

Useful Resources:

```
ControlP5  
red()  
green()  
blue()  
hue()  
saturation()  
brightness()
```

Instructions:

Please download the ControlP5 library and import it to each file created before running. This can be done by clicking Sketch -> Import Library -> Add Library, scrolling down to ControlP5, selecting it, and clicking install. Once this is done it should appear under the Contributed section of Sketch-> Import Library. When ControlP5 is selected here, it should add some lines of code to the beginning of your file.

All parts of lab questions that have **R** at the beginning are mandatory, and their submission by each student is required. The parts of each question that begin with **O** are open-ended enhancements for further investigation into course material. These should be completed only after the mandatory parts of the lab are completed, and are not explicitly required for submission. If you choose to complete further optional add-ons, you are welcome to explore your own options, and not just the suggestions provided. Submit a zip document to the dropbox folder L0 on LEARN. The document should contain separate folders for the following files:

- **L2_ColourSlider**
- **L2_Radio**

Lecturette: Basic ControlP5 functions, found at <http://www.kasperkamperman.com/blog/processing-code/controlp5-library-example1/>

1. ColourSlider

- R** Using the ControlP5 library, create a sketch that uses 3 sliders, named “red”, “green”, and “blue”, that change the colour of the background using the RGB colour mode.
- R** Add a feature so that you can also change the colour of the background using HSB mode as well (with a total of 6 sliders on screen). To change between the two, create an on-screen toggle that will be set to either HSB or RGB mode. Depending on the value of this toggle, only one set of sliders will work at any given time.

- c. **O** Add a final feature that allows the sliders of the one colour mode to update, in real time, the sliders of the other colour mode. For example, moving the blue slider of the RGB mode would update the hue, saturation, and brightness sliders of the HSB mode.

2. Radio

Note that for the purpose of this lab, the functions we ask you to recreate do not mimic how actual radio waves are expressed.

- a. **R** Create a sketch, using the ControlP5 library, which acts as the visual representation of a radio. First, implement a toggle that acts as an on/off switch. When the toggle is 'on', the Boolean value associated with it is true. Use the starter code `L2_RadioStarterCode` (adapted from the version at processing.org/examples/sinewave.html), to make it so that if the on/off toggle is switched on, the bottom half of your canvas will be taken up by the sine wave. If the toggle is switched off (false), the sine wave disappears.
- b. **R** Add a slider to your sketch that acts as a tuner of an FM radio dial. Program the slider to go from 88 to 108 (the available FM station frequencies, in Megahertz, or MHz). Set the frequency of the sine wave (the float variable `frequency`) to change depending on the position of the slider.
- c. **R** Finally, add a volume knob to your file, which controls a float variable called `amplitude`. Make the knob go from 0 to 10, which controls the amplitude, the height of the wave. Map the minimum amplitude to a small positive number, so that the wave never goes completely flat; map the maximum amplitude so that the wave doesn't quite reach the boundary of the readout area.