CS106 W2021 - Lab 02

Due Tuesday January 19 at 11:59pm.

---

1) Write a function repeat() that takes two values as input: a string and an integer that's at least 1. It returns a new string in which the provided string has been repeated the given number of times. So, for example, repeat("cat", 5) would return "catcatcatcatcat".

   To begin, fork the following sketch on Open Processing: "L02_Q1_StarterCode" (https://openprocessing.org/sketch/1061535#code).

   A video demonstration is at: https://vault.cs.uwaterloo.ca/s/TCBHcpieQEpHaFQ

2) Write a function isIncreasing() that takes an array of integers as input and returns a boolean value. The function checks if the array of integers is strictly increasing: that is, if each number in the array is strictly larger than the one before it. For example, the array [-3, 4, 9] is strictly increasing, but [1, 4, 2] and [3, 5, 5, 6] are not.

   To begin, fork the following sketch on Open Processing: "L02_Q2_StarterCode" (https://www.openprocessing.org/sketch/1061670#code).

   A video demonstration is at: https://vault.cs.uwaterloo.ca/s/SAKcfjt25YmynqE

3) Write a function countOccurrences() that takes two values as input: an array of strings and a single search string in a separate parameter. The function returns the number of times that the search string occurs in the array. For example, if arr holds the array ["apple", "pear", "peach", "apple", "apple"], then countOccurrences(arr, "apple") would return 3, and countOccurrences(arr, "durian") would return 0.

   To begin, fork the following sketch on Open Processing: "L02_Q3_StarterCode" (https://openprocessing.org/sketch/1061682#code).

   A video demonstration is at: https://vault.cs.uwaterloo.ca/s/6HMegWJSeZxikr5

4) In CS105 we used the sound file "honk.wav". Write a sketch that has the "honk.wav" sound play every three seconds (i.e. every 3,000 milliseconds). Remember we did this by using the millis() function in JavaScript p5. We first create a variable called startTime or something like that. Then we assign the current milliseconds to "startTime" by using the milliseconds function millis(). Then every time through the function "draw()" we check to see if millis() minus startTime is greater than 3,000. If not, we don't do anything. If so, we play the honk sound and reset startTime to be the current millis().

   There is no starter code. The sound file is in a zip file on LEARN.

   A video demonstration is at: https://vault.cs.uwaterloo.ca/s/AtMWk8WjNyX2KPg

Note that in Open Processing you must use the menu to enable the JavaScript p5.sound library. This is shown in the video above. This is also shown near the bottom of the graphic below.
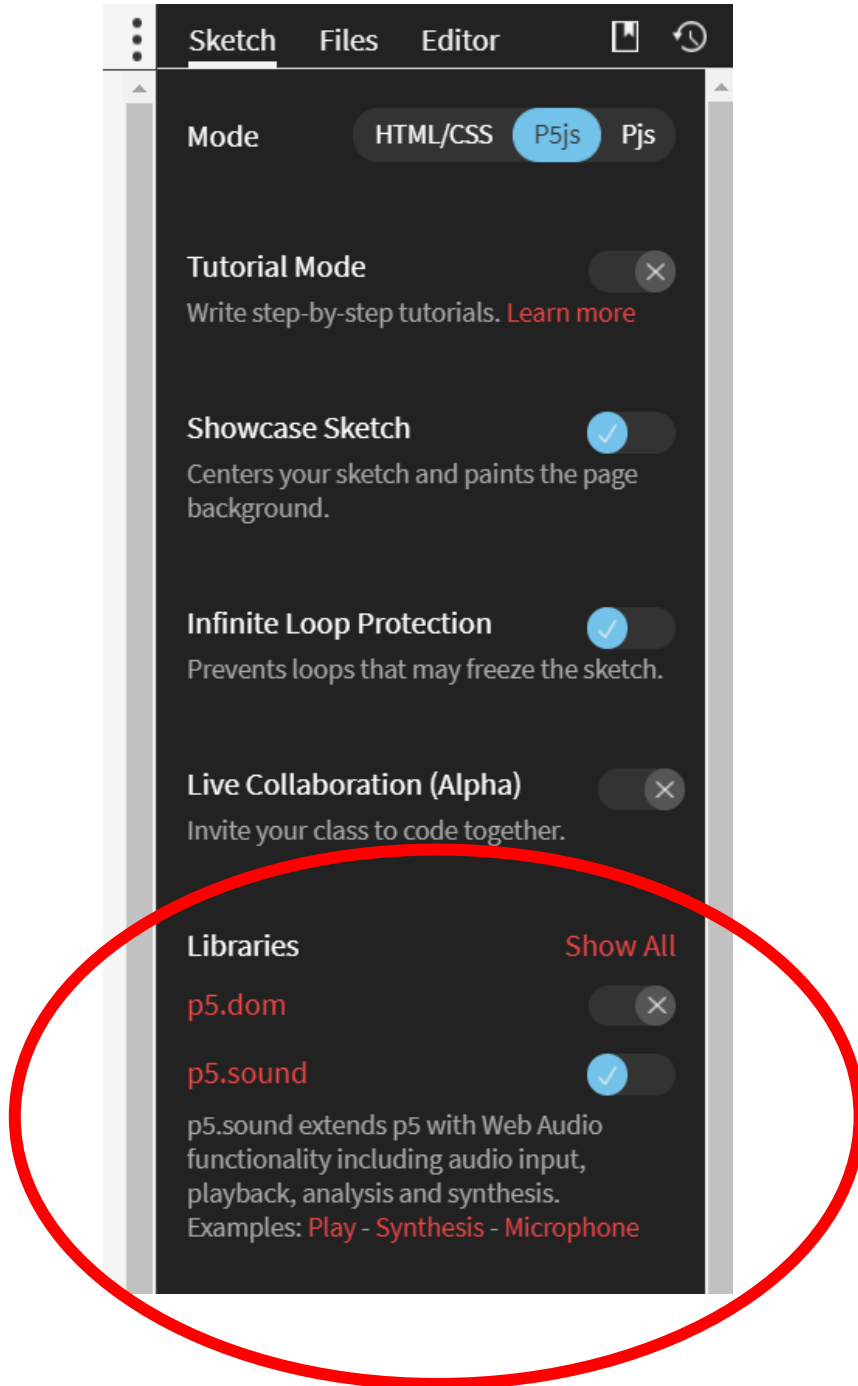
5) You are to mimic the behavior in the following video:
   https://vault.cs.uwaterloo.ca/s/TtERt6fdcyfer38

   There is an array. Each element in the array is either a "C" or a "S". If it is a "C", a circle is drawn, and a "C" is drawn on top. If it is an "S", a square is drawn, and the letter "S" is shown on top. The size of the circles and squares varies to fill up the canvas from left to right.

   To begin, fork the following sketch on Open Processing: "L02_Q5_StarterCode" (https://openprocessing.org/sketch/1061730#code).

Submitting

Use the template file in Word "CS106 Lab Template" in LEARN to create your Lab 02 submission.

Then convert your Word file to pdf. **Please ensure that your URLs are hot links. The TAs need to be able to click on each link in your pdf and go directly to your sketch.**

So for example, don't have a link like this: https://openprocessing.org/sketch/1050954

but rather have that link as a hot link as follows: https://openprocessing.org/sketch/1050954

Ensure that each URL you submit has its settings so that the access is as follows:
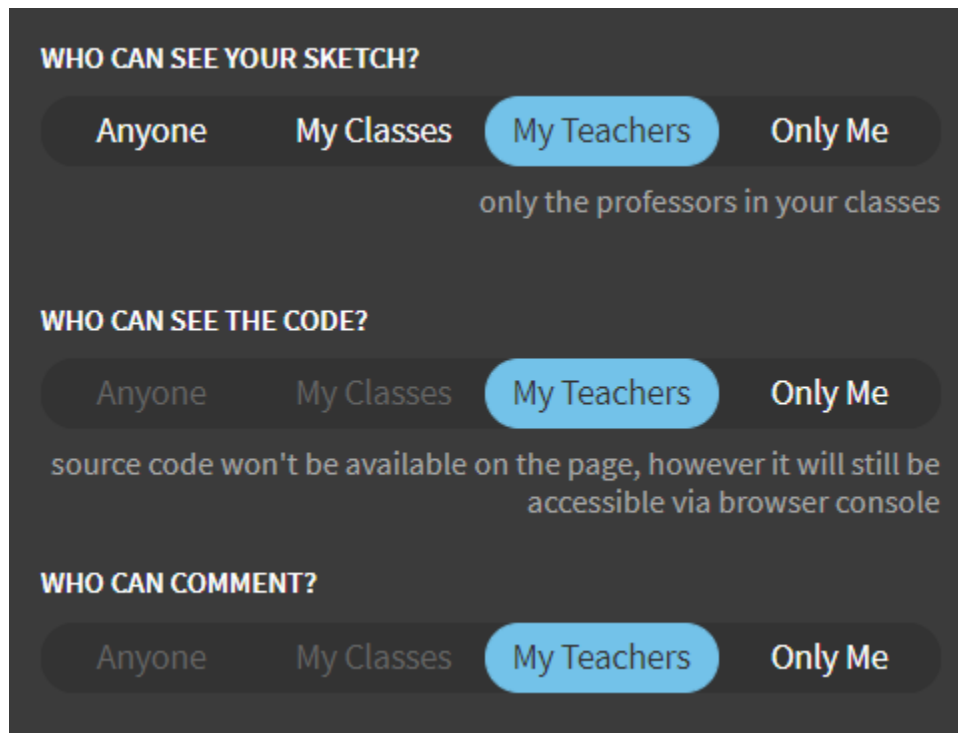
Submit that pdf file to the Lab 01 dropbox on LEARN.

An example of how to do submit a Lab is shown in the following video:
https://vault.cs.uwaterloo.ca/s/9Xx7AGsewaea773

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