

Javascript Research Notes - Joshua Williams

- `getElementById("IDName")` <- how to access HTML elements inside of javascript
 - The style of the element can be accessed using `.style.property`
 - Property being any css property such as display
- `console.log()` <- can be used to log to the console (can be seen by pressing F12 in chrome)
- Two types of values
 - Fixed values known as literals
 - Variable values known as variables
- All common arithmetic can be used
- `=` is the assignment operator
- Comments are done through `//`
- Javascript variables do not have fixed types => use `let` or `var`
- Semicolon is optional
- `const functionName = function(params){}`
 - This defines a javascript function
 - Return value type does not need to be specified
- Variable scope is very similar to other languages, local and global variables exist
- Node.js is typically used to backend javascript, however many other languages like php can also be used with javascript (javascript frontend and php backend)
 - Similar to Ruby's Event Machine, Python's Twisted, etc.
 - Designed without threads, however you can use `child_process.fork()` API to take advantage of multiple cores
- Javascript Objects
 - Ex:
 - `var person = {`
 - `firstName: "John",`
 - `lastName: "Doe",`
 - `age: 50,`
 - `eyeColor: "blue"`
 - `};`
 - The properties of the object can be accessed like `person.age` or `person.eyeColor`
- `==` equal to (does not take type into account `false == 0` would return true)
- `===` equal to that does take type into account
- `?` - ternary operator
- `&& || !` - logical operators, same as other languages
- Javascript array notion:
 - `var cars = ["Saab", "Volvo", "BMW"];`
 - Can access elements like `cars[0]` or `cars[1]`
- For loops work similar to other languages, but you use `let` instead of `int` in variable declaration
- While loops also work pretty much the same as other languages
- `const` can be used if the variable you are creating is constant, otherwise use `let`

- Javascript is mainly used as a frontend language, however Node.js does exist for backend development
- Javascript Math function
 - Math.PI - returns the value of pi
 - Math.round() rounds the value up to the nearest int
 - Math.pow() allows for exponents
 - Math.sqrt() takes the square root of a number
 - Math.min() and Math.max() return the minimum value and maximum value respectively
- 5 primitive types in javascript
 - Undefined
 - Null
 - Boolean
 - String
 - Number
- Non-Primitive types
 - Objects
- Date datatype
 - new Date()
 - new Date(year, month, day, hours, minutes, seconds, milliseconds)
 - new Date(milliseconds)
 - new Date(date string)
- Arrow functions
 - Allows for shorter syntax, ex:
 - hello = () => {
 - return "Hello World!";
 - }
- Node.js allows javascript to perform SQL queries