

CS106 W21 - Assignment 2

Due: Friday, January 29, 11:59 PM

Note: On each assignment we will be running a program called MOSS to detect cheating.

Assignment 2 is graded out of 33 marks.

QUESTION ONE: Slide Show

You are to write a JavaScript p5 program to mimic what you see in the following video:

<https://vault.cs.uwaterloo.ca/s/CBJH64QnGGcsMEg>



My neighbour's cat hoping for treats

Question One Requirements and Grading [18 marks]

There is no specific starter code, but you may use any sketch that was provided this semester in Open Processing.

Before you start, watch the video which is linked above.

- 1) [2 marks] Download 8 photos from your camera and name them "1", "2", "3", etc. They will probably be JPG files but any file format is ok. It doesn't matter what the photos are but they should be from your phone and not from somewhere else.
- 2) [2 marks] Create a text file called "MyPhotosDescriptions.txt". There should be 8 lines in the text file. Each line describes uses text to describe one of the 8 photos. The first line describes photo "1", the second line describes photo "2", etc. So for example the text file might look something like the following:

```
Rattlesnake Park in Ontario  
Raisin bread from Woodfired Cafe in Waterloo  
Unusual looking but good Russian dish  
A rabbit in my yard  
Waterloo skyline  
Puzzle  
My neighbour blowing snow in his short pants  
My neighbour's cat hoping for treats
```

- 3) [2 marks] Load the photos into an array called "myPhotos".
- 4) [2 marks] Load the lines from the text file into an array called "myDescriptions".
- 5) [2 marks] When your sketches starts the photo in myPhotos[0] should be displayed as well as the corresponding text which will be in myDescriptions[0].
- 6) [2 marks] If the user clicks on the right hand side of the screen the display should change so that it displays the next items in the array so the first click would display myPhotos[1] and myDescriptions[1].
- 7) [2 marks] If the user clicks on the left hand side of the screen the display should change so that it displays the previous items in the array so the first click would display myPhotos[7] and myDescriptions[7].
- 8) [2 marks] At any time the user can press "r" or "R" for reset to start over.
- 9) [2 marks] if the user presses the "s" or "S" key, the current canvas is saved to a file named "screen.png".

QUESTION TWO: Create an Image

Question Two Requirements and Grading [10 marks]

There is no specific starter code, but you may use any sketch that was provided this semester in Open Processing. There is no video to watch.

This question is meant to be a bit of fun for you. You are to think about a game of your choice. Any game is fine. Make it up. It does not need to be an existing game. For this question, you are creating the splash screen only, and not creating the game itself. You have a lot of flexibility.

- 1) [2 marks] You must have at least one image from your camera.
- 2) [2 marks] You must have at least two animations from the provided sprite sheets in the zip file on LEARN named “uWaterlooSprites.zip”. These are the sprites for camel, chest, cards, etc.
- 3) [2 marks] You must include some text that is from a text file named “myText.txt”. You must create this text file.
- 4) [2 marks] if the user presses the “s” or “S” key, the current canvas is saved to a file named “screen.png”.
- 5) [2 marks] In comments in your JavaScript p5 sketch you must include a description of your splash screen. This description should come just after the first three lines that include your name and ID and a blank line.

You may use attributes and drawing functions from JavaScript p5 including ones such as `textSize()`, `rect()`, `ellipse()`, and any others that we have covered in CS105 or CS106.

[5 marks] Coding Style and Efficiency

Follow the course coding style for whitespace and comments. Consult the **“Code Style Guide”** on LEARN. For example:

- 1) [0.5] Include your name on the first line of code and your student ID number on the second line of code.
- 2) [0.5] Leave the third line blank.
- 3) [0.5] Comment your code appropriately. Avoid superfluous comments.
- 4) [0.5] Correctly and consistently indent your code blocks.
- 5) [0.5] Use correct inline spacing for variable declaration and assignment.
- 6) [0.5] Use good line spacing to chunk sections of your code.
- 7) [0.5] There are no variables that are declared or assigned, but not used.
- 8) [0.5] There are no unnecessary variables that are duplicates of other variables.
- 9) [0.5] There is no unnecessarily repeating the same code in multiple places.
- 10) [0.5] Semicolons were used appropriately (i.e. at the end of most lines).

Restrictions

- You may not use any functions or statements not covered in lecture or labs. This includes, but is not limited to:
 - No translate(), rotate(), or scale() functions.

Submitting

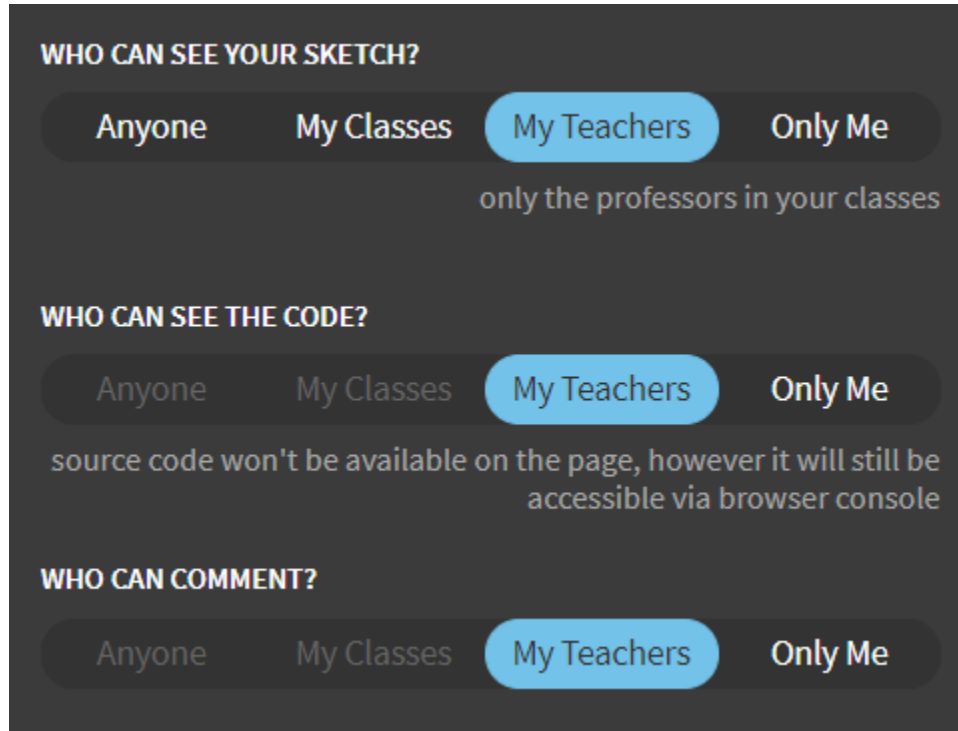
Use the template file in Word “CS106 Assignment Template” in LEARN to create your Assignment 01 submission.

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 Assignment 01 dropbox on LEARN.

An example of how to 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 file before the deadline. Otherwise you may receive a mark of 0.

Academic Integrity

All assignments in CS106 are done individually. Group work and sharing of code is not allowed.

Detecting Plagiarism:

- We monitor Reddit, File Trading Sites, past year CS106 assignments, etc.
- Measure Of Software Similarity (MOSS)
 - automatic system for determining the similarity of code

Discipline

- Discipline (Policy 71)
 - <https://uwaterloo.ca/secretariat-general-counsel/policies-procedures-guidelines/policy-71>