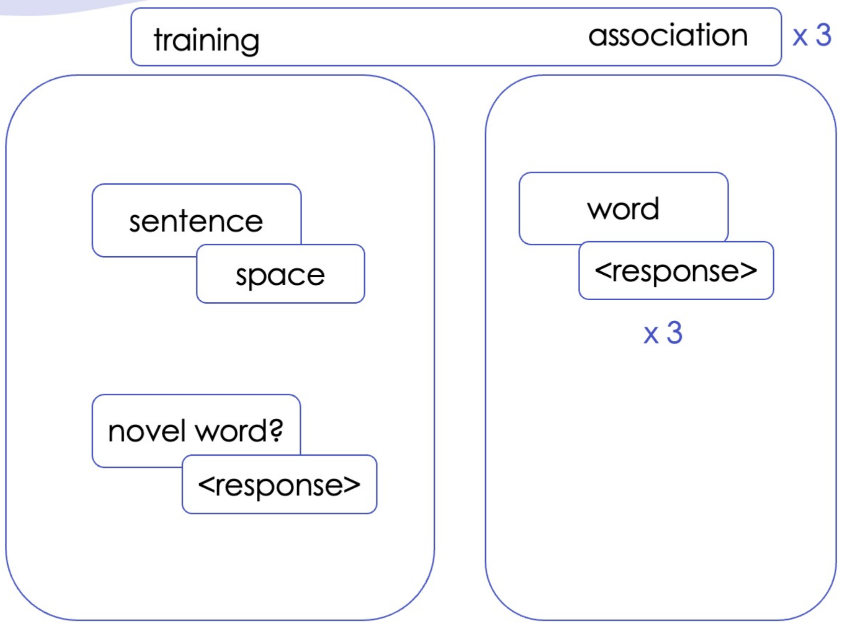
```
var association_cues = [
    {
        cue: "foobly"
    },
    {
        cue: "mipp"
    },
    {
        cue: "dodish"
    },
    {
        cue: "geck"
    }

]
```

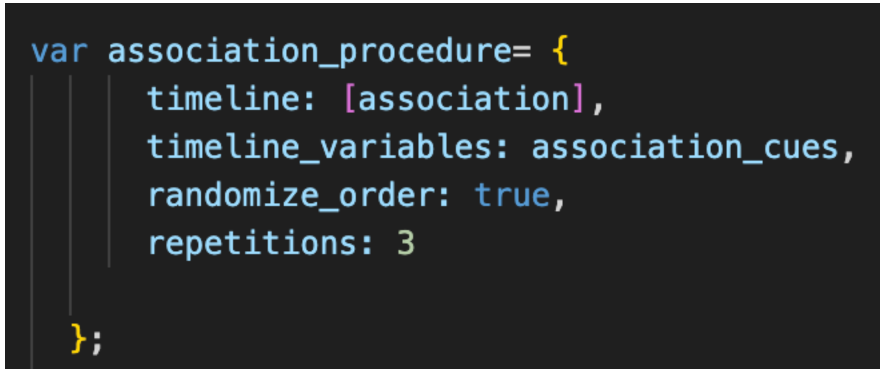


```
var training_procedure = {
  timeline: [sentence],
  timeline_variables: sentences,
  randomize_order: true
};

var association_procedure= {
      timeline: [association],
      timeline_variables: association_cues,
  };
```

repeating associations

- we also want the association procedure to be repeated three times and in random order

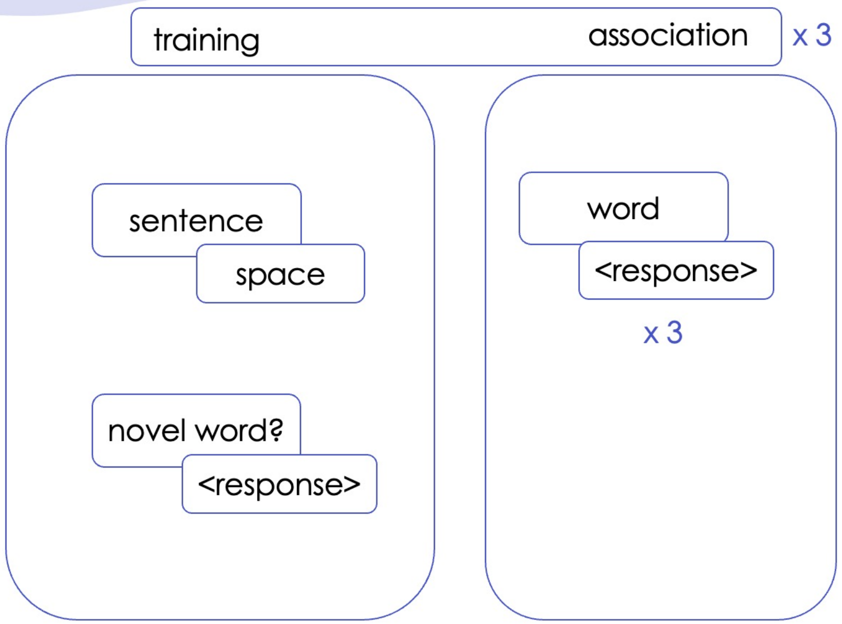


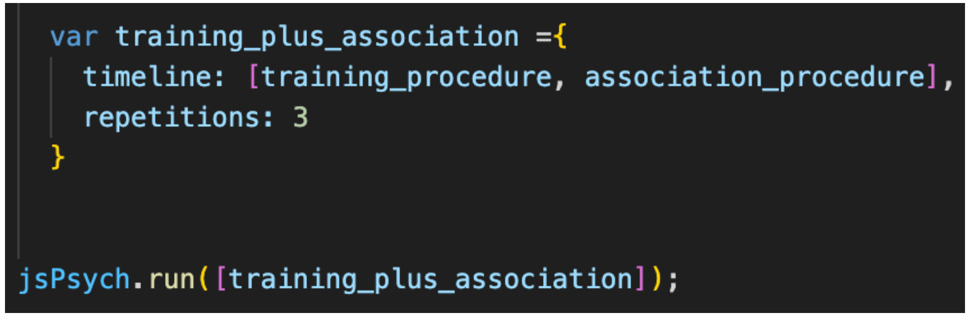


```
var association_procedure= {
        timeline: [association],
        timeline_variables: association_cues,
        randomize_order: true,
        repetitions: 3

  };
```

putting it all together

- the sentence + (attention) + association sequence occurs three times in the experiment
- we achieve this by defining yet another timeline variable

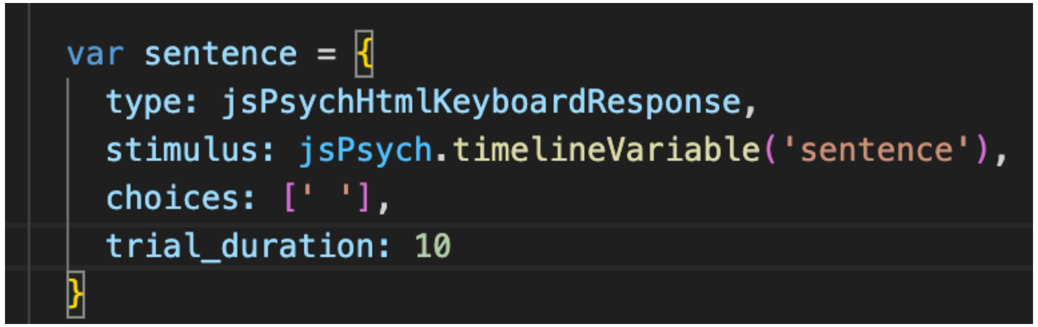




```
  var training_plus_association ={
    timeline: [training_procedure, association_procedure],
    repetitions: 3
  }


jsPsych.run([training_plus_association]);
```

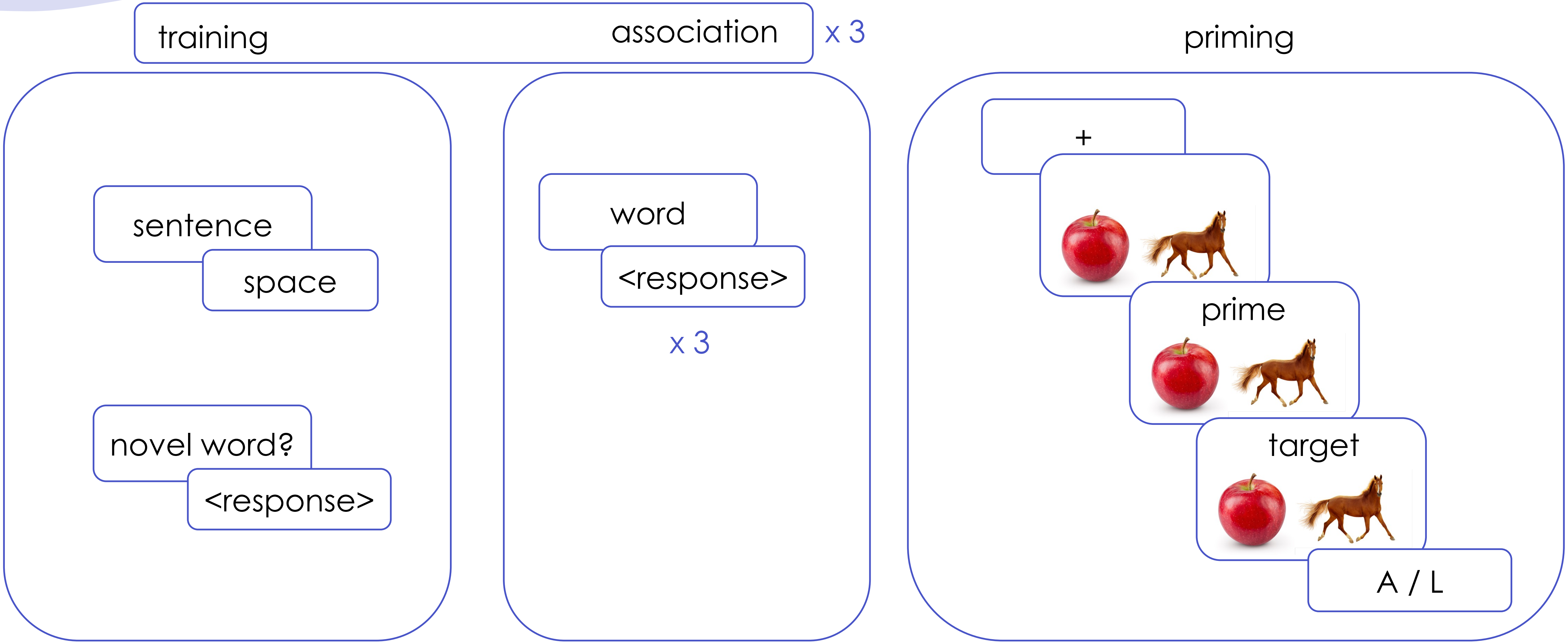
testing hacks

- running through all 40 sentences can be annoying
- solution: you could reduce the trial_duration when you’re testing different parts of your experiment



```
var sentence = {
  type: jsPsychHtmlKeyboardResponse,
  stimulus: jsPsych.timelineVariable('sentence'),
  choices: [' '],
  trial_duration: 10
}
```

experiment recap



training association x 3

sentence

space

novel word?

<response>

word

<response>

x 3

priming

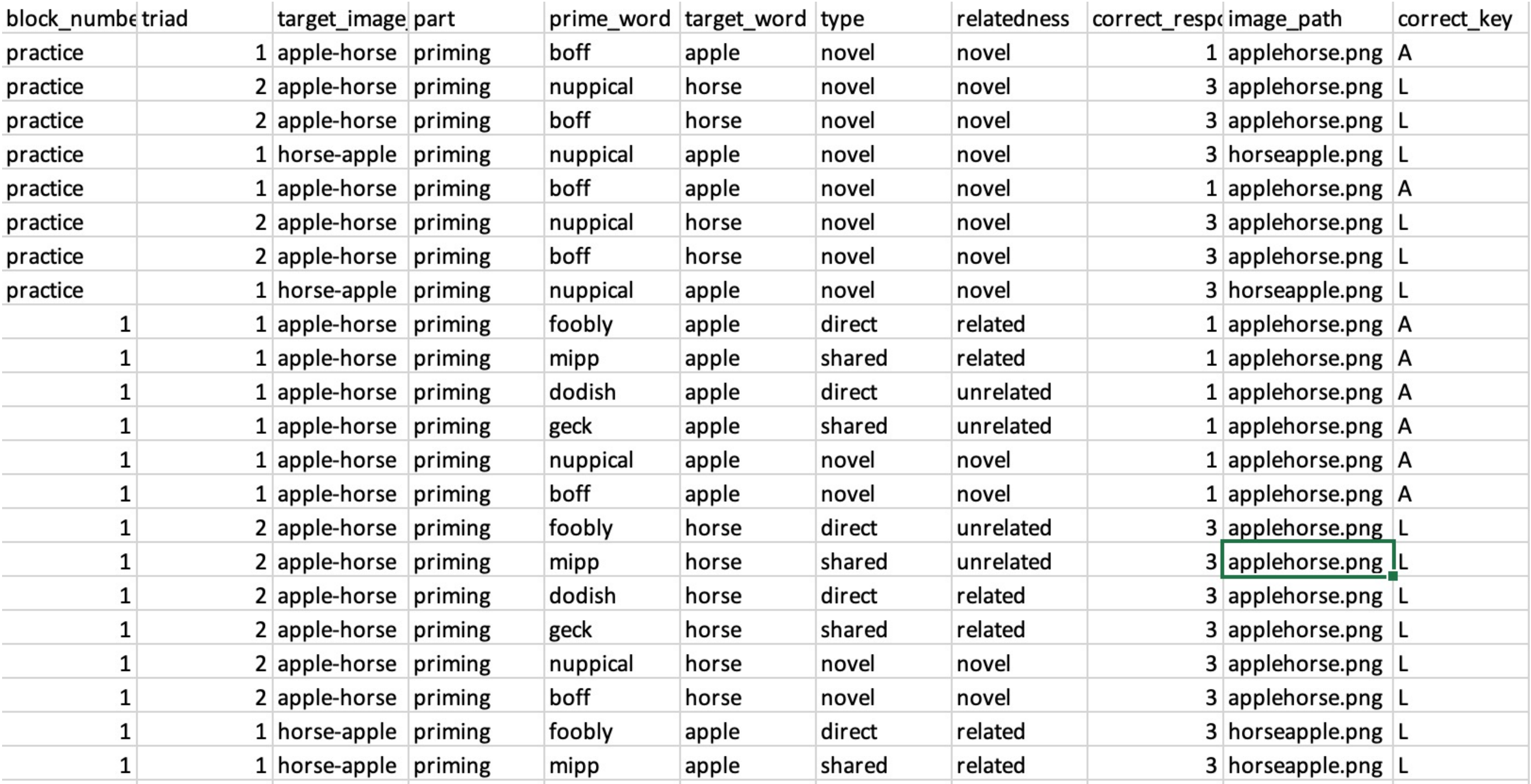
+

prime

target

A / L

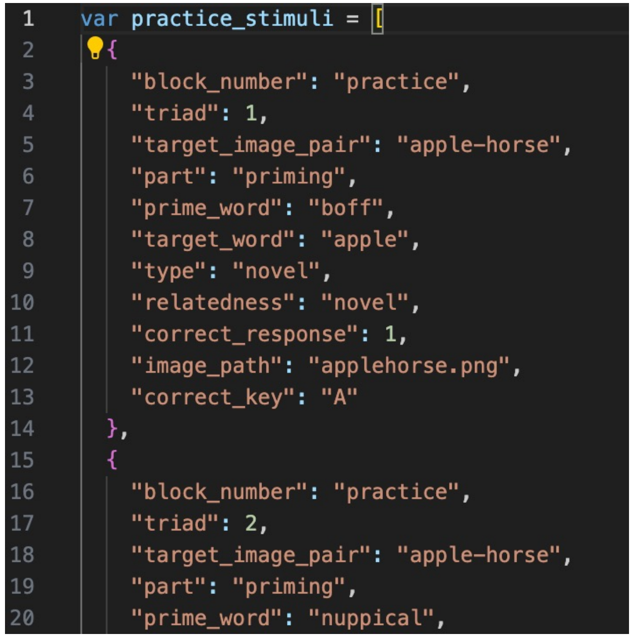
explore priming.csv



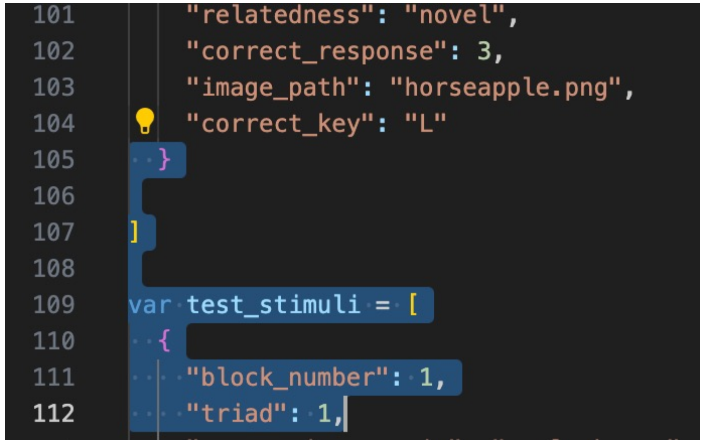
block_numbe	triad	target_image_	part	prime_word	target_word	type	relatedness	correct_resp	image_path	correct_key
practice	1	apple-horse	priming	boff	apple	novel	novel	1	applehorse.png	A
practice	2	apple-horse	priming	nuppical	horse	novel	novel	3	applehorse.png	L
practice	2	apple-horse	priming	boff	horse	novel	novel	3	applehorse.png	L
practice	1	horse-apple	priming	nuppical	apple	novel	novel	3	horseapple.png	L
practice	1	apple-horse	priming	boff	apple	novel	novel	1	applehorse.png	A
practice	2	apple-horse	priming	nuppical	horse	novel	novel	3	applehorse.png	L
practice	2	apple-horse	priming	boff	horse	novel	novel	3	applehorse.png	L
practice	1	horse-apple	priming	nuppical	apple	novel	novel	3	horseapple.png	L
1	1	apple-horse	priming	foobly	apple	direct	related	1	applehorse.png	A
1	1	apple-horse	priming	mipp	apple	shared	related	1	applehorse.png	A
1	1	apple-horse	priming	dodish	apple	direct	unrelated	1	applehorse.png	A
1	1	apple-horse	priming	geck	apple	shared	unrelated	1	applehorse.png	A
1	1	apple-horse	priming	nuppical	apple	novel	novel	1	applehorse.png	A
1	1	apple-horse	priming	boff	apple	novel	novel	1	applehorse.png	A
1	2	apple-horse	priming	foobly	horse	direct	unrelated	3	applehorse.png	L
1	2	apple-horse	priming	mipp	horse	shared	unrelated	3	applehorse.png	L
1	2	apple-horse	priming	dodish	horse	direct	related	3	applehorse.png	L
1	2	apple-horse	priming	geck	horse	shared	related	3	applehorse.png	L
1	2	apple-horse	priming	nuppical	horse	novel	novel	3	applehorse.png	L
1	2	apple-horse	priming	boff	horse	novel	novel	3	applehorse.png	L
1	1	horse-apple	priming	foobly	apple	direct	related	3	horseapple.png	L
1	1	horse-apple	priming	mipp	apple	shared	related	3	horseapple.png	L

creating priming.js

- convert the priming.CSV file to priming.js using https://csvjson.com/csv2json
- download and save as priming.js
- define var practice_stimuli and var test_stimuli



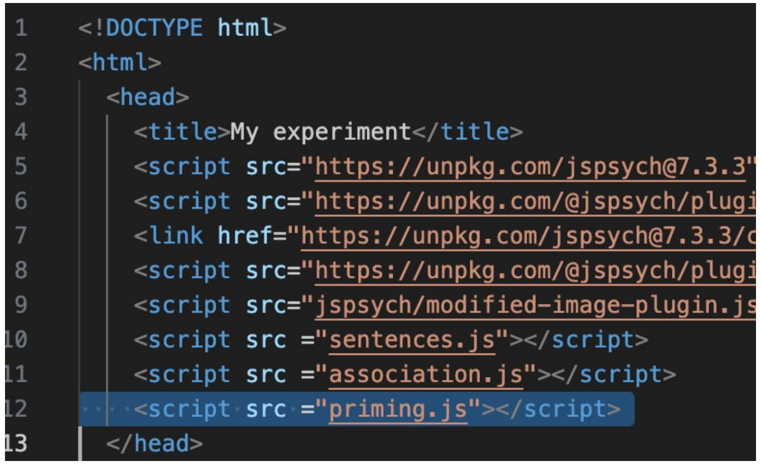
```
var practice_stimuli = [
  {
    "block_number": "practice",
    "triad": 1,
    "target_image_pair": "apple-horse",
    "part": "priming",
    "prime_word": "boff",
    "target_word": "apple",
    "type": "novel",
    "relatedness": "novel",
    "correct_response": 1,
    "image_path": "applehorse.png",
    "correct_key": "A"
  },
  {
    "block_number": "practice",
    "triad": 2,
    "target_image_pair": "apple-horse",
    "part": "priming",
    "prime_word": "nuppical",
```



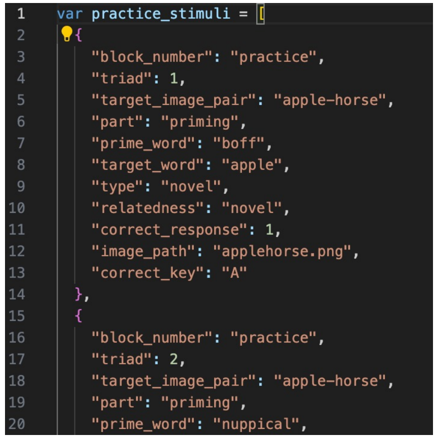
```
    "relatedness": "novel",
    "correct_response": 3,
    "image_path": "horseapple.png",
    "correct_key": "L"
  }

]

var test_stimuli = [
  {
    "block_number": 1,
    "triad": 1,
```

load into index.html

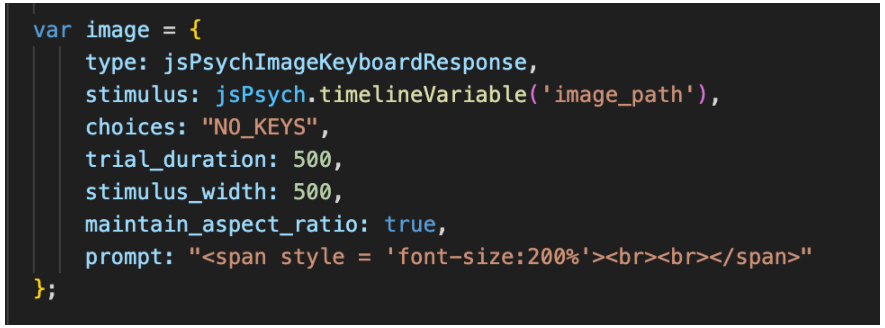
- use the <script> tags as before
- change the stimulus parameters for the image plugin trial to the column that stores the names of the images that need to be displayed



```
<!DOCTYPE html>
<html>
  <head>
    <title>My experiment</title>
    <script src="https://unpkg.com/jspsych@7.3.3"
    <script src="https://unpkg.com/@jspsych/plugi
    <link href="https://unpkg.com/jspsych@7.3.3/c
    <script src="https://unpkg.com/@jspsych/plugi
    <script src="jspsych/modified-image-plugin.js
    <script src ="sentences.js"></script>
    <script src ="association.js"></script>
    <script src ="priming.js"></script>
  </head>
```



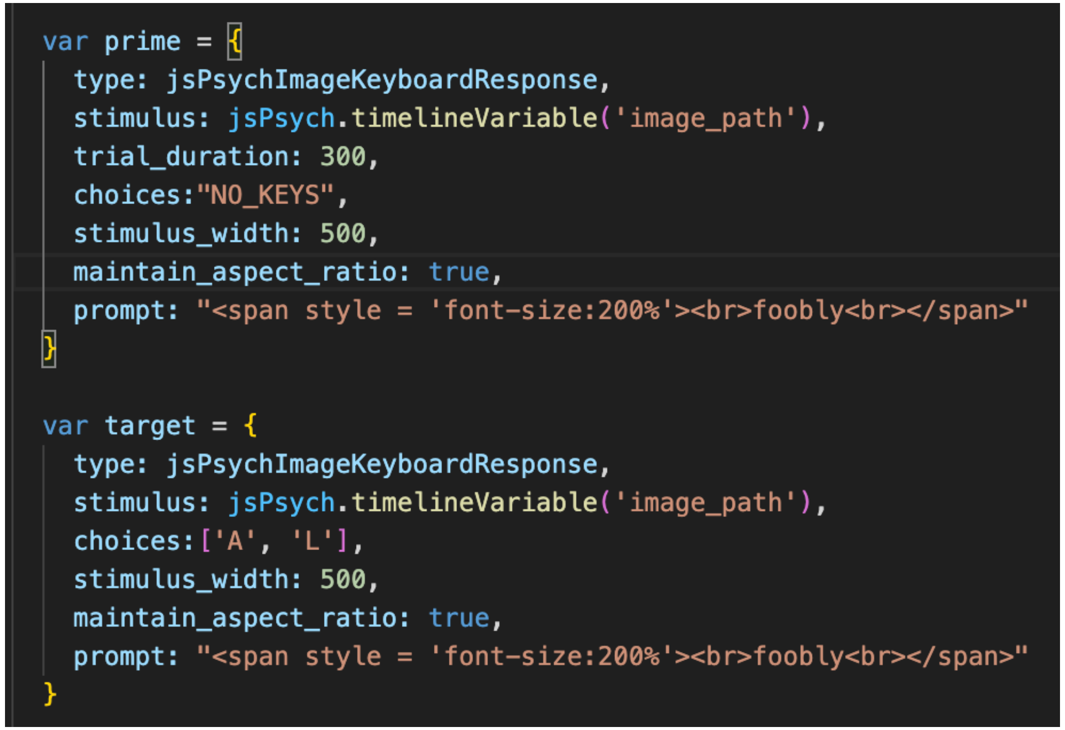
```
var practice_stimuli = [
 {
   "block_number": "practice",
   "triad": 1,
   "target_image_pair": "apple-horse",
   "part": "priming",
   "prime_word": "boff",
   "target_word": "apple",
   "type": "novel",
   "relatedness": "novel",
   "correct_response": 1,
   "image_path": "applehorse.png",
   "correct_key": "A"
 },
 {
   "block_number": "practice",
   "triad": 2,
   "target_image_pair": "apple-horse",
   "part": "priming",
   "prime_word": "nuppical",
```



```
var image = {
    type: jsPsychImageKeyboardResponse,
    stimulus: jsPsych.timelineVariable('image_path'),
    choices: "NO_KEYS",
    trial_duration: 500,
    stimulus_width: 500,
    maintain_aspect_ratio: true,
    prompt: "<span style = 'font-size:200%'><br><br></span>"
};
```

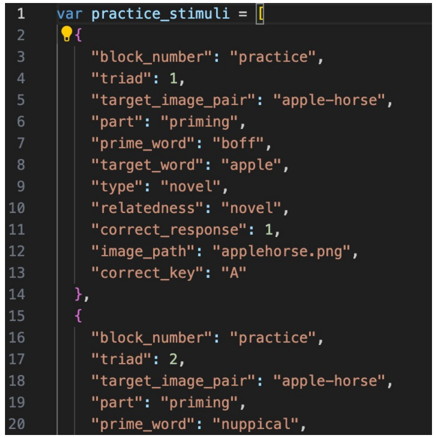
modifying prime and target trials: 1

- change the stimulus being displayed
- do we need to change anything else?
- we also need to change the prompt being displayed, as our prompt contains the prime/target words



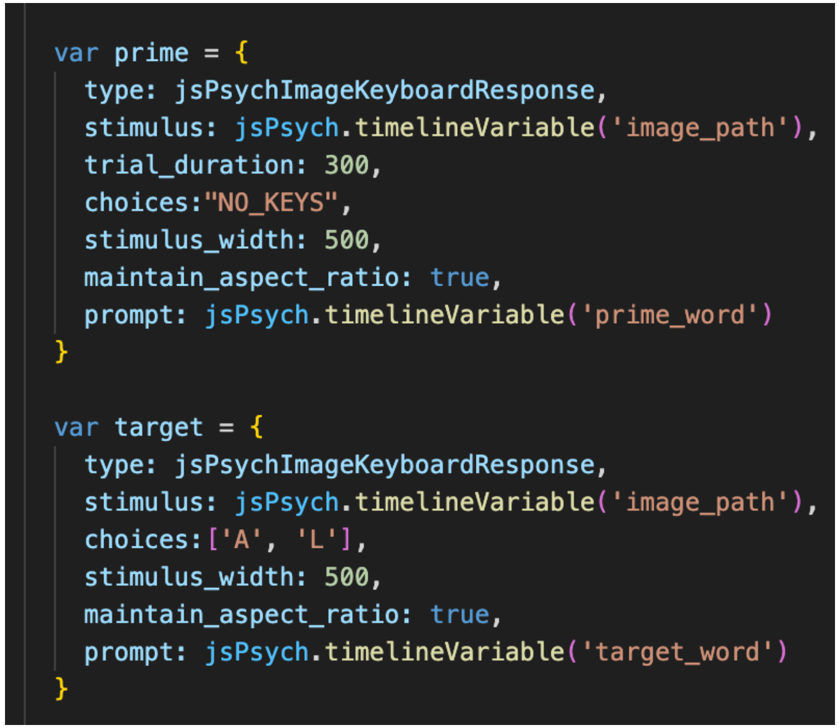
```
var prime = {
  type: jsPsychImageKeyboardResponse,
  stimulus: jsPsych.timelineVariable('image_path'),
  trial_duration: 300,
  choices:"NO_KEYS",
  stimulus_width: 500,
  maintain_aspect_ratio: true,
  prompt: "<span style = 'font-size:200%'><br>foobly<br></span>"
}

var target = {
  type: jsPsychImageKeyboardResponse,
  stimulus: jsPsych.timelineVariable('image_path'),
  choices:['A', 'L'],
  stimulus_width: 500,
  maintain_aspect_ratio: true,
  prompt: "<span style = 'font-size:200%'><br>foobly<br></span>"
}
```

modifying prime and target trials: 1



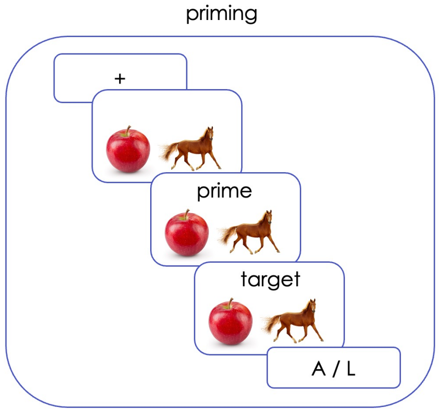
```
var practice_stimuli = [
  {
    "block_number": "practice",
    "triad": 1,
    "target_image_pair": "apple-horse",
    "part": "priming",
    "prime_word": "boff",
    "target_word": "apple",
    "type": "novel",
    "relatedness": "novel",
    "correct_response": 1,
    "image_path": "applehorse.png",
    "correct_key": "A"
  },
  {
    "block_number": "practice",
    "triad": 2,
    "target_image_pair": "apple-horse",
    "part": "priming",
    "prime_word": "nuppical",
```

- inside priming.js, what column names contain the words to be displayed on prime and target trials?
- modify the prompt accordingly

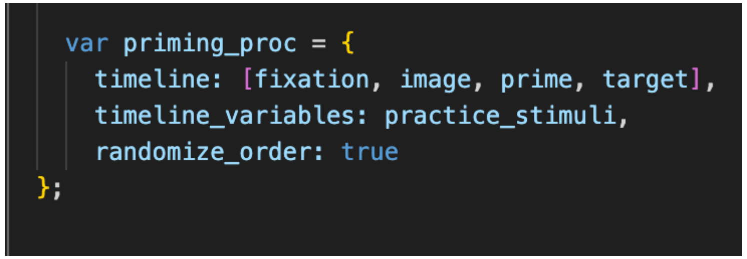


```
var prime = {
  type: jsPsychImageKeyboardResponse,
  stimulus: jsPsych.timelineVariable('image_path'),
  trial_duration: 300,
  choices:"NO_KEYS",
  stimulus_width: 500,
  maintain_aspect_ratio: true,
  prompt: jsPsych.timelineVariable('prime_word')
}

var target = {
  type: jsPsychImageKeyboardResponse,
  stimulus: jsPsych.timelineVariable('image_path'),
  choices:['A', 'L'],
  stimulus_width: 500,
  maintain_aspect_ratio: true,
  prompt: jsPsych.timelineVariable('target_word')
}
```

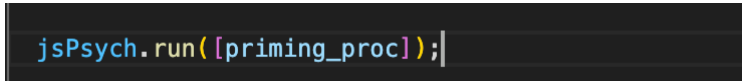
creating a priming procedure



- create a timeline variable that has a sequence of events that all use the priming.js file
- run this priming procedure
- save and reload index.html in your browser



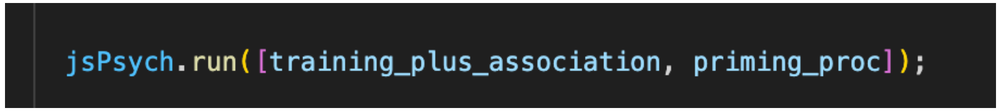
```
  var priming_proc = {
    timeline: [fixation, image, prime, target],
    timeline_variables: practice_stimuli,
    randomize_order: true
};
```



```
jsPsych.run([priming_proc]);
```

sum of the parts!

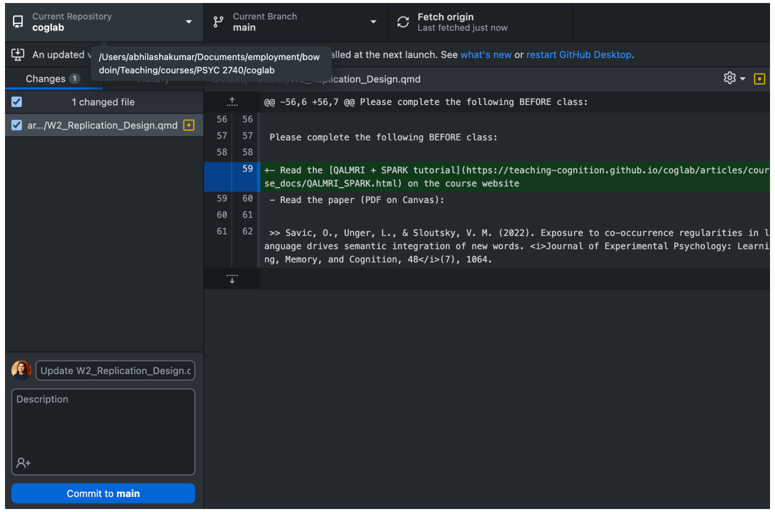
- now, we have a version of the experiment where training sentences are presented, free association happens, and the priming task is conducted
- modify the run statement to see the current experiment workflow



```
jsPsych.run([training_plus_association, priming_proc]);
```

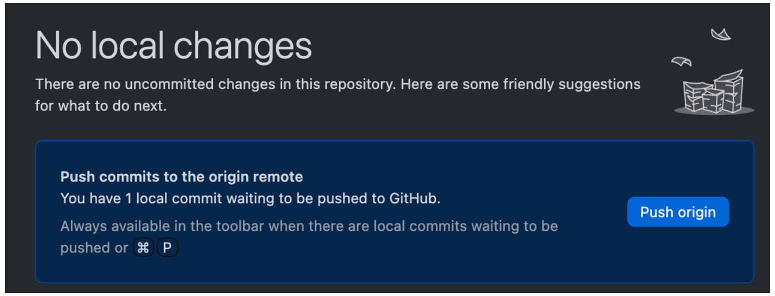
saving your progress so far…

- save your index.html file
- open GitHub Desktop
- review changes, commit, and push
- check if changes have reflected online!



An updated v… /Users/abhilashakumar/Documents/employment/bowdoin/Teaching/courses/PSYC 2740/coglab …alled at the next launch. See what's new or restart GitHub Desktop.

```
@@ -56,6 +56,7 @@ Please complete the following BEFORE class:

 Please complete the following BEFORE class:

+- Read the [QALMRI + SPARK tutorial](https://teaching-cognition.github.io/coglab/articles/course_docs/QALMRI_SPARK.html) on the course website
 - Read the paper (PDF on Canvas):

 >> Savic, O., Unger, L., & Sloutsky, V. M. (2022). Exposure to co-occurrence regularities in language drives semantic integration of new words. <i>Journal of Experimental Psychology: Learning, Memory, and Cognition, 48</i>(7), 1064.
```



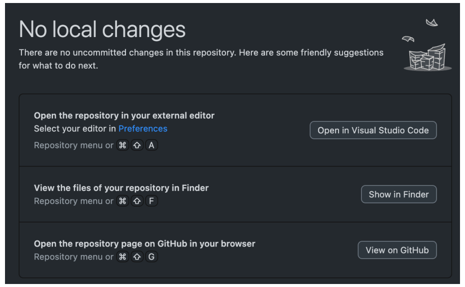
No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Push commits to the origin remote

You have 1 local commit waiting to be pushed to GitHub.

Always available in the toolbar when there are local commits waiting to be pushed or ⌘ P



No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

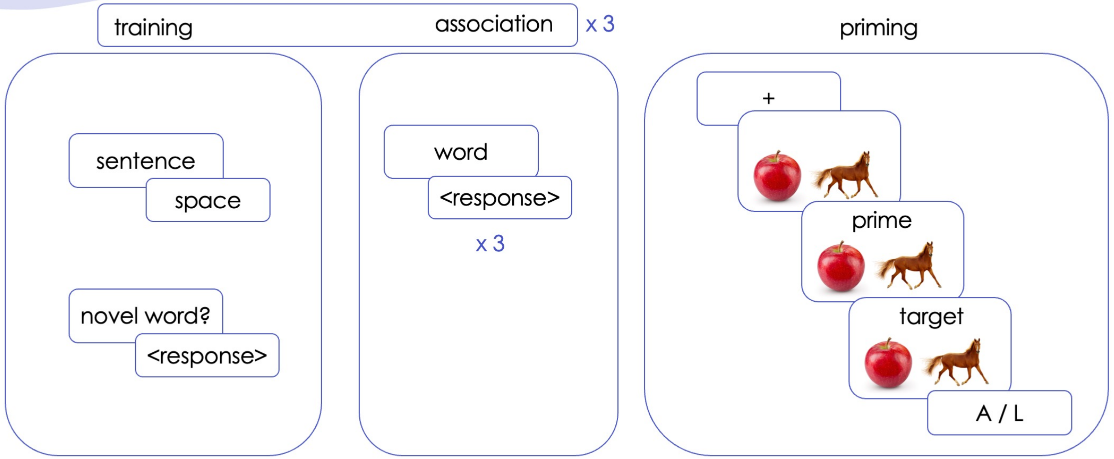
Open the repository in your external editor
Select your editor in Preferences
Repository menu or ⌘ ⇧ A

View the files of your repository in Finder
Repository menu or ⌘ ⇧ F

Open the repository page on GitHub in your browser
Repository menu or ⌘ ⇧ G

outstanding issues

- fixing position & style of prime/target words
- adding instruction screens
- attention checks
- feedback
- recording data



next class

- **before** class
  - *prep:* conditional timelines and providing feedback
  - *prep:* design draft (project milestone #4)
- **during** class
  - fixing position & style of prime/target words
  - adding instruction screens
  - attention checks
  - feedback