

# CMPSCI 373 (S20) Homework 1: Warmup

## Overview

This homework is a quick introduction to computer graphics programming using Javascript, WebGL, and three.js. Through this homework, you will ensure WebGL runs correctly in your browser, and you will experiment with some basic features of a provided demo program.

## Before you begin

- Have a modern browser, such as Firefox or Chrome. No Internet Explorer or Edge please!
- Have a decent program code editor, such as Sublime, Atom, Eclipse etc.

## Starter Code

- **Download** the starter code from Piazza, and **unzip** it. Windows Users: if you encounter issues due to anti-virus, you can do a git checkout of the git repository. The starter code consists of these files:
  - **index.html** is the main HTML file. It defines the title of the page, style (e.g. background color), and then includes the other .js (Javascript) files.
  - **three.min.js** is a Javascript library, based on WebGL, for easily managing 3D models, scenes, and creating graphics renderings.
  - **teapot.js** contains the definition of the teapot model.
  - **glWarmup.js** contains the main program code of this homework. It looks long and complicated, but don't worry, in this homework we will only use it as a starting example and modify some parts of it to gain insights. Over the course of the semester, we will gradually learn every part of it so that you can write such programs from scratch in the future.
- Open **index.html** in a browser, by either double clicking it, or opening an empty browser window, and dragging it to the browser window. You should see an orange colored triangle. Experiment with the following features:
  - Click on the object using **Mouse Left Button** and move up/down to **vertically rotate** the object. Note that your mouse pointer must be on the object for this action to take effect.
  - Click on the object using **Mouse Right Button** to **scale / resize** the object.
  - Click on the object using **Mouse Middle Button** to **translate / move** the object
  - Press key '**w**' to toggle between wireframe and solid rendering modes.
  - Press the **space** key to switch the object you are drawing.
- After the above experiments, open **glWarmup.js** in a code editor. Take a look at the code and try to understand it as much as you can. Think about the following questions on your own, and modify the program to see how the results change. Every time you modify the program, simply save it and refresh the browser window (e.g. Ctrl+R, or Command+R on Mac) to see updated results.
  - Where is the material defined? Try to modify the material color so that the shapes are drawn in **blue** instead of **orange**.
  - The triangle that you see initially: where is it defined? Try to modify the triangle vertices (x,y,z coordinates) and observe how it changes the shape of the triangle.
  - Where does the code detect key presses (e.g. pressing on space or the 'w' key)? Try to modify the code so that it responds to a different key (e.g. pressing 's' to switch shape).

# CMPSCI 373 (S20) Homework 1: Warmup

## What You Need to Do (12 points)

- **Written Question (4 pts)**

You have noticed that the mouse actions only apply if the mouse pointer is on the object, and won't apply if it's outside the object. This is controlled by the `clickedOnShape` variable, which is set to true if the mouse points on the object. The program sets this variable in the block of code around line 124. Check the code, and in **1-2 sentences**, **explain how the code detects mouse points on the object**, and **write down your answer at line 120**. It's ok if you don't fully understand how it works, just write down as much as you understand.

- **Programming Questions (4 pts + 4 pts)**

For this part, modify the demo code in the following ways:

1. Add **two additional** shapes to the program. To get an idea of what additional shapes there are, go to [three.js documentation](#), scroll down to the **Geometries** section, and check the list of geometries supported. Then follow the code of existing shapes to add the two new shapes. This way, when pressing the space key, it will iterate through all shapes including your new shapes. *[Hint: use `shapes.push` to add your new shapes to the list].*
2. The demo code only supports vertical (i.e. up and down) rotation of objects. Modify the code so you can **also rotate objects horizontally** (i.e. left and right) when you left click on the object and move horizontally. This allows the program to support rotations in both dimensions.  
*[Hint: search the code for all occurrences of the `angleV` variable, which handles the vertical rotation angle. Create a corresponding `angleH` variable to handle horizontal rotation angle].*

## Submission

Zip your homework 1 folder (including all files) and submit the zip file to Gradescope.

## Additional Reading

- If you have never programmed in Javascript before, find a Javascript tutorial online, such as [this one](#), and read the first few chapters to get familiar with it. Generally, Javascript bears a lot of similarities to Java, but there are some key differences, particularly in the flexibility of variable definition, types, and handling of arrays.
- Take a look at this [Introduction to WebGL slides by David Scott](#).