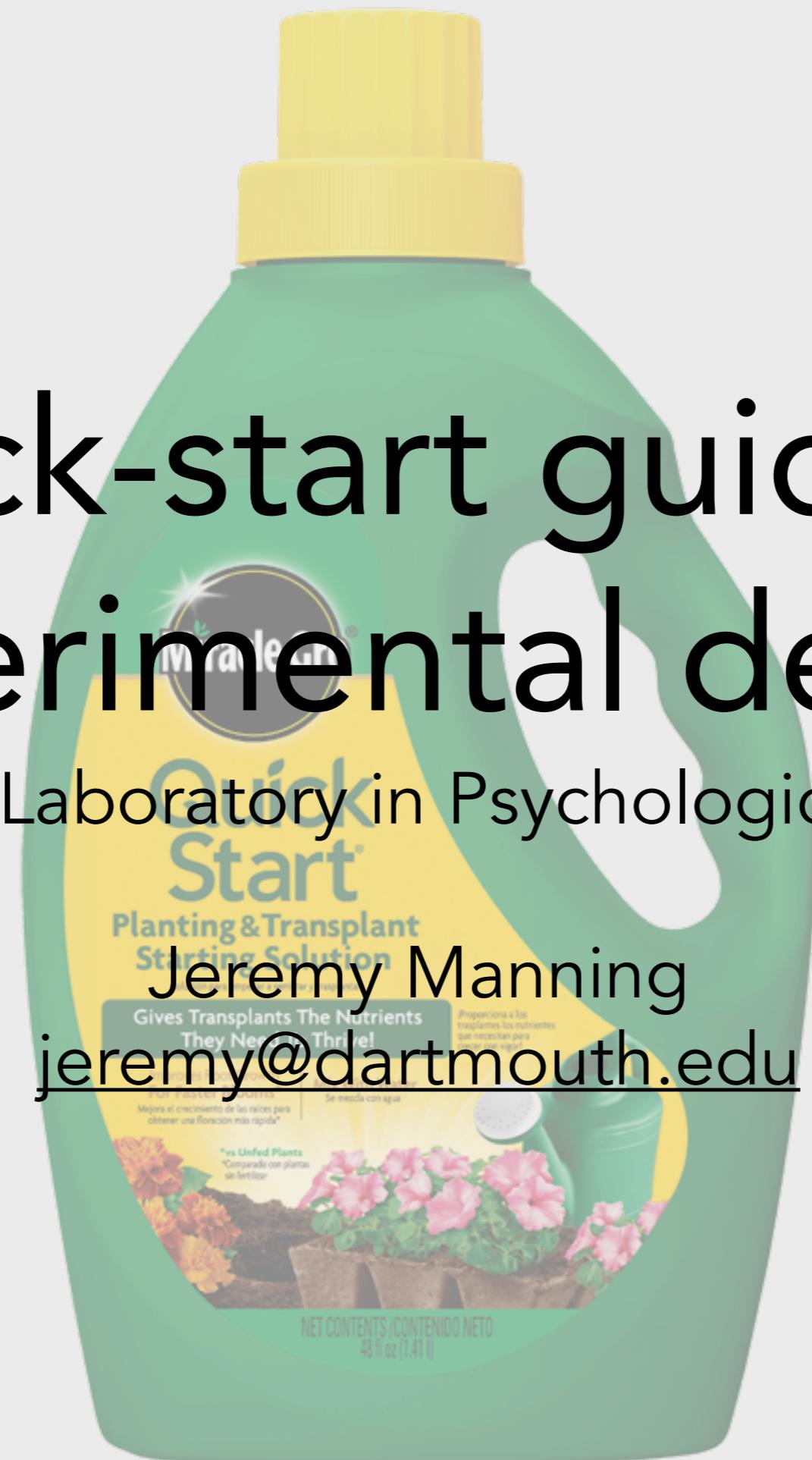


Quick-start guide to experimental design

PSYC 11: Laboratory in Psychological Science

Jeremy Manning

jeremy@dartmouth.edu



What is the purpose of running an experiment?

- Understand or explore how something works
- Distinguish between several potential alternatives
- Get more information (data!)

What “counts” as an experiment?

- Observing something happening without manipulating it
- Self-reflection or self-report
- Constructing a scenario in which the process you want to study occurs
- Running a simulation (via real-life or computation)

Observational studies

- Prime directive: do not interfere
- Go out into the world and collect data
- Examples:
 - Studying an animal colony
 - Watch how people interact at a party
 - Record facial expressions as people enter and exit a movie screening
 - Measure highway traffic density at different times of day

Self-report

- Socrates: “To know thyself is the beginning of wisdom”
- Trust people to accurately report their opinions, thoughts, feelings, and behaviors
- Examples:
 - Voter preference/opinion surveys
 - Mental health screening forms
 - Feedback surveys
 - Personal narratives

Constructed scenarios

- In general, there are two widely used approaches:
 - Classic: maximize **control**
 - Naturalistic: maximize **realism**

Classic experiments

- Figure out the simplest possible version of the process or phenomenon you're interested in
- If you think the process depends on participants' experiences (e.g., stimuli, framing, effort, etc.), carefully control those factors across several conditions
- Use behaviors (across conditions if needed) as a stand-in for the “real thing”

Classic experiments: examples

- Vision: can you see this tiny spot of light?
- Audition: what word do you hear when I play this sound?
- Memory: which images in this set do you recognize?
- Cognitive control: how tightly can you regulate your responses to a cue?

Naturalistic experiments

- Create a scenario that closely approximates the real-world process you want to study
- Measure as much stuff as possible about people's behaviors
- Mine the data for patterns

Naturalistic experiments: examples

- Vision: where do you fixate at each moment while moving around in a VR environment?
- Memory: how you describe “what happened” in a movie?
- Motor control: how do people adjust their gait on different terrains?

Simulations

- Real-world: use props, actors, and technology to make it seem like the participant is in a particular situation, or that the world works a particular way
- Simulated: change the properties of a simulated environment and study an artificial agent's behaviors

Simulations: examples

- Confederate experiments
- Misleading participants about the situation or rules
- Simulating how robots of different configurations complete obstacle courses under different gravitational loads

Implementation tools

- Notebooks, audiovisual recordings
- Google forms, qualtrics
- Slideshows
- PsychoPy, jsPsych
- Google Colaboratory

Notebooks and recordings

- Good for observational studies
- Collect lots of richly structured data!
- No setup required!
- Can be tough to analyze (either time-consuming to annotate, or tricky to design effective automated approaches)

Google forms and qualtrics

The screenshot shows the Google Forms editor. At the top, there's a toolbar with icons for saving, undoing, redoing, and other settings. Below the toolbar, the navigation bar includes 'Questions' (which is selected), 'Responses', and 'Settings'. The main area displays a form titled 'Untitled form' with a 'Form description' placeholder. A single question is present: 'Untitled Question' (type: Multiple choice). It has two options: 'Option 1' and 'Add option or add "Other"'. Below the question are standard edit controls: a trash can icon, a required indicator, and a more options menu. On the right side of the screen, there's a vertical toolbar with icons for adding new sections, tables, images, and other form elements.

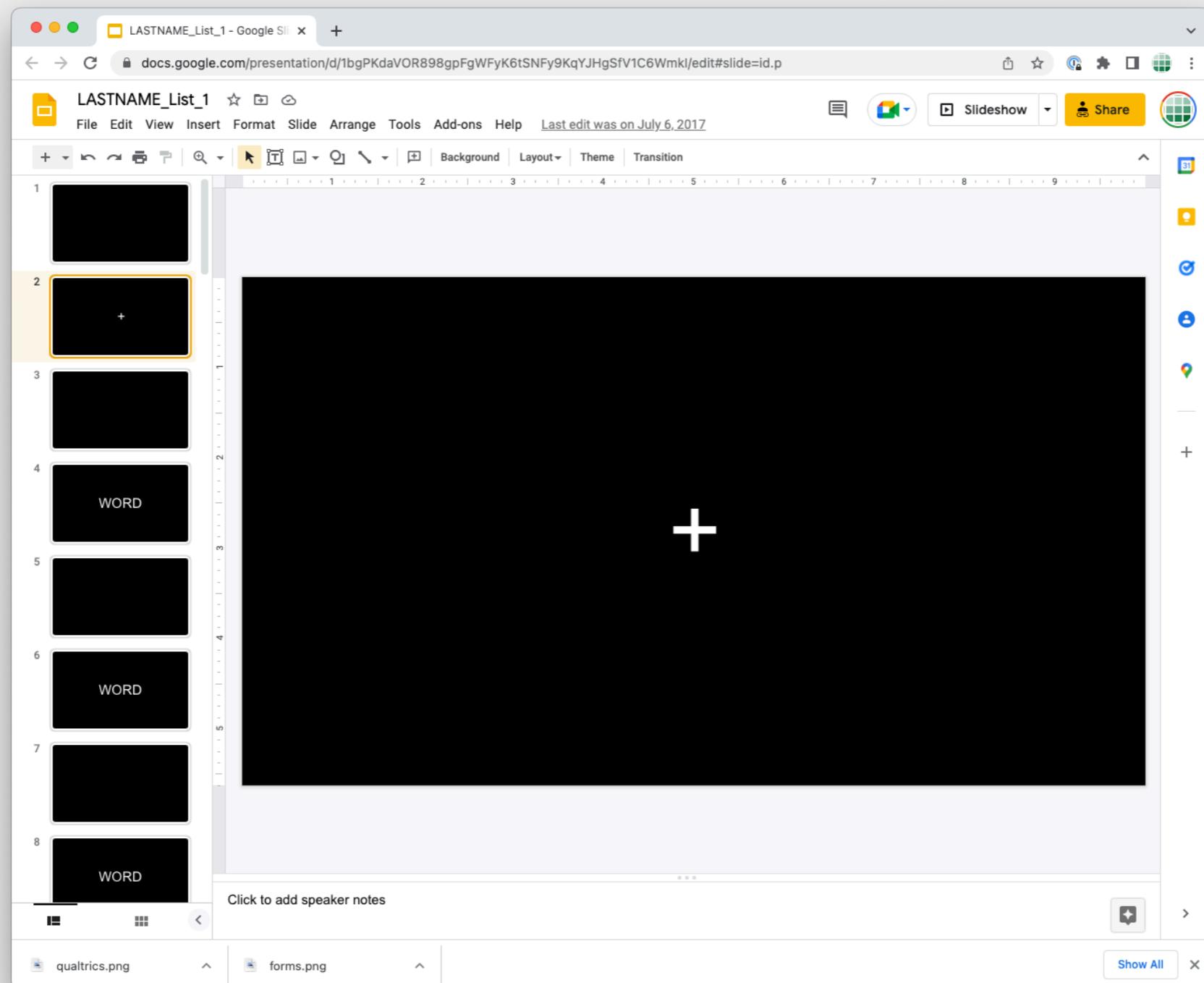


Google Forms



Qualtrics

Slideshows



PsychoPy

NavonTask.psyexp - PsychoPy Builder (v2021.2.0)

File Edit View Tools Experiment Demos Pavlovia.org Window Help

Routines

feedback instrMain instrPractice thanks trial x

Components

Favorites

- Keyboard
- < >
- Image
- Polygon
- Sound
- Textbox

Stimuli

Responses

Custom

EEG

Eyetracking

I/O

Flow

Insert Routine

Insert Loop

instrPractice

trial (0.00s)

feedback (1.00s)

instrMain

practiceTrials (1x16 random)

trial (0.00s)

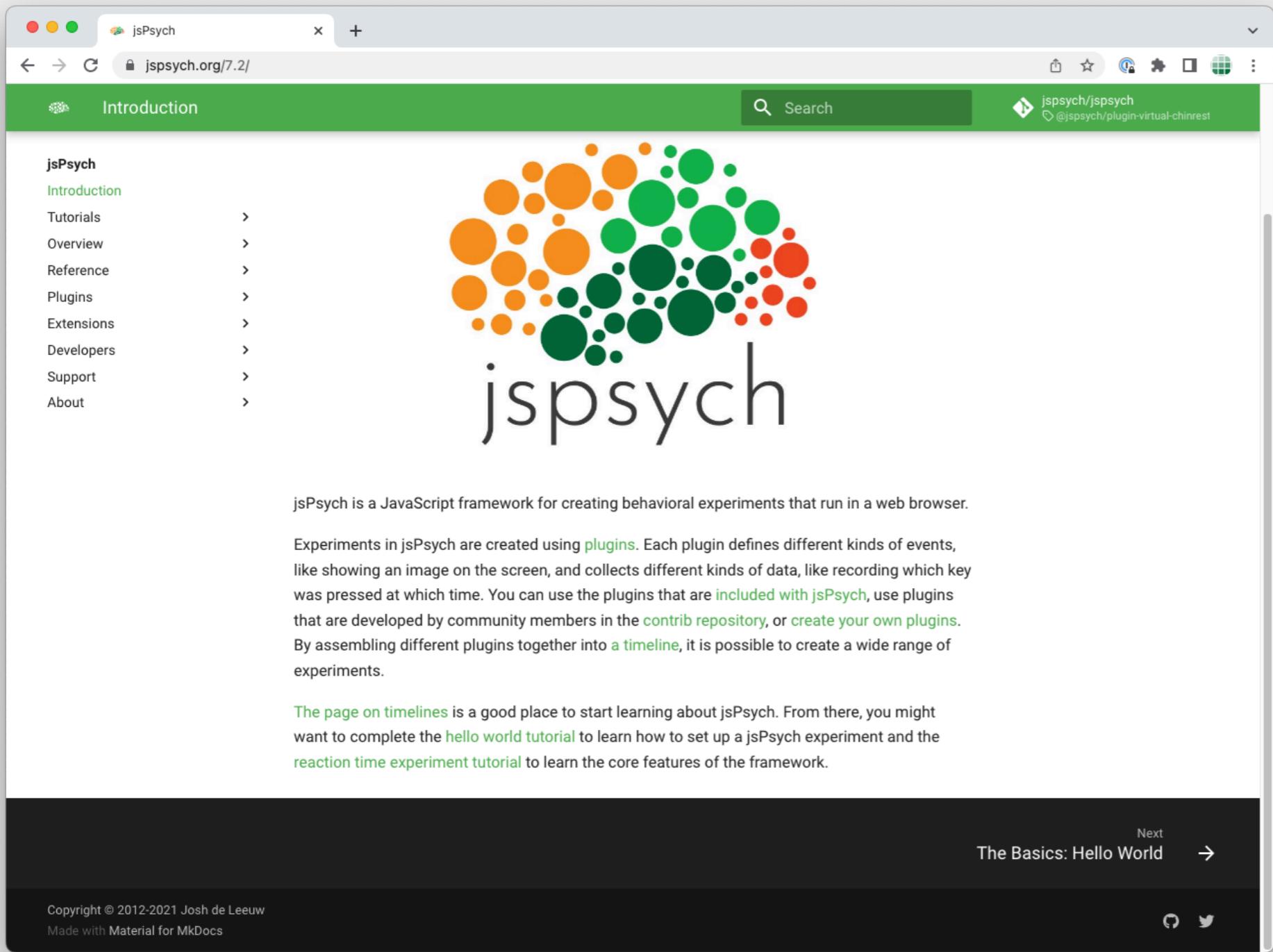
thanks (2.00s)

trials (4x16 random)

The screenshot shows the PsychoPy Builder interface. The top menu includes File, Edit, View, Tools, Experiment, Demos, Pavlovia.org, Window, and Help. The toolbar contains various icons for file operations, code, and components. The main area features a timeline from 0 to 8 seconds labeled 't (sec)' with four events: 'fixate' (red T), 'stimulus' (image), 'mask' (image), and 'resp' (key icon). To the right is a 'Components' panel with sections for Favorites (Keyboard, <>, Image, Polygon, Sound, Textbox) and categories like Stimuli, Responses, Custom, EEG, Eyetracking, and I/O. Below is a 'Flow' section with a sequence of routines: instrPractice, trial (0.00s), feedback (1.00s), instrMain, practiceTrials (1x16 random), trial (0.00s), thanks (2.00s), and trials (4x16 random). A sidebar on the left offers Insert Routine and Insert Loop options.



jsPsych



The screenshot shows a web browser window displaying the jsPsych.org/7.2/ page. The page has a green header bar with the title "jsPsych" and a navigation menu. The main content area features the jsPsych logo (a cluster of colored circles) and the word "jspsych". Below the logo, there is a brief introduction to the framework, followed by a section about experiments and plugins, and finally a link to the timeline page. At the bottom, there is a footer with copyright information and social media links.

jsPsych is a JavaScript framework for creating behavioral experiments that run in a web browser.

Experiments in jsPsych are created using [plugins](#). Each plugin defines different kinds of events, like showing an image on the screen, and collects different kinds of data, like recording which key was pressed at which time. You can use the plugins that are [included with jsPsych](#), use plugins that are developed by community members in the [contrib repository](#), or [create your own plugins](#). By assembling different plugins together into a [timeline](#), it is possible to create a wide range of experiments.

The [page on timelines](#) is a good place to start learning about jsPsych. From there, you might want to complete the [hello world tutorial](#) to learn how to set up a jsPsych experiment and the [reaction time experiment tutorial](#) to learn the core features of the framework.

Next →
The Basics: Hello World

Copyright © 2012-2021 Josh de Leeuw
Made with Material for MkDocs



Other considerations

- Use what you already know
- Simplicity: the art of maximizing the amount of work **not** done
- Work together
- Ask for help