

# CS106 W2020 - Lab 01

Due: Wednesday, January 8, 2020, 11:59 PM

This lab is a recap of topics covered in CS105.

Starter code for this lab is available in L01\_StarterCode.zip.

This lab is in two parts. The first part is a recap of the “Processing IDE” and its features. The second part is coding recap exercises in “JavaScript p5”.

---

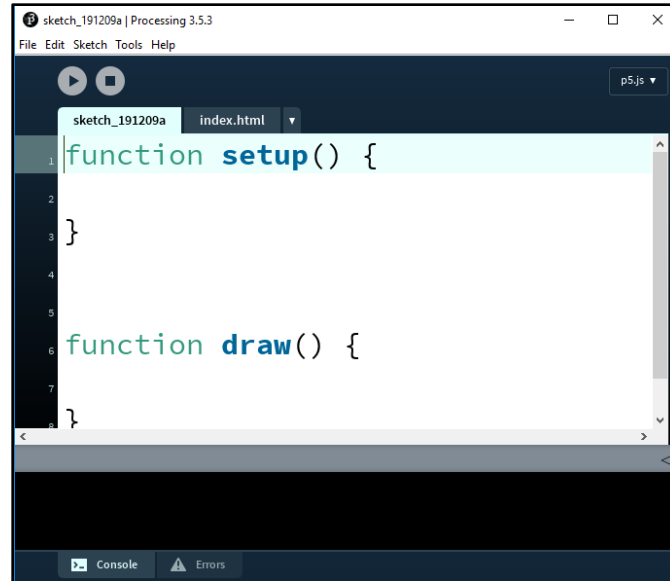
## **PART ONE: Recap of the Processing IDE**

1. Log in to one of the Mac computers in your lab room. That's it, just make sure you can access your account. You might be planning to use your own laptop all term, but laptops crash, and we want to make sure you always have access to the lab as a backup. (For that matter, now is a good time to ensure you have a good backup strategy in place. Consider saving your work into a folder that's automatically cloned by Google Drive or Dropbox.)

To demonstrate that you have successfully logged in to your account, do a screen capture and paste that screen capture into a file called “L01\_Q1LabLogin”. Save that file as a pdf file (L01\_Q1LabLogin.pdf).

2. In a Mac computer in the lab, launch the Processing IDE and make sure it runs. A screen shot of the Processing IDE is shown below. You must use p5.js mode by selecting it in the upper right corner (it may already be your default).

To demonstrate that you have successfully loaded the Processing IDE in your account, do a screen capture of the Processing IDE and paste that screen capture into a file called “L01\_Q2IDE”. Save that file as a pdf file (L01\_Q2IDE.pdf). It should look like the image below.



3. For the remainder of the Lab, you may use the lab computer or your laptop. Create a directory with the name “CS106\_L01”. Copy the following files into this directory:

- L01\_Q1LabLogin.pdf
- L01\_Q2IDE.pdf

Type the following code into the Processing IDE and run it. You should see a red ball animating across the screen. When the mouse is pressed, the ball starts again from the left-hand side of the canvas. Do not modify the code, except add a comment line at the beginning (i.e. use //) with your name and student ID number.

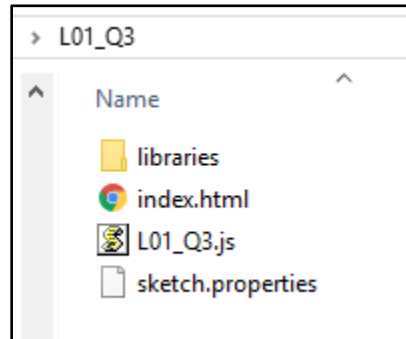
```
let ballX = 0;
let s = 20;

function setup() {
  createCanvas(200, 100);
}

function draw() {
  background(220);
  fill(255, 0, 0);
  ellipse(ballX, height / 2, s, s);
  ballX = ballX + 1.0;
  print(ballX);
}

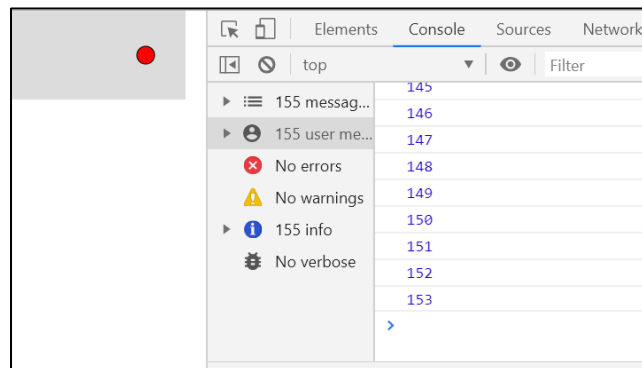
function mousePressed() {
  ballX = 0;
}
```

Save this sketch as “L01\_Q3”, within the directory “CS106\_L01”. This will create a directory “L01\_Q3” within “CS106\_L01” with the following files and directory.



You can use your browser’s debugger to help debug your code. To get into your debugger (also called the developer console), first run your program above. Then, to open the developer console window on Chrome, use the keyboard shortcut Ctrl Shift J (on Windows) or Ctrl Option J (on Mac). Alternatively, you can use the Chrome menu in the browser window, select the option "More Tools," and then select "Developer Tools."

You should see something like the following.

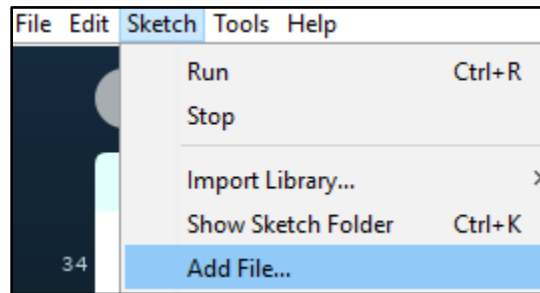


To demonstrate that you have successfully invoked the debugger, do a screen capture of the debugging console and paste that screen capture into a file called “L01\_Q3Debugger”. Save that file as a pdf file (L01\_Q3Debugger.pdf) in the directory CS106\_L01. The image you save should look similar to the image above.

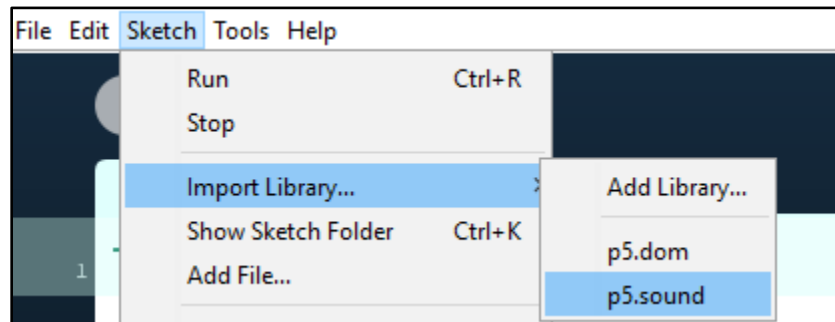
4. In CS105 we saw a simple JavaScript p5 program that uses sound. We are using that code as starter code in this question. The Starter Code for this question is located in a directory “L01\_Q3Sound”, and contains a text file with the JavaScript code, and two sound files.

Open the text file in a text editor such as “notepad” on Windows or “TextEdit” on a Mac. Cut and paste the JavaScript p5 code from the text file to the Processing IDE. (It won’t run without errors just yet.)

To use sound files in the Processing IDE you must load each sound file by selecting “Sketch” and then “Add File”, and then navigating to the sound file to add it to the sketch. Add the sound files “honk.wav” and “horn.wav”. A new directory will be created in your sketch folder called “data” to contain the two sound files.



To use sound in the Processing IDE you must import the sound library by selecting “Sketch”, then “Import Library”, and then “p5.sound”.



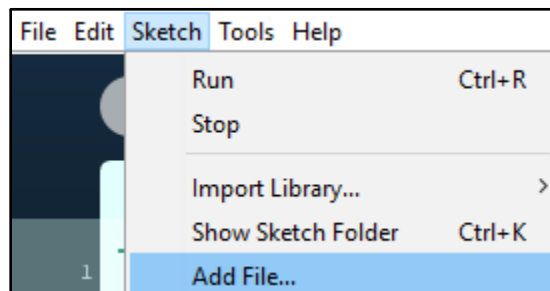
The code should now run and you should hear the sounds when you mouse click on the right and left sides of the canvas. Do not modify the code, except add a comment line at the beginning (i.e. use //) with your name and student ID number.

Save your sketch as “L01\_Q4”, within the directory “CS106\_L01”.

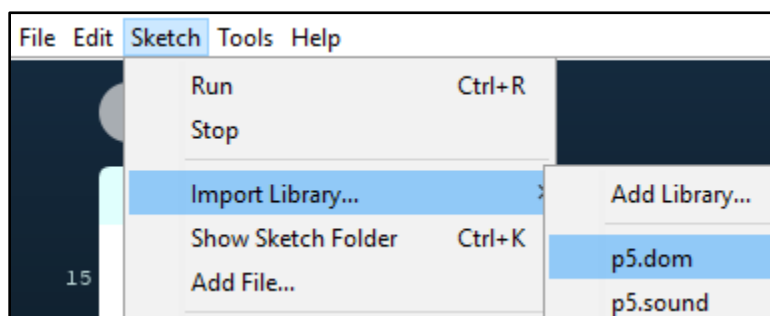
5. In CS105 we saw a simple JavaScript p5 program that uses video. We are using a version of that code as Starter Code in this question. The Starter Code for this question is located in a directory “L01\_Q5Video”, and contains a text file with the JavaScript code, and a video file.

Open the text file in a text editor such as “notepad” on Windows or “TextEdit” on a Mac. Copy the JavaScript p5 code from the text file to the Processing IDE. (It won’t run without errors just yet.)

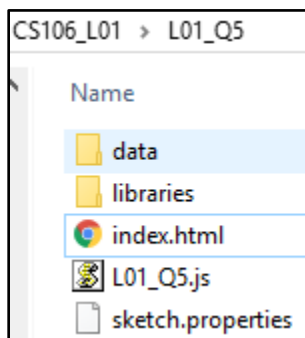
To use video files in the Processing IDE you must load each video file by selecting “Sketch” and then “Add File”, and then navigating to the video file to add it to the sketch. Add “flyboard.mp3” to your sketch. A new directory will be created in your sketch folder called “data” to contain the video file.



To use video in the Processing IDE you must import the “p5.dom” library by selecting “Sketch”, then “Import Library”, and then “p5.dom”.



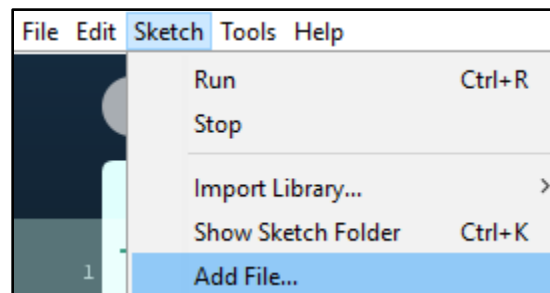
The code should run and you should see and hear the video (you need to press the mouse to start or pause the video). Do not modify the code, except add a comment line at the beginning (i.e. use //) with your name and student ID number. Save your sketch as “L01\_Q5”, within the directory “CS106\_L01”. The contents of the directory should be as follows:



6. In CS105 we saw a JavaScript p5 program that does a walk cycle animation. We are using a variation of that code as Starter Code in this question. The Starter Code for this question is located in a directory “L01\_Q6Animation”, and contains a text file with the JavaScript code, and 8 walk cycle images.

Open the text file in a text editor. Copy the JavaScript p5 code from the text file to the Processing IDE. (It won't run without errors just yet.)

To use image files for the animation in the Processing IDE you must load each image file by selecting “sketch” and then “Add File”, and then navigating to each image file to add it to the sketch. Add the 8 image files.

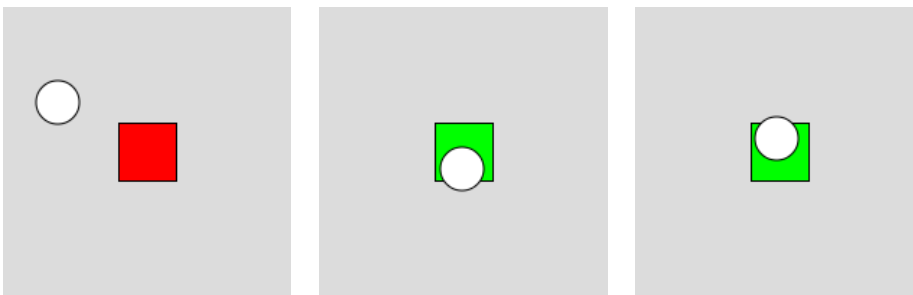


The code should run and as you press the right arrow key you should see the animation. Do not modify the code, except add a comment line at the beginning (i.e. use //) with your name and student ID number. Save your sketch as “L01\_Q6”, within the directory “CS106\_L01”.

## PART TWO: Recap of JavaScript

7. In CS105 you programmed many hit tests. Create a JavaScript p5 sketch that contains a hit test as follows (there is no Starter Code):
  - The canvas is 200 x 200.
  - The sketch has a red rectangle (40x40) in the middle of the canvas.
  - A white ellipse is centred at the mouse position mouseX and mouseY. The ellipse is a circle with diameter of 20 pixels.
  - When the centre of the ellipse hits the rectangle, the rectangle turns green.

Sample screen shots are shown below.



Save your sketch as "L01\_Q7", within the directory "CS106\_L01".

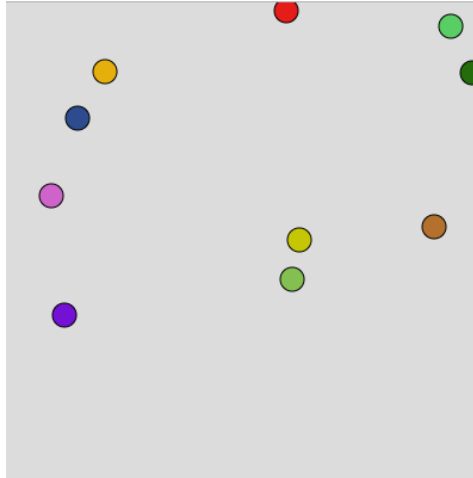
8. The Starter Code for this question is located in a directory "L01\_Q8Jitter", and contains a text file with the JavaScript Starter Code.

Open the text file in a text editor. Copy the JavaScript p5 code from the text file to the Processing IDE. (It won't run without errors just yet.)

The starter code Lab01\_Q8 produces ten circles on the canvas with each having a random location and a random color. Arrays are used to keep track of the position and color of each circle.

A video of a sample solution is shown at: [https://youtu.be/IZcOk\\_6Zf4U](https://youtu.be/IZcOk_6Zf4U)

Variations of this code are sometimes referred to as "Jitterbug".



- a) Add a `mousePressed()` function that will add one circle with each mouse press. After one mouse press there will be 11 circles, after two mouse presses there will be 12 circles, and so on.
- b) Modify the code so that each circle moves around by random amount on each frame. You can experiment with the amount of movement (i.e. jitter), but start with a movement of `random(-2, 2)` in each of the X and Y directions.

Save this program as “L01\_Q8”, within the directory “CS106\_L01”.

## Submitting

Submit all sketch directories from this lab as one ZIP file called `CS106_L01.zip` to the lab dropbox on LEARN. Consult “How to Submit: Handing in Code for Labs and Assignments” on LEARN for more information how to create a ZIP.

It is your responsibility to submit to the correct dropbox with the correct files before the deadline. Otherwise you will receive a mark of 0.