**Notes: Week 2**

Background:

Programming is about making computers do what we want. A program, then, is a collection of instructions that you give your computer to perform. High-level languages make programming easier for people to read/write, as it separates them from the 1’s and 0’s of machine language. Scripting languages are high level, but are interpreted, meaning they are translated into machine code at run time.

JavaScript is one of the most popular programming languages. It is a high-level scripting language. It requires an engine to run it, most commonly web browsers.

JavaScript was produced by Brendan Eich, a Netscape employee, to make webpages more interactive. It showed up in version 2 of Netscape Navigator browser in 1995.

By 2005, sites like google maps appeared, showing the JavaScript could create rich internet applications that behaved like native desktop applications. Around the same time, AJAX appeared, a technique for getting server data without a full page reload.

In 2009, Node.js was created, which allows server-side applications to be written in JavaScript.

JavaScript in the Console:

The console is useful as it allows you to enter code and it displays the output after by using the REPL or Read Eval Print Loop. Type ‘node’ in the terminal to enter node js console. Or, you can use the console that comes with your browser.

Your first javascript program:

It is a tradition to make a hello world program. Type:

Console.log(‘hello world!’)

To achieve this.

Three layers of the web:

The three layers of the web are HTML, CSS, and JavaScript. HTML marks up the content, CSS styles it, and JavaScript adds interactivity.

You can place the JavaScript code within the HTML, or put it within its own <script> tag. Then, you can put JavaScript in its own file and connect it to the HTML within the <script> tag. The last option is what you want to do.

Graceful Degradation and Progressive Enhancement:

Graceful degradation is the process of building a website so it works best in a modern browser that uses JavaScript, but still works to a reasonable standard in older browsers, or if JavaScript or some of its features are unavailable.

Progressive Enhancement is the process of building a web page from the ground up with a base level of functionality, then adding extra enhancements if they are available in the browser.

Backwards Compatibility:

JavaScript also needs to be backwards compatible. Old code must be able to work with new code. Transpilers are used to convert old versions of JavaScript code to new versions.

Ch 2, programming basics:

JavaScript grammar is known as a C-style syntax.

Certain words are reserved and cannot be used to name variables.

There are seven data types

Six primitive data types are: string, symbol, number, Boolean, undefined, null

Declaring variables involves using const or let. Const cannot be reassigned to another value. The variables are scoped, meaning that they are only useable within the block they are declared in. Anything declared outside of a block has global scope.

Template literals use the backtick (`) character. It is meant to deliminate the string.

Symbols are mainly used as object property keys

If a number starts with 0x, then it is considered hexadecimal. If it starts with 0o, then it is octal.

Exponential notation uses e, such as 1e6, 2E3, or 2.5e-3.

Error values: infinity means the number is too big. NaN stands for not a number.

Undefined is used for variables that have not been given a value. Null means no value.

Ch 3 Arrays, Logic, and Loops:

An array literal is an array that already has some initial values. Values can be removed by using the delete operator.

Destructuring an array is to take the values out of the array and present them as individual values.

Arrays can be multidimensional.

A set is a data structure that represents a collection of unique values, so it cannot include any duplicate values.

Maps are a data structure that can keep key value pairs. They are similar to hashes, hash tables, or dictionaries in other languages.

A short-hand way of doing if else logic is to use the ternary operator.

A for-of loop can iterate over an array.

Ch. 4 Functions:

Functions can be defined using a function declaration

Another way is a function expression, which assigns an anonymous function to a variable.

Functions can also be declared using the constructor function(), which is not recommended.

Functions can also be declared through arrow functions. The parameters come before the arrow and the main body of the function comes after. Arrow functions are anonymous and require being assigned to a variable if you wish to refer to them. They have the following advantages:

* They are less verbose
* Single parameters don’t need parenthesis
* The body of the function doesn’t need placing inside a block if it’s only one line
* The return keyword isn’t required if the return statement is the only statement in the body of the function
* They don’t bind their own value of “this” to the function

Multiple parameters require parenthesis. If no parameters are required, empty parenthesis must go before the arrow. Longer arrow functions require curly braces

Hoisting is moving a variable and function declarations to the top of the current scope.

***Callbacks:***

Functions can be given as a parameter to another function. A function that is passed as an argument to another is known as a *callback*. A function can be provided as a parameter, and then invoked within the function, as so:

function sing(song) {

console.log(`I'm singing along to ${song}`);

}

sing('Let It Go')

<< 'I'm singing along to Let It Go'

function sing(song,callback) {

console.log(`I'm singing along to ${song}.`);

callback();

}

“callback” is provided as a parameter, then invoked inside the body of the function.