## WEB DESIGN

## HTML & CSS

### STRUCTURE & PRESENTATION

### Structure: HTML

content

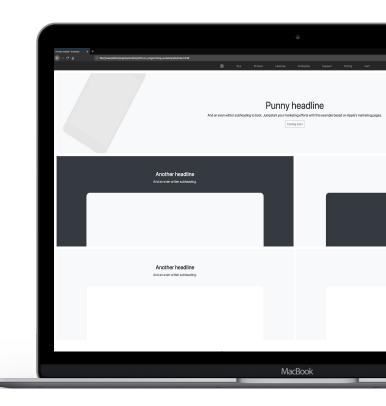
### Presentation: CSS

p.content { color:"red"; }

## GET THE CODE: https://bit.ly/2T4TF5Q

### Once you have the code:

- Open up the folder in a file explorer
- Open **site/index.html** in a browser
- Make a change to **product.css**
- **CMD+R** (Mac OSX) or **SHIFT+F5** (Windows) to refresh the page
- See your changes!



# SONIC PI

"I see software primarily as a form of communication that is not just limited to a conversation between a programmer and the computer, but between many participants, all of whom have an interest in understanding parts of the software."

- Dr. Sam Aaron

#### **Download and Run Sonic PI**

- First, go to:
  - https://sonic-pi.net
- Choose your platform
  - Download the installer
  - Follow the instructions
- Open Sonic Pi when finished
- https://sonic-pi.net/tutorial.html

```
Run 🕨 Stop 🔳 Save 💙 Rec 🔘
      use_sample_pack "Users/rbn/Desktop/samples/AKWF"
      sring = (ring :sinharm_0005,:granular_0043,:fmsynth_0050)
      snp=note(:d3)
      define :pl do |sname,n,d,r=1,sp=note(:d3) |
        sample sname, rate: pitch_to_ratio(note(n)-snp)*r, sustain: d*0.95, release: d*0.05, amp: 2, cutof
 10 with_fx :level do |amp|
       control amp.amp: 0 #set initial level
sleep 0.01 #allow initial level to settle without a click
         #three samples playing takes 30*3 ticks
        live_loop :b do
| with_fx :panslicer,wave: 1,pulse_width: 0.2 do
| with_fx :echo,mix: 0.4 do
                tick_set :sam, look/30
sn =sring.look(:sam)
notes=scale(:d3,:minor_pentatonic,num_octaves: 2)
pl(sn,notes.choose ,0.15) if spread(7,13).look
                 pl(sn,notes.look,0.15,2) if !spread(7,13).look
                 sleep 0.15
                 #puts "look b "+look.to_s
                stop if look >= 30*3*5+90 #stop after 3 samples played five times plus fade out time
         live_loop :audio do
           #puts "audio look "+look.to_s #uncomment for debugging purposes
           #increase volume over the first 90 ticks
control amp.amp: ( (look.to_f)/90) if look <= 90</pre>
          **Moderase volume after 5 complete cycles of 3 samples control amp,amp: (1- (tick(:stop).to_f)/98) if look >= 30*3*5 stop if look(:stop) >= 90 #stop loop after volume is zero (90 tick(:stop)) sleep 0.15
38 sto
39 sle
40 end
                                                          Buffer 0 Buffer 1 Buffer 2 Buffer 3 Buffer 4 Buffer 5 Buffer 6 Buffer 7 Buffer 8
                                                                                            MacBook
```

### Happy hacking!