EE 267 Virtual Reality: Lab 1

Instructions

Students should complete this lab before starting on Homework 1. Completing the lab will give you useful information that will aid in completing the homework assignment.

Task 1: Javscript Tutorial

Alongside HTML and CSS, JavaScript is one of the three core technologies of the World Wide Web content development. HTML is used to define content of web pages, CSS is used to specify the layout and style of web pages, and JavaScript is used to specify the behavior of web pages. Javscript is supported by the majority of websites, and all modern web browsers support it.

In this class, you will be using JavaScript for the majority of the homeworks. With this in mind, it is important for you to gain a solid understanding of JavaScript before attempting the homeworks. Please go through the JavaScript tutorial offered by W3 which you can find here: https://www.w3schools.com/js/default.asp. Complete the tutorial up to and including the "JS Scope" section, but we encourage students to go through each of the sections.

Task 2: THREE Tutorial

While JavaScript is ubiquitous and used in many applications, in this class, we will be mainly using it to interface with WebGL. WebGL is a JavaScript API for rendering interactive 2D and 3D graphic from a browser. It is closely related to OpenGL, and both APIs are used to interact with a GPU, achieving hardware-accelerated rendering. However, working directly with WebGL can be tedious and many libraries have been created to simplify the job of developers. In this class we are using Three.js, which is a popular library wrapping WebGL. Some of the rendering pipeline will already be implemented for you in the homeworks, but you will find many functions offered by Three.js useful over the duration of the course. Please watch the video introducing WebGL with Three.js from the Front Porch Conference, https://github.com/davidlyons/threejs-intro. You can find the full Three.js documentation here, https://threejs.org/docs/.

Task 3: Chrome Developer Tools

You will need some systematic way of debugging your JavaScript code. JavaScript runs from within the browser and luckily Google Chrome has native developer tools to aid you with debugging. There are two main features that you will be using throughout the homework: the console panel and the JavaScript debugger. The console allows you to view a chronological log of messages output by our JavaScript using the `console.log()` function, and even allows you to interact with your JavaScript variables directly. You can think of it as analogous to a terminal.

The JavaScript debugger native to Chrome is a powerful and efficient way of debugging your code with breakpoints, stepping, and inspecting in-scope variables. To get familiar with the JavaScript debugging, read the "Debug JavaScript": https://developer.chrome.com/docs/devtools/javascript/ and "Console overview (https://developer.chrome.com/docs/devtools/console/)". You can find more Chrome DevTools tutorials at https://developer.chrome.com/docs/devtools/. Although you can also edit the code and save the change in Chrome, we generally recommend edit the code with your favorite editor such as Vim, Visual Studio Code and Atom.