0 -> red
60 -> yellow
120 -> green
180 -> cyan
240 -> indigo
300 -> magenta
360 -> red

0 -> red
120 -> green) 120

60 -> yellow
240 -> indigo) 180

R G B

triadic -> 1 2 3 | 3 1 2 | 2 3 1

Conversions on website bookmarked

colors
↓↓
↓

saturation
~~light~~

low ↓ high ↓ low — low ↓ med ↓ high

high med low — high low high

med med med — low low low

high high high — low low high

high high low

color schemes:
complimentary
analogous
triadic + saturation
split complimentary + lightness
tetradic
monochrome

light
↓
dark

↓

dark
↓
light

↓
same lightness throughout

↑
med
↓

light
↓
light

dark
↓
dark

->
permutation
low, med, high
-> — — —

* allow user input optional

___ ___ ___ ___ ___ ___ ___

↓

[hue, sat, light]

Saturation: ~~hard code~~ possible ~~fa~~ sets

low = ___

red = ___ → 7 diff levels

high = ___

___ ___ ___ ___ ___ ___ ___

↓

1 2 3 4 5 6 7

↓

auto/randomly populate:

5 4 2 2 1 3 7

↓

sort in order:

1 2 2 3 4 5 7

↓

apply to → [hue, sat, lit]

lightness →

highlight

light

light half tone

half tone

dark half tone

dark

dark accent

Color temperature adjustment

↓

micro adjust hue

↓

button?

↓

user input

↓ or comparators

↓

list out hues

↓

comparative warmth?

↓

against what?

↓

preset colors

very last step

complimentary → 2 colors → a, b

↙ in hues

a b a b a b a

b a b a b a b

↓

a, b → random values

→ for each space in array

→ compare, bigger gets put in

→ no need to permutate

↙ 50% chance of being selected

triadic: a, b, c

___ ___ ___ ___ ___ ___ ___

randomly populate array but at least one of each value

→ ~~flag for each value~~

→ after random population check if all true (included)

→ if not, randomly replace one with unincluded value

→ check again and repeat until all values included at least once

→ ~~once by~~ → includes()

(javascript)
code layout:

☒ HSL -> other colors
conversion
functions

some default
colors
+
saturation
+
lightness
constants

~~color array~~

color array (empty) (4 spots)
-> [Hue, saturation, lightness]
↑
each array item

input color (user input,
default random)

colorscheme array populating functions
monochromatic
greyscale
coplimentye
etc
etc

saturation array populating

lightness array populating

choose between
-> hard coded light -> dark
default
-> random population

actual function
color
↓
saturation
↓
lightness
↓
return array -> display it on HTML

micro adjustment functions

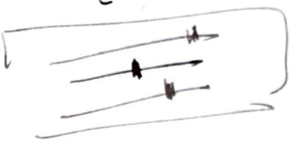
-> warmth

-> lightness

-> saturation

for each individual color, OR all at once

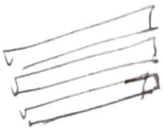
clickable



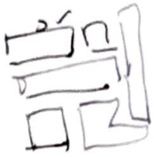
slider menu

different color display formats

or



or



tetris-y?

or

place colors next to each other for contrast

or in circles

or nested squares

just ~~one~~ two squares

-> allow user to choose what to display

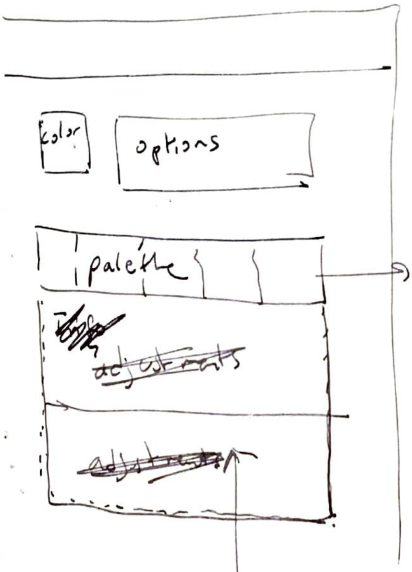
pastel, dark, mid, etc

adjust saturation + lightness values so they fall within a certain range

for art, keep in mind -> don't want pure black

-> add ~~to~~ limits to random saturation + lightness generation

in hardcoded constants



click on each color for info + adjustments