Week 3 Note

**What is Object?**

An object in JavaScript is a self-contained set of related values and functions. They act as a collection of named properties that map to any JavaScript value such as strings, numbers, booleans, arrays and functions. If a property’s value is a function, it is known as a method.

* Objects are usually created to represent entities of the real world
* let user = {
* name: "John",
* age: 30
* };
* .
* .
* To Access the object. A method can use the “this” keyword
* sayHi() {
* // "this" is the "current object"
* alert(this.name);
* }
* };
* user.sayHi(); // John

“this” keyword is a keyword whose value changes depending on how a function gets called.

**The Math Object**

The Math object is a built-in object that has several properties representing mathematical constants, as well as methods that carry out a number of common mathematical operations.

* Math.abs(3);
* << 3
* .
* Math.ceil(4.2);
* << 5
* .
* Math.floor(-4.2);
* << -5

**The Date Object**

* const christmas = new Date('2017 12 25');
* christmas.toString();
* << 'Mon Dec 25 2017 00:00:00 GMT+0000 (GMT)'

**The Document Object Model**

The Document Object Model (DOM) allows you to access elements of a web page and enable interaction with the page by adding and removing elements, changing the order, content and attributes of elements, and even altering how they are styled.

* HTML document as a network of connected nodes that form a tree-like structure.
* The DOM treats everything on a web page as a node. HTML tags, the text inside these tags, even the attributes of a tag are all nodes.

**The classList Property**

* is a list of all the classes an element has.
  + Add
  + Remove
  + Toggle
  + Contains

Appending Nodes

* appendChild() method will add another node as a child node
* flash.appendChild(flashText);

Event Listeners

* They are like setting a notification to alert you when something happens.
* if (click) {
* doSomething();
* } else {
* // carry on with rest of the program
* }

Exercise 14.3, The cat’s hat.

* <!doctype html>
* <meta charset="utf8">
* <base href="https://eloquentjavascript.net/">
* <style>body { min-height: 200px }</style>
* <img src="img/cat.png" id="cat" style="position: absolute">
* <img src="img/hat.png" id="hat" style="position: absolute">
* <script>
* let cat = document.querySelector("#cat");
* let hat = document.querySelector("#hat");
* let angle = 0;
* let lastTime = null;
* function animate(time) {
* if (lastTime != null) angle += (time - lastTime) \* 0.001;
* lastTime = time;
* cat.style.top = (Math.sin(angle) \* 40 + 40) + "px";
* cat.style.left = (Math.cos(angle) \* 200 + 230) + "px";
* hat.style.top = (Math.sin(angle + Math.PI) \* 40 + 40) + "px";
* hat.style.left = (Math.cos(angle + Math.PI) \* 200 + 230) + "px";
* requestAnimationFrame(animate);
* }
* requestAnimationFrame(animate);
* </script>

Exercise 15.2, Mouse trail

* <!doctype html>
* <style>
* .trail { /\* className for the trail elements \*/
* position: absolute;
* height: 6px; width: 6px;
* border-radius: 3px;
* background: teal;
* }
* body {
* height: 300px;
* }
* </style>
* <body>
* <script>
* let dots = [];
* for (let i = 0; i < 12; i++) {
* let node = document.createElement("div");
* node.className = "trail";
* document.body.appendChild(node);
* dots.push(node);
* }
* let currentDot = 0;
* window.addEventListener("mousemove", event => {
* let dot = dots[currentDot];
* dot.style.left = (event.pageX - 3) + "px";
* dot.style.top = (event.pageY - 3) + "px";
* currentDot = (currentDot + 1) % dots.length;
* });
* </script>
* </body>