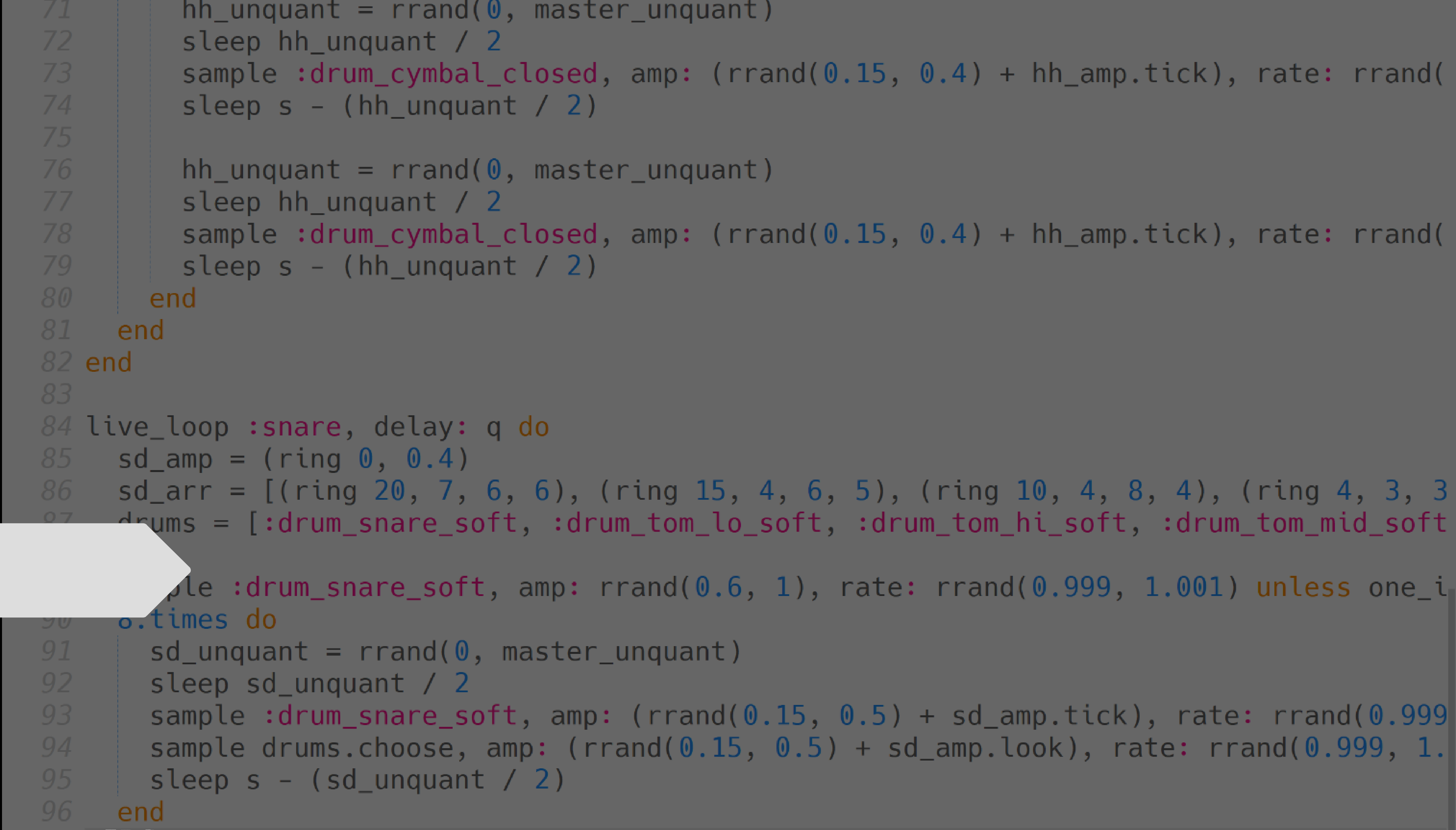
Toward Equitable Learning through Rhizomatic Design

Jared O'Leary
BootUp PD

K-8 Technology -> Coding

Is this equitable?

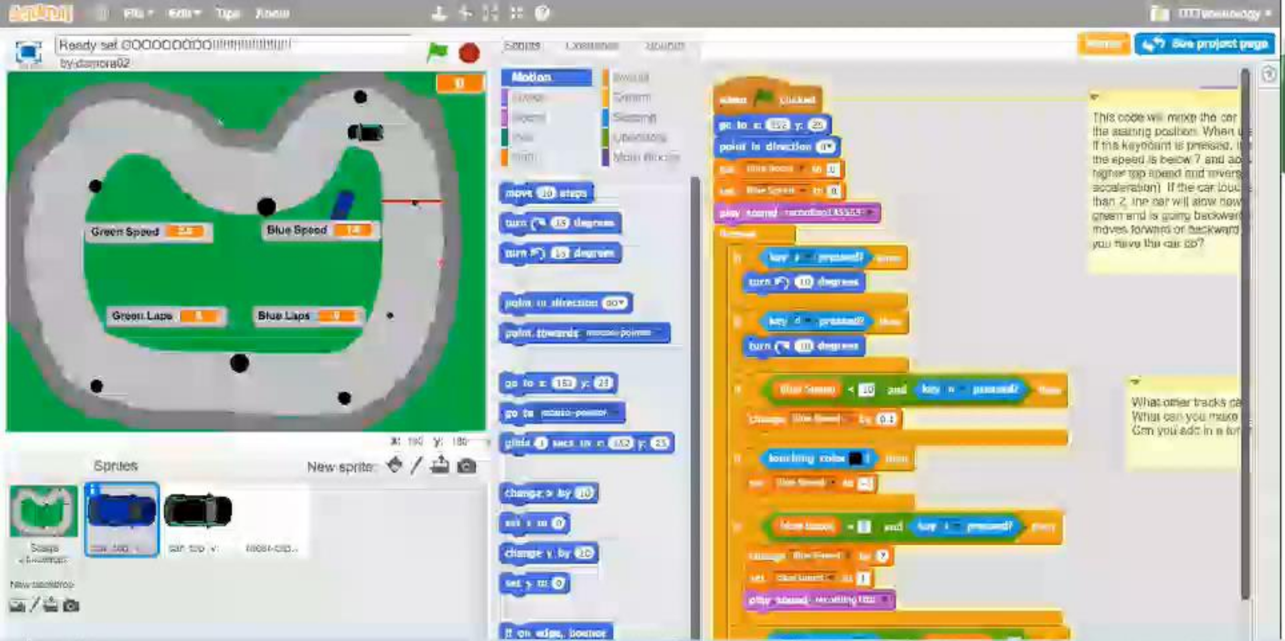
Maybe



```
      hh_unquant = rrand(0, master_unquant)
      sleep hh_unquant / 2
      sample :drum_cymbal_closed, amp: (rrand(0.15, 0.4) + hh_amp.tick), rate: rrand(
      sleep s - (hh_unquant / 2)

      hh_unquant = rrand(0, master_unquant)
      sleep hh_unquant / 2
      sample :drum_cymbal_closed, amp: (rrand(0.15, 0.4) + hh_amp.tick), rate: rrand(
      sleep s - (hh_unquant / 2)
    end
  end
end

live_loop :snare, delay: q do
  sd_amp = (ring 0, 0.4)
  sd_arr = [(ring 20, 7, 6, 6), (ring 15, 4, 6, 5), (ring 10, 4, 8, 4), (ring 4, 3, 3
  drums = [:drum_snare_soft, :drum_tom_lo_soft, :drum_tom_hi_soft, :drum_tom_mid_soft

   ple :drum_snare_soft, amp: rrand(0.6, 1), rate: rrand(0.999, 1.001) unless one_i
  8.times do
    sd_unquant = rrand(0, master_unquant)
    sleep sd_unquant / 2
    sample :drum_snare_soft, amp: (rrand(0.15, 0.5) + sd_amp.tick), rate: rrand(0.999
    sample drums.choose, amp: (rrand(0.15, 0.5) + sd_amp.look), rate: rrand(0.999, 1.
    sleep s - (sd_unquant / 2)
  end
```

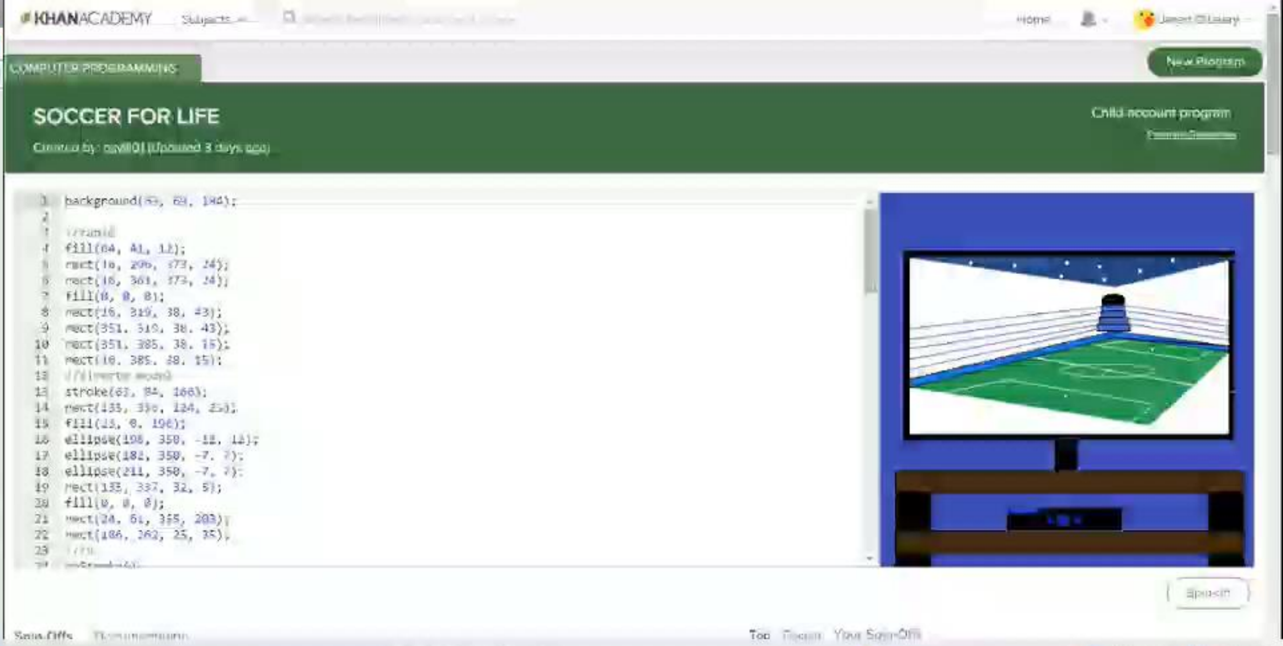
Experience Design Impacts Equity

How might we encourage equitable learning in our CS classes?



This code will make the car the starting position. When if this keypoint is pressed, the speed is below 7 and higher top speed and revers acceleration) If the car touc than 2, the car will slow dow green and is going backward moves forward or backward you have the car do?

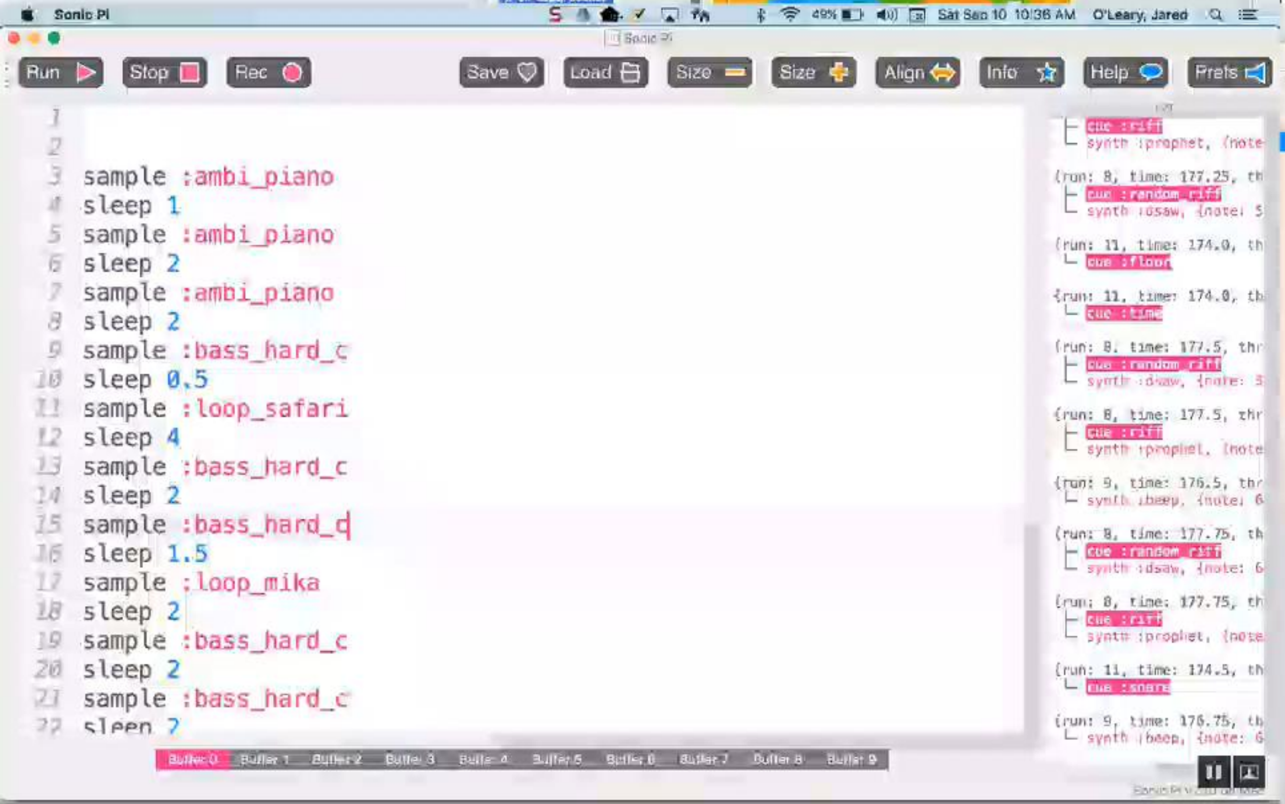
What other tracks ca What can you make Can you add in a fun



SOCCER FOR LIFE

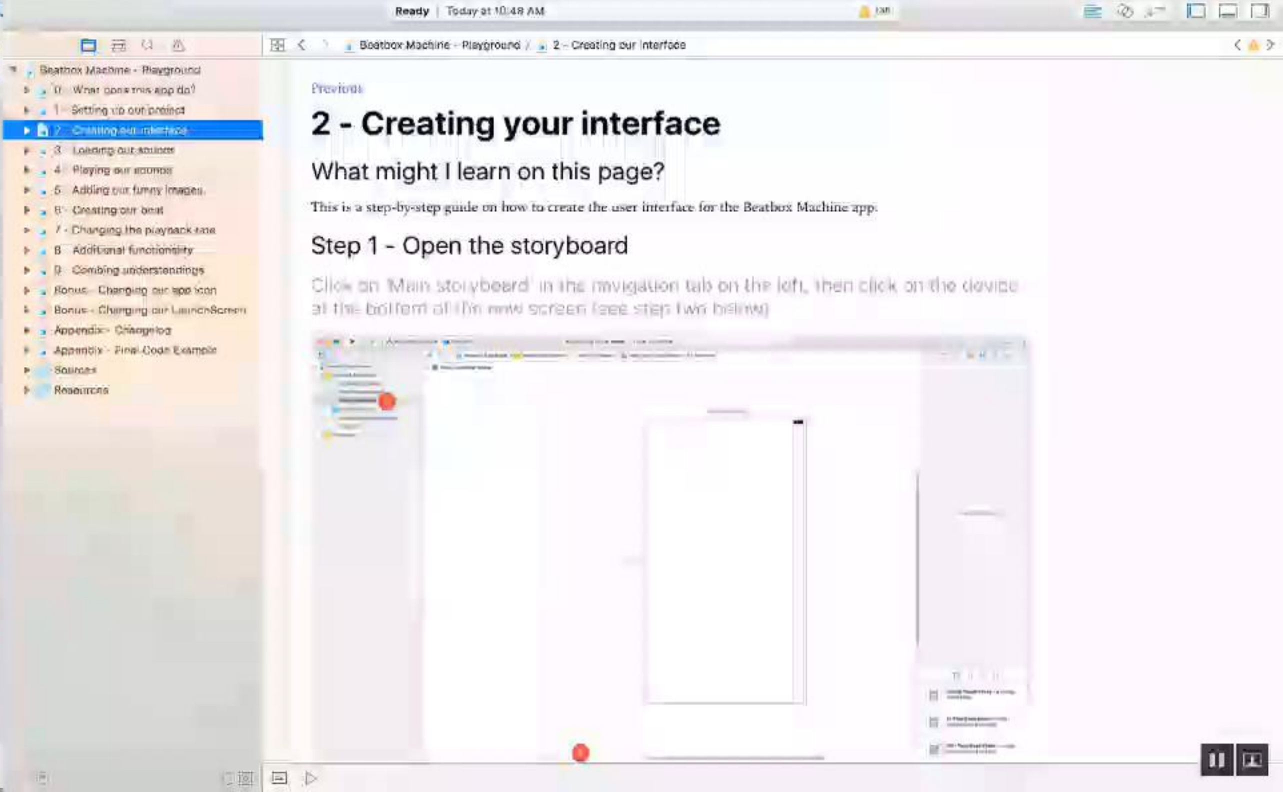
Created by: ravi801 (Updated 3 days ago)

```
background(63, 68, 194);

//stand
fill(64, 41, 12);
rect(16, 296, 373, 24);
rect(16, 361, 373, 24);
fill(0, 0, 0);
rect(16, 319, 38, 43);
rect(351, 319, 38, 43);
rect(351, 385, 38, 15);
rect(16, 385, 38, 15);
//directv model
stroke(67, 84, 166);
rect(135, 336, 124, 25);
fill(23, 0, 196);
ellipse(198, 350, -12, 12);
ellipse(182, 350, -7, 7);
ellipse(211, 350, -7, 7);
rect(133, 337, 32, 5);
fill(0, 0, 0);
rect(24, 61, 355, 203);
rect(186, 262, 25, 35);
//tv
```



```
sample :ambi_piano
sleep 1
sample :ambi_piano
sleep 2
sample :ambi_piano
sleep 2
sample :bass_hard_c
sleep 0.5
sample :loop_safari
sleep 4
sample :bass_hard_c
sleep 2
sample :bass_hard_c
sleep 1.5
sample :loop_mika
sleep 2
sample :bass_hard_c
sleep 2
sample :bass_hard_c
sleep 2
```

```
 ├ cue :riff
 └ synth :prophet, {note

{run: 8, time: 177.25, th
 ├ cue :random_riff
 └ synth :dsaw, {note: 5

{run: 11, time: 174.0, th
 └ cue :floor

{run: 11, time: 174.0, th
 └ cue :time

{run: 8, time: 177.5, thr
 ├ cue :random_riff
 └ synth :dsaw, {note: 5

{run: 8, time: 177.5, thr
 ├ cue :riff
 └ synth :prophet, {note

{run: 9, time: 176.5, thr
 └ synth :beep, {note: 6

{run: 8, time: 177.75, th
 ├ cue :random_riff
 └ synth :dsaw, {note: 6

{run: 8, time: 177.75, th
 ├ cue :riff
 └ synth :prophet, {note

{run: 11, time: 174.5, th
 └ cue :snare

{run: 9, time: 176.75, th
 └ synth :beep, {note: 6
```



Ready | Today at 10:48 AM
Beatbox Machine - Playground / 2 - Creating our interface
Beatbox Machine - Playground
0 - What does this app do?
1 - Setting up our project
2 - Creating our interface
3 - Loading our sounds
4 - Playing our sounds
5 - Adding our funny images
6 - Creating our beat
7 - Changing the playback rate
8 - Additional functionality
9 - Combing understandings
Bonus - Changing our app icon
Bonus - Changing our LaunchScreen
Appendix - Changelog
Appendix - Final Code Example
Sources
Resources
Previous

2 - Creating your interface

What might I learn on this page?

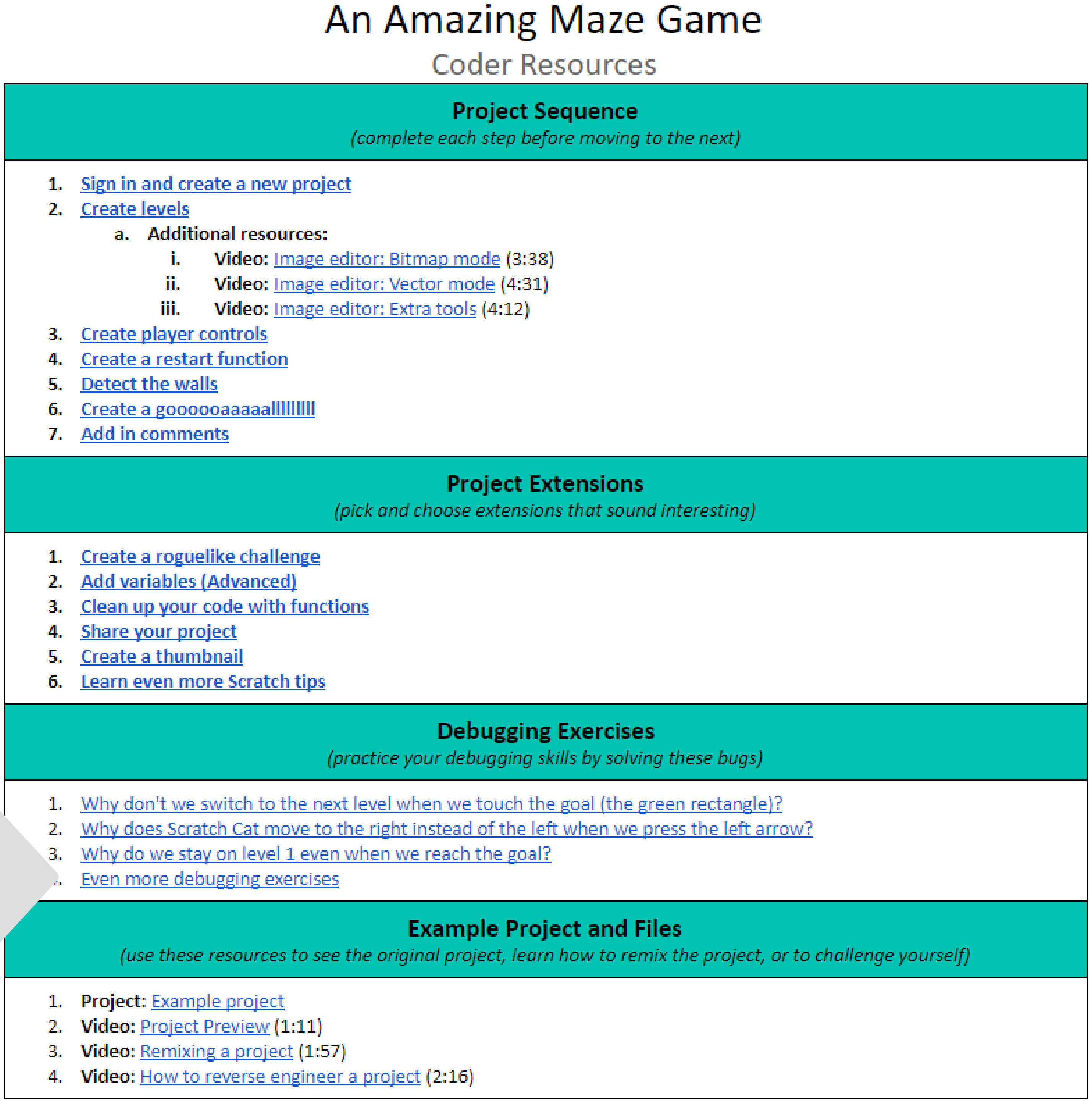
This is a step-by-step guide on how to create the user interface for the Beatbox Machine app.

Step 1 - Open the storyboard

Click on 'Main storyboard' in the navigation tab on the left, then click on the device at the bottom of the new screen (see step two below)

Applications in Curricula

Presentation by Jared O'Leary and uses Creative Commons licensing Attribution-NonCommercial-ShareAlike (BY-NC-SA)



An Amazing Maze Game

Coder Resources

Project Sequence

(complete each step before moving to the next)

1. Sign in and create a new project
2. Create levels
   a. **Additional resources:**
      i. **Video:** Image editor: Bitmap mode (3:38)
      ii. **Video:** Image editor: Vector mode (4:31)
      iii. **Video:** Image editor: Extra tools (4:12)
3. Create player controls
4. Create a restart function
5. Detect the walls
6. Create a goooooaaaaalllllllll
7. Add in comments

Project Extensions

(pick and choose extensions that sound interesting)

1. Create a roguelike challenge
2. Add variables (Advanced)
3. Clean up your code with functions
4. Share your project
5. Create a thumbnail
6. Learn even more Scratch tips

Debugging Exercises

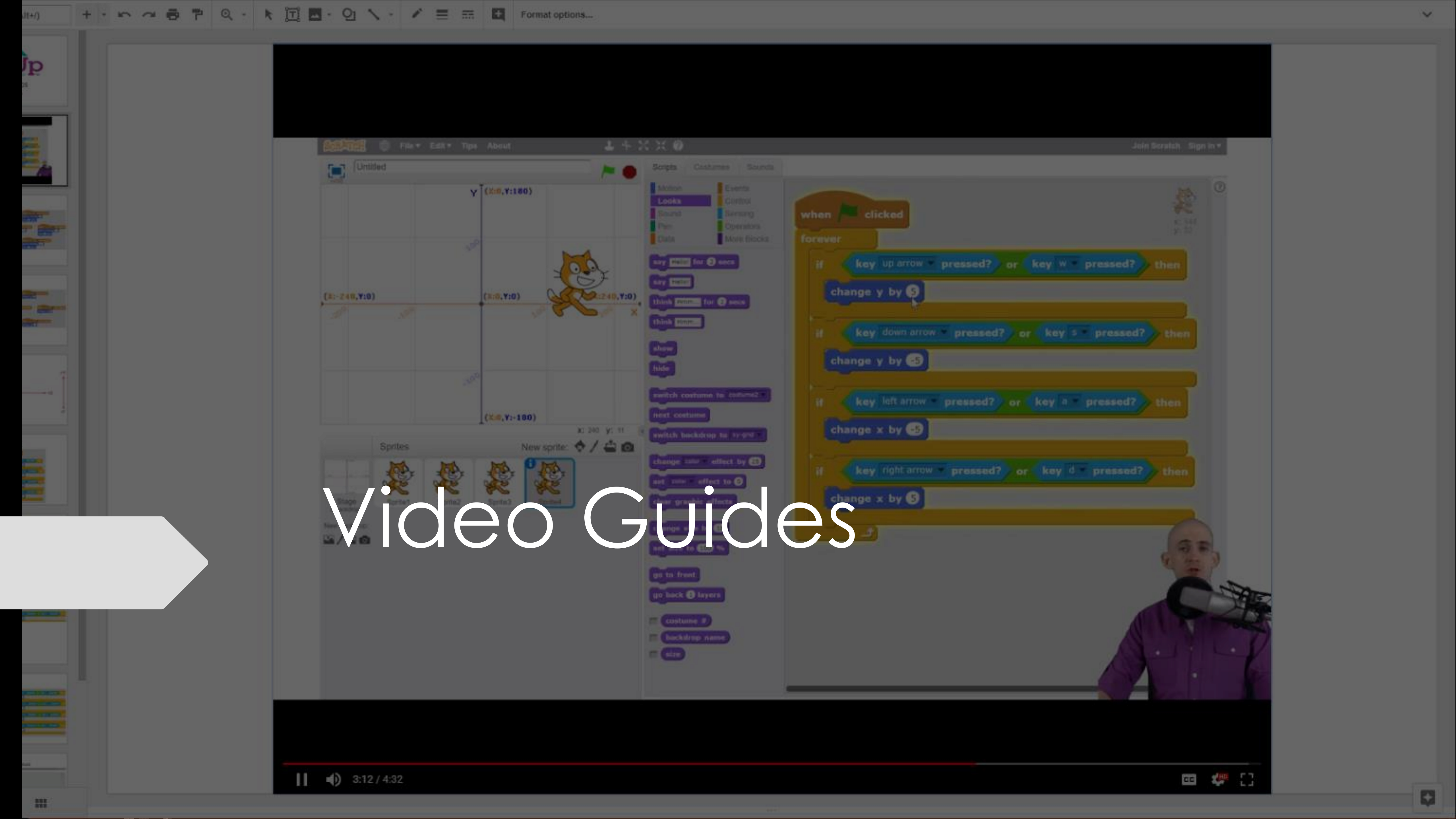
(practice your debugging skills by solving these bugs)

1. Why don't we switch to the next level when we touch the goal (the green rectangle)?
2. Why does Scratch Cat move to the right instead of the left when we press the left arrow?
3. Why do we stay on level 1 even when we reach the goal?
4. Even more debugging exercises

Example Project and Files

(use these resources to see the original project, learn how to remix the project, or to challenge yourself)

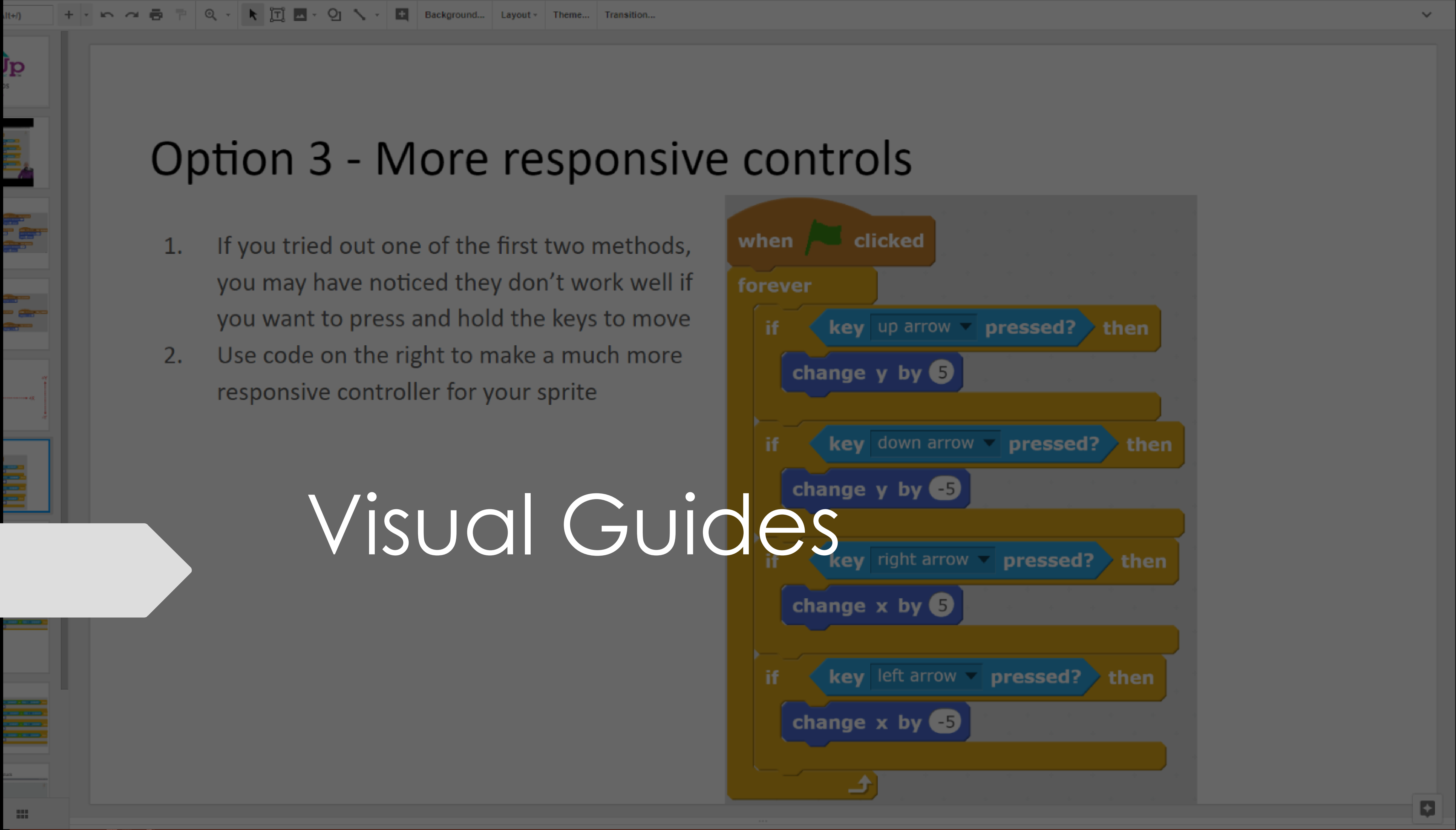
1. **Project:** Example project
2. **Video:** Project Preview (1:11)
3. **Video:** Remixing a project (1:57)
4. **Video:** How to reverse engineer a project (2:16)



Video Guides

```
when clicked
forever
    if key up arrow pressed? or key w pressed? then
        change y by 5
    if key down arrow pressed? or key s pressed? then
        change y by -5
    if key left arrow pressed? or key a pressed? then
        change x by -5
    if key right arrow pressed? or key d pressed? then
        change x by 5
```

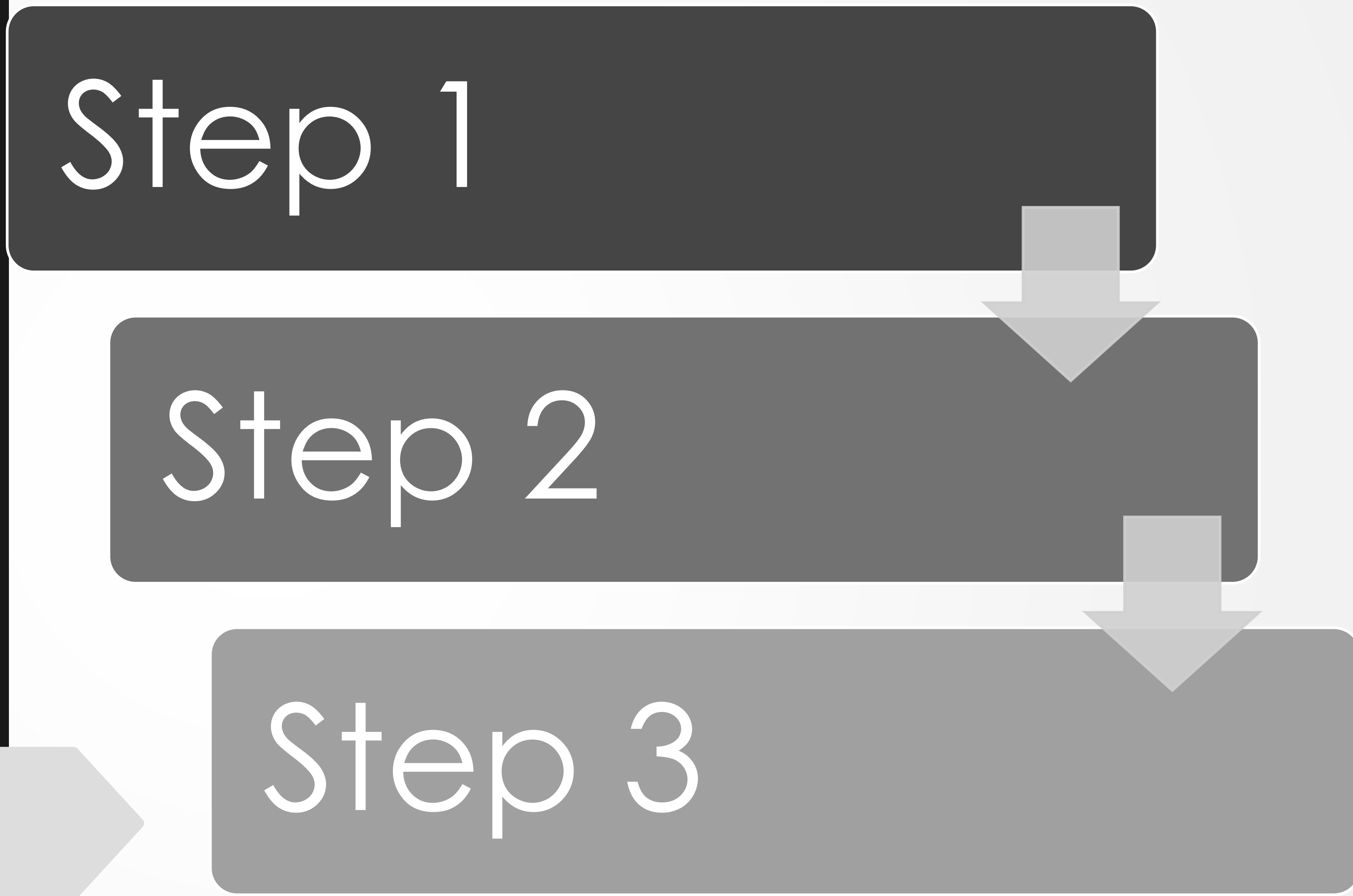
Visual Guides



Option 3 - More responsive controls

1. If you tried out one of the first two methods, you may have noticed they don't work well if you want to press and hold the keys to move
2. Use code on the right to make a much more responsive controller for your sprite

```
when [flag] clicked
forever
    if <key (up arrow) pressed?> then
        change y by (5)
    if <key (down arrow) pressed?> then
        change y by (-5)
    if <key (right arrow) pressed?> then
        change x by (5)
    if <key (left arrow) pressed?> then
        change x by (-5)
```



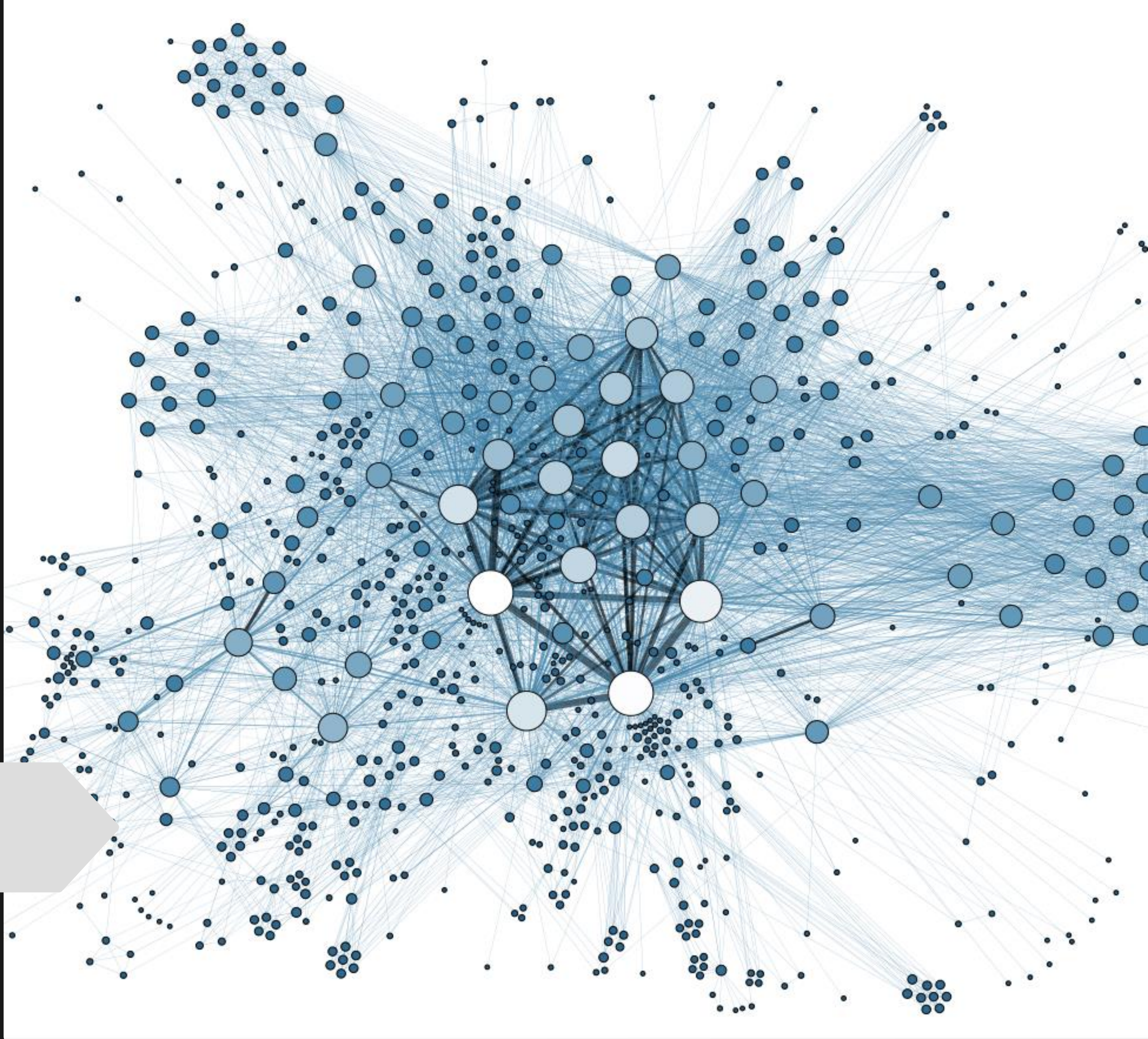
Step 1

Step 2

Step 3

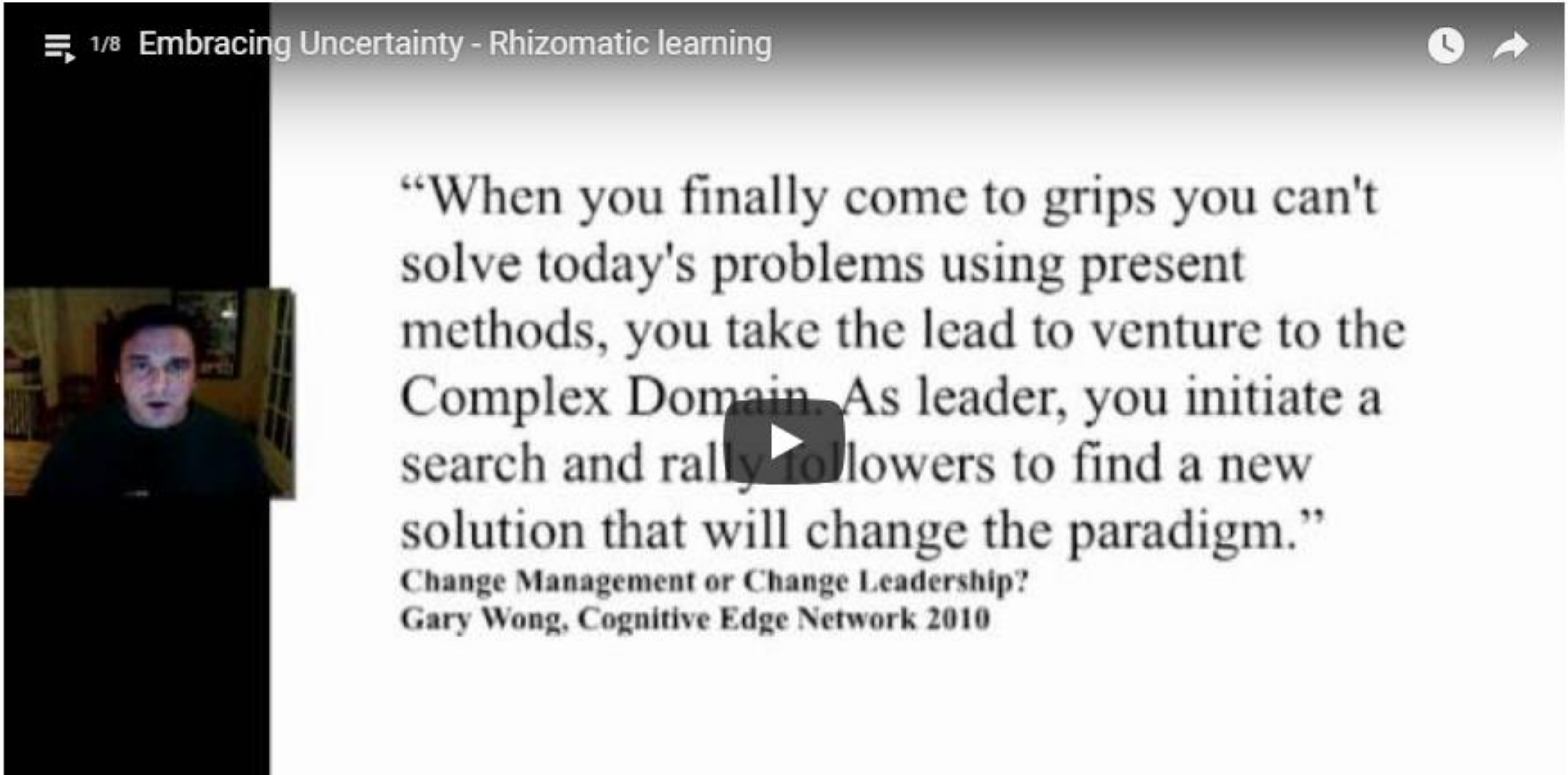
Sequential Design

Rhizomatic Design



Exploring Rhizomatic Learning

Relevant Lectures And Discussions



"When you finally come to grips you can't solve today's problems using present methods, you take the lead to venture to the Complex Domain. As leader, you initiate a search and rally followers to find a new solution that will change the paradigm."

Change Management or Change Leadership?
Gary Wong, Cognitive Edge Network 2010

Relevant Presentations

- Assessing Coding Projects
    - This session discusses how to seamlessly integrate formative, summative, and ipsative assessment practices within K-12 coding projects and lessons. We will explore each of the three approaches and I will provide both formal and informal examples of how each type of assessment might occur within a project or lesson.
- Facilitating Multiple Programming Languages in One Space
    - This lightning talk describes considerations for facilitating multiple programming languages in one space. I provide video examples of what it looks like when young coders select from four different programming languages to create projects of interest. Following an overview of what coders created in the classes I designed and facilitated, I discuss considerations for simultaneously facilitating multiple languages; this discussion includes quick suggestions for selecting and creating resources, questioning techniques, peer-to-peer mentoring, room setup, and more.
- Interest-driven Coding and Learning (ADE)
    - The video in this link is a mock version of a three minute showcase on interest-driven coding and learning I presented at the 2017 Apple Distinguished Educators (ADE) US Academy.
- Interest-driven Coding Projects (Scratch@MIT)
    - This ignite talk describes considerations for designing interest-driven coding projects with Scratch. I provide examples of what an interest-driven coding class looks like and how projects are designed for a variety of experience levels and interests within a

Relevant Publications

- **Publications by other educators and scholars**
    - Rhizomatic Education: Community as Curriculum - Dave Cormier
        - Introductory paragraph: "The increasingly transitory nature of what is lauded as current or accurate in new and developing fields, as well as the pace of change in Western culture more broadly, has made it difficult for society in general and education in particular to define what counts as knowledge. The existing educational model with its expert-centered pedagogical planning and publishing cycle is too static and prescribed to accommodate the kind of fluid, transitory conception of knowledge that is necessary to understand the simplest of Web-based concepts. The ephemeral nature of the Web and the rate at which cutting-edge knowledge about it and on it becomes obsolete disrupts the painstaking process by which knowledge has traditionally been codified. Traditional curricular domains are based on long-accepted knowledge, and the "experts" in those domains are easily identified by comparing their assertions with the canon of accepted thought (Banks 1993); newer concepts, whether in technology, physics, or modern culture, are not easily compared against any canon. This lack of a center of measurement for what is "true" or "right" makes the identification of key pieces of knowledge in any of these fields a precarious task. In less-traditional curricular domains then, knowledge creators are not accurately epitomized as traditional, formal, verified experts; rather, knowledge in these areas is created by a broad collection of knowers sharing in the construction and ongoing evolution of a given field. Knowledge becomes a negotiation (Farrell 2001)."
    - Rhizomatic Learning - Wikipedia
        - Introductory paragraph: "Rhizomatic learning is a variety of pedagogical practices informed by the work of Gilles Deleuze

CS Resources I Created And Used For Rhizomatic Learning

A free elementary coding curriculum I've developed that encourages rhizomatic learning

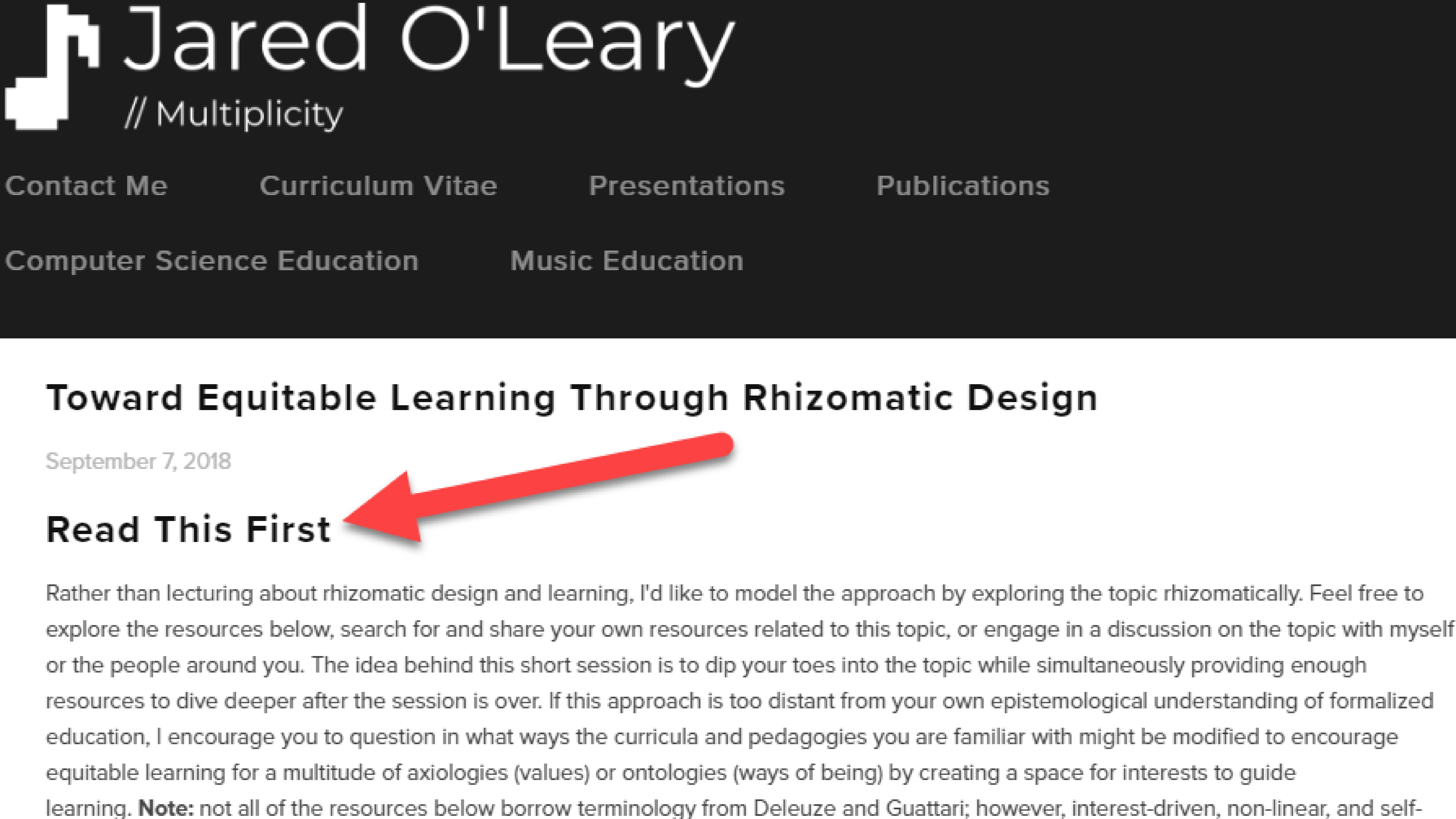
JavaScript Resources

Media Arts & Technology Makerspace

Scratch Resources

Sonic Pi Resources

Swift Resources



Toward Equitable Learning Through Rhizomatic Design

September 7, 2018

Read This First

Rather than lecturing about rhizomatic design and learning, I'd like to model the approach by exploring the topic rhizomatically. Feel free to explore the resources below, search for and share your own resources related to this topic, or engage in a discussion on the topic with myself or the people around you. The idea behind this short session is to dip your toes into the topic while simultaneously providing enough resources to dive deeper after the session is over. If this approach is too distant from your own epistemological understanding of formalized education, I encourage you to question in what ways the curricula and pedagogies you are familiar with might be modified to encourage equitable learning for a multitude of axiologies (values) or ontologies (ways of being) by creating a space for interests to guide learning. **Note:** not all of the resources below borrow terminology from Deleuze and Guattari; however, interest-driven, non-linear, and self-

goo.gl/qrwkYb

(case sensitive)