1. Create Repository in the Cloud (github.com)
2. Create word folder in local computer.
3. Initialize **git** on workfolder \*\*\*(purpose of Git is to track your work)\*\*\*
   1. Git init
      1. Or go to Source control tab 
      2. Got to Icon
4. Create GitHub Repository
5. Sync
   1. Go to Terminal at the top.
      1. Select New Terminal
   2. Git remote add origin (repository url)
6. Create file commit & Push
7. When you pull data
   1. Create a remote (already done
   2. Git pull origin master

Console.log (‘string)

“this string”

\*\* use single quotes for strings. You will bel about to add double quotes inside of them.

Example ‘ simple text “like this” is cool ‘

Javascript Engines

* Every browser has one
* Node.js

Variable can start with :

* \_
* $ (points to JQuery)
* Letters

DOM – document object model

\*\*\*ES Lint \*\*\* helps you write clean code

* Got to the extension icon 
* Type ESLint
* When you create a file you have to use the code ‘use strict’ for it to work

JavaScript has:

* Variable (things that can be changed)
* Constants (things that can’t be changed)
* Data Types
  + Numbers
    - Variable is number
      * It is a double precision
        + An interger value ex. 10
        + An exponent value which tells you were to store the decimal ex. 10e-1
        + Ex 0x00F;
        + 0o007
        + Infinity
  + Strings
    - Enclose in quotes
    - Unicode characters
      * 2 bytes per character
  + Boolean
    - True & False
  + Object
  + Null and Undefined
  + Symbol
  + Array

Literals (is a constant) are unmutable (can’t be changed)

Variable

Variable

* + Are assigned literal

Literal

* + Var myValue = “my Value”

There is an operations called Casting. In order to use it the variables need to be the same type. Ex.

let valueA = 10;

let valueB = “5”

let total = valueA \* valueB;

JSON – Java Script Object Notation Ex. Let obj1 = {};

XSD – XML Schema Definition

\*\* If you need help with JavaScript:

* W3schools.com
* Stackoverflow.com

Boolean Values

* False
  + Undefinied
  + 0
  + NaN
  + Null
  + Empty string ‘’
* True
  + Everything other than 0

To create Boolean expressions you need:

* Comparison Operators
* Logical Operators
* Var x if(x) undefined = false

Expressionish Statement

* EXPRESSION – SOMETHING THAT RETURNS A VALUE

Literal

Ex. 3 + 5

undefined

Var x,

If (!x) { undefined = false

X = 10;

}

Nonexpressionish Statement

Comparison Operator

equality

== straight –type and value

=== abstract - value

>=

relational

>

<=

<

Loops

🡪 repeat statements

Conditionals

* Decide path