

Programming Assignment #1

Due Sep 24 by 11:59pm **Points** 4 **Submitting** a text entry box or a file upload
Available until Sep 29 at 11:59pm

This assignment was locked Sep 29 at 11:59pm.

Due: Friday, September 24th (at midnight).

Late Policy: Assignment submission will remain open for 5 additional calendar days after the formal deadline. Late submissions will be penalized by a maximum of 1 point on our 4-point scale (i.e. if you submit 5 days late and your assignment was otherwise deserving a "3", you might get a grade as low as a "2" due to this penalty). Some leniency might be exercised on this general rule based on how late the assignment was, and whether turning in an assignment was a repeating occurrence or a one-time event.

Synopsis: You need to create a program that draws a picture on a 2D Canvas on a Web page. Your submission must demonstrate the use of lines and polygons, and the use of slider elements for input.

Learning Objectives: To make sure everyone understands how to make pictures with Canvas, using the most basic features that we reviewed in class. We also want to make sure that you can turn in assignments, and that you know how to create an HTML page with JavaScript (as a stand-alone file or files; not within a pastebin like JSbin).

Evaluation: Based on our 4-point grading scheme, as discussed in our introductory lecture. You get a check ("3") if you turn in a viable, and complete submission (even if it just draws a rectangle like the example in the tutorial). "Above and beyond" grades (i.e. a "4") will be awarded for people who have crafted something particularly cool. As a general rule, no more than 1/3 of all assignments turned in (the very best ones, that is) will be considered for a "4" grade.

Collaboration policy: This is an assignment to be done individually. Code not written by you needs to include proper attribution (see [this post](https://graphics.cs.wisc.edu/WP/cs559-fall2020/2020/09/10/collaboration-policy/) [\(https://graphics.cs.wisc.edu/WP/cs559-fall2020/2020/09/10/collaboration-policy/\)](https://graphics.cs.wisc.edu/WP/cs559-fall2020/2020/09/10/collaboration-policy/) here). It is always ok to use code provided in our in-class examples as a starting point, but you need to add your own effort to raise those examples (or other sources) to what is asked by the programming assignment (i.e. finding some code on some online forum that does all the job for you that is needed to satisfy the assignment is not the intent, if you haven't added any of your own effort to it). If you use somebody else's code (other than our GitHub examples), make sure to clarify in your submission notes what **you** did, and what you repurposed from the external source.

Hand-in: Electronic turn-in on Canvas. Make sure that you turn in all files needed for your program to run. It is acceptable to turn in a single HTML file with your program, but even preferable to separate your code into an .html file and a separate .js file containing the JavaScript code, similar to the examples in our [GitHub repository](https://github.com/sifakis/CS559F21_Demos) [\(https://github.com/sifakis/CS559F21_Demos\)](https://github.com/sifakis/CS559F21_Demos) (see, e.g. Demos

0-2 from Week 2). If you submit anything else than a single HTML file, please put everything in a single ZIP archive. ***It is not acceptable to submit a link to JSbin for this assignment!*** For this assignment, we discourage you from using any libraries. But if you do use a library, make clear to mention it in the comment box.

Description

In our recent lecture (in particular, on September 8th) we saw a few introductory examples of using the HTML5 canvas for simple drawing in 2D. Those include a simple demonstration of creating a canvas element and drawing a single colored line [\[JSbin link\]](https://jsbin.com/bovuhok) [_\(https://jsbin.com/bovuhok\)_](https://jsbin.com/bovuhok), a demonstration of the use of function calls, and filled polygons [\[JSbin link\]](https://jsbin.com/suhujar) [_\(https://jsbin.com/suhujar\)_](https://jsbin.com/suhujar), and an introduction to the use of sliders as input elements along with their associated callback mechanisms [\[JSbin link\]](https://jsbin.com/fesukexori) [_\(https://jsbin.com/fesukexori\)_](https://jsbin.com/fesukexori). Incidentally, you can find these same examples in our [GitHub repository](https://github.com/sifakis/CS559F21_Demos) [_\(https://github.com/sifakis/CS559F21_Demos\)](https://github.com/sifakis/CS559F21_Demos) (under the subdirectory Week2/; look for the subdirectories Demo0, ..., Demo2 under it) more properly written as .html files and .js files containing the HTML/JavaScript code respectively.

The goal of this assignment is to have you figure out the programming aspects of drawing pictures on web pages using the JavaScript 2D Canvas library. You should create a new program (as a single HTML file, or even better a separate .html file and a .js file with the JavaScript code, as exemplified in the demos from the GitHub repository) that includes – at minimum – the following visual elements:

- You should draw at least one **filled** shape (e.g. a polygon).
- You should either have multiple, separate filled shapes (not merely identical copies of one another; craft a different shape instead), **or draw some shapes that are not filled** alongside the filled polygons, for example polygonal shapes that are only drawn as a line (if you do the latter, make sure that this shape includes more than one line segment). *You are welcome to do both, of course, and include multiple filled shapes and multiple line-drawn (non-filled strokes) along them!*
- You must demonstrate the use of more than one color in your drawing
- You must include at least one **slider** in your program, and have it affect the image drawn in some way. For example, it could control the location of one of the polygon vertices being drawn; or it could control the thickness of some line; or it could move an entire shape.

The requirements above are what is asked of you to get a “Satisfactory” grade (i.e. a score of “3”) in the programming assignment. If you want to contend for an “above-and-beyond” grade (i.e. “4”), you should consider drawing something more elaborate. For example (not an exhaustive list) :

- Something that creatively combines many shapes to create an interesting appearance.
- Something that changes shape (perhaps as triggered by the value of the slider) in an interesting way.
- Something that includes an animation (we will talk about options for that on the September 16th lecture).

You can satisfy the requirements by taking the short examples in the tutorials (and giving proper attribution – see the collaboration policy) and combining or modifying them. However, we recommend

that you put some effort into really understanding what is going on. And to make a picture that is more fun. Go ahead, show off your creativity!

We will give you a bunch of resources for getting started with JavaScript and Canvas (Start with the [“learning JavaScript”](http://graphics.cs.wisc.edu/WP/tutorials/learning-javascript/) [_ \(http://graphics.cs.wisc.edu/WP/tutorials/learning-javascript/\) _](http://graphics.cs.wisc.edu/WP/tutorials/learning-javascript/) page). If you’re new to JavaScript, take one of the example programs to start with, and experiment with changing the drawing command stop see what happens. In fact, you can tinker with the examples right in JSBin. And you might want to tinker with other parts of the program to see how it works.

Even if you know JavaScript and Canvas, please look through the tutorials – they will connect things to the other concepts in class.

To do this assignment, we recommend that you start with: [HTML5 Canvas What and Why](http://graphics.cs.wisc.edu/WP/tutorials/html5-canvas-what-and-why/) [_ \(http://graphics.cs.wisc.edu/WP/tutorials/html5-canvas-what-and-why/\) _](http://graphics.cs.wisc.edu/WP/tutorials/html5-canvas-what-and-why/) (well, maybe you should look at the [Learning JavaScript](http://graphics.cs.wisc.edu/WP/tutorials/learning-javascript/) [_ \(http://graphics.cs.wisc.edu/WP/tutorials/learning-javascript/\) _](http://graphics.cs.wisc.edu/WP/tutorials/learning-javascript/) page first)

And then read the: [Getting Started with HTML5 Canvas](http://graphics.cs.wisc.edu/WP/tutorials/getting-started-with-html5-canvas/) [_ \(http://graphics.cs.wisc.edu/WP/tutorials/getting-started-with-html5-canvas/\) _](http://graphics.cs.wisc.edu/WP/tutorials/getting-started-with-html5-canvas/) page. You might also want to go look at some more of the JavaScript Canvas resources listed on the [HTML5 Canvas What and Why](http://graphics.cs.wisc.edu/WP/tutorials/html5-canvas-what-and-why/) [_ \(http://graphics.cs.wisc.edu/WP/tutorials/html5-canvas-what-and-why/\) _](http://graphics.cs.wisc.edu/WP/tutorials/html5-canvas-what-and-why/).

This will give you enough to start doing the assignment. Then you’ll want to learn a bit more about what you can do with Canvas, and probably to become a better JavaScript programmer as well. The next programming assignments won’t be as easy...