Assignment 5 – Elm

due at 23:59 on Sun Apr 16 (100 points)

For this assignment, we will use the Elm language for client-side web development. Here are links to some documentation and other resources that I used in class:

- Documentation index – no need to install Elm locally unless you want to, but work through "Core Language" and "Elm Architecture" in the Get Started guide.
- Online editor – easy way to try out Elm without installing anything. Includes links to load some simple examples from the documentation.
- Online IDE – more sophisticated than the above, because it allows you to import additional modules from the package repository, and allows you to edit the HTML/CSS into which the Elm app is embedded.
- Online REPL – if you want to try out simple expressions.
- Convert HTML to Elm – can be a useful tool for starting the view function.
- Package repository – search and read documentation about packages, including essential built-in ones like core and html.

Using ELM, you should build a **color chooser** application with the following features:

1. Sliders for the red/green/blue components
2. Sliders for hue/saturation/lightness
3. A swatch (rectangle) that displays the chosen color
4. A **complement** button that rotates the hue by 180°.
5. It should display the RGB components in three formats:

- 8-bit base ten, such as: `rgb(108,255,21)`
- real numbers in the range 0-1, such as `(.422,.996,.082)`.
- 8-bit hexadecimal numbers, such as `#6CFF15`

6. It should display the Hue value in both degrees: `98°` and radians: `1.71`
7. It should display the saturation and lightness values as percentages, like `100%` or `54%`.
8. It should have a drop-down input of ten or more **named** color values that can be loaded: refer to X11 color names for a long list of possibilities.

I have provided a starting point: see `elm-demo/rgb-chooser.elm` in the repository. You can copy/paste that code into the online editor to try it out and modify it. Be careful though, that editor doesn't save your work – you have to copy/paste back into an editor on your system to keep your changes. (The "Ellie" IDE does save changes, you would just need to keep the URL your code is assigned.) My starter code relies on the core Color module in Elm. You may also want to look into the Color.Convert module (but it requires installing the `elm-color-extra` package, which is possible with "Ellie").

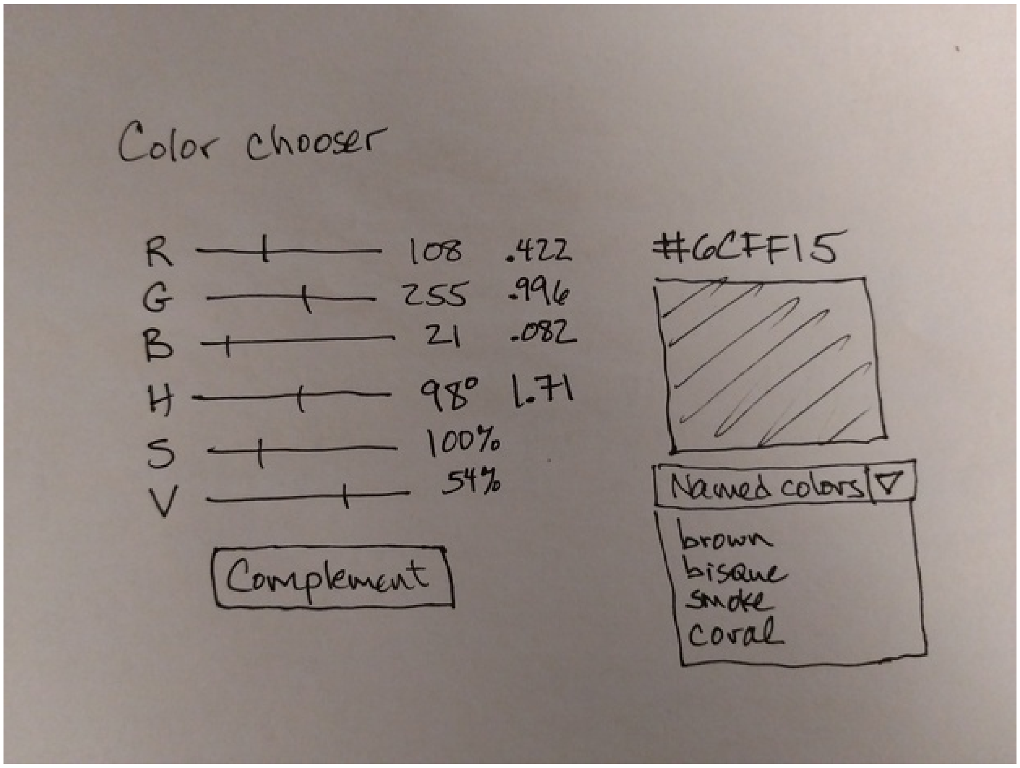

Figure 1:

Here's a mock-up of what it could look like if you design the layout carefully:

To submit, save your code in a file named `colors.elm` in your repository, then commit/push.