○ not started
⊖ in progress
⊕ Complete
? Stretch goals / unsure of implementation

Color Theory "Game"

Base Functions

⊕ Color Storage (singular)
  ⊕ Attributes RGB
⊕ Color Palette storage

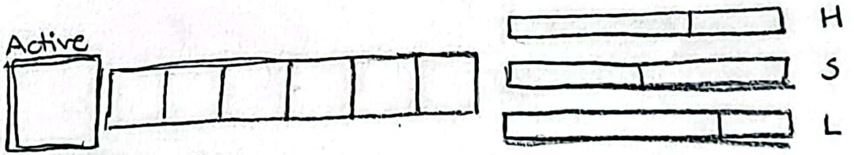
⊕ Adjusting color values
  ○ Adjusting palette

⊕ Draw simple objects of given color —— ⊕ Visuals for palette
  ○ update visuals @ user input (repaint)
  ○ Export palette ○

ArrayList ActiveColors <Color>
for Each ... colorAction()
repaint visuals()   Buttons/visuals

○ Select single color ○
○ Select color palette ○
○ Select multiple colors ○
?○ Combine colors $\left(\frac{r1+r2}{2}, \frac{g1+g2}{2}, \frac{b1+b2}{2}\right)$ ○
⊕ add lightness / darkness (all RGB) ○
⊕ increase / dec saturation   difference between low + high R, G, and B values ○
⊕ shift hue   convert HSV? ○
⊕ Warmth Red/Blue relationship ○

○ Color Analysis → ○ Comparison of two colors
  ⊕ Saturation   difference between lowest + highest RGB values
  ⊕ Value   adjust all RGB values at once
  ⊕ Hue   easiest in HSV space
  ⊖ Warmth   balance between Red and Blue



Stretch Goals / Extended Functions

○ Color in context, ask user to
  ○ choose 'most' or 'least' ____ object
  ○ choose color that best fits in context
  ☆○ adjust color to fit in context
  ○ choose complementory color / color palette questions
○ Color palette prompts, user can shift individual colors or entire palette until satisfied
○ Simple color wheel

Get values

⊕ Color
⊕ R
⊕ G
⊕ B
⊕ Hue
⊕ Saturation
⊕ Brightness

Scale values

Set Values w/ restriction of range

(0-255)
(0.0-1.0)

⊕ Color
⊕ R
⊕ G
⊕ B
⊕ Hue
⊕ Saturation
⊕ Brightness

Shift Values

⊕ R
⊕ G
⊕ B
test ⊕ Hue
test ⊕ Saturation
test ⊕ Brightness
⊕ Warmth
⊖ Value (RGB)

Select colors

*○ single active color
○ entire palette
○ active colors
int[] ArrayList (indexes)

(less priority)

Compare values

- return Color Object of greater value/ return index?
input Objects or index?

○ Hue
○ Saturation
○ Brightness
○ Warmth - closeness to arbitrary warm hue value (abs value diff) (255, 200, 0)
⊖ R
○ G
○ B

Wait to see needed function
define
○ Color (equals())

Scale values

⊕ R
⊕ G
⊕ B
○ ~~Hue~~ (
⊕ Saturation
⊕ Brightness
○

Visuals / Quality

○ appealing layout
*○ javadoc comments
○ Combine colors $\left(\frac{r_1+r_2+\dots+r_n}{n}, \frac{g_1+g_2+\dots g_n}{n}, \frac{b_1+b_2 \dots b_n}{n}\right)$

Alchemy
alchemy

○ Select colors to combine, slot to assign new color
○ Select initial spot, then other colors to add (can be weighted or restricted to certain attributes)

Strength of other colors to add: (out of 100)
Attributes to modify:
□ Hue
□ Saturation
□ Brightness

Future: Which color added to A most closely matches B?

Buttons / Controls

○ select Color(s)
⊖ decide which modifiers to show
⊖ decide visual layout
⊖ Buttons/incremental change or *text boxes for num values
easier start

When is it significant to use this.method()
as opposed to method() inside of a class?
is it only needed when accessing attributes?