6th Mar ‘20

<https://www.youtube.com/watch?v=PkZNo7MFNFg> freeCodecamp.org

code editors: sublime, Code visual studio, atom.

Online editors: codepen, scrimba. Codepen.com [okstamps@gmail.com](mailto:okstamps@gmail.com) 1@678582

Js ignore comments. Inline comments: // comment

Multi line comment /\* text \*/

Data types and variables: 7 data types: undefined, string, number, Boolean, null, symbol, object

Symbol is an immutable primitive value that is unique. Object can store key:value pairs

Declare variable is 3 ways: var <varName> = value (global scope). let <varName> = value (local scope)

const <varName> = value (cannot change the value) // var a; every js statement ends in ‘;’ semi colon

console.log allows to see things in the console.

Variable names and function names in Js is **case sensitive**

myVar++ , compound assignment var += can be used. If a number divided by 2 is 0, then the number is even.

Backtick ‘`’ can be used in include single and double quotation marks inside a string.

Concatenate strings with ‘+’. A variable can be considered as a ‘box’.

Strings are immutable.

var myVar

function formSentence(noun, proNoun, adjective, adVerb) {

myVar = "The " + noun + proNoun + adjective + adVerb // variable has function scope

return myVar

}

var text = formSentence("quick", "brown", "fox", "jumped")

console.log(text)

console.log(myVar)

output: "The quickbrownfoxjumped"

"The quickbrownfoxjumped"

--

Array can store any datatype. Arrays are mutable. Array.push()

var myArray = [ ["Jaison", "Jacob", 10], ["Sonia", "Jaison"], ["Esther", "Sarah"]]

myArray.push("Hello", "world")

console.log(myArray)

array.pop() pops the last element. // array.shift() removes the first element from the array.

Array.unshift() adds element to the beginning of the array. Typeof – used to find the type of a variable.

**If a variable is defined without ‘var’ or other specifiers, it becomes ‘global’** automatically.

Variables defined inside a function and parameters has a ‘local’ scope.

‘undefined’ is the default return value of a function.

var myArray = [1,2,3,4,0,5];

console.log(myArray);

myArray.push(6,7);

console.log(myArray);

myArray.pop();

console.log(myArray);

myArray.shift();

console.log(myArray);

var js = JSON.**stringify**(myArray);

console.log(js);

Always use ‘paratheses’ for an ‘if’ condition. true / false is lowercase.

Strict equality check ‘===’ uses no type conversion before the comparison it does with like ‘==’

function checkVal(a) {

if (a === 10) {

return 'a == 10'

} else if (a > 10) {

return 'a > 10'

} else

return 'a < 10'

}

console.log(checkVal(10))

function switchCase(val) {

var answer = " ";

switch (val) {

case "a":

answer = "Apple";

break;

case "b":

answer = "Banana";

break;

case "c":

answer = "Carrot";

break;

case "d":

anwer = "Dango";

break;

default:

answer = "Happy"

break;

}

return answer;

}

var res = switchCase("a");

console.log(res)

objects use properties to access data.

Properties can be accessed using object.propertyname or object[property name]

Delete object.property can be used to delete the property.

randomRagne(), parseInt(string, base). Ternary: <condition> ? ‘return if true’ : ‘return if false’

function checkSign(num) {

return num > 0 ? "positive" : num < 0 ? "negative" : "zero"; // see the usage like 'else if'

}

console.log(checkSign(0));

‘let’ will not allow a variable to be declared twice in the same scope. let maps only to its ‘scope’

“use strict” at the top of the programs to catch coding errors.

Can mutate an array declared with ‘const’

Object.freeze(objectconstVARIABLE) immutates the variable.

// anonymous function

Const var just = function () {

return new Date();

};

// another form

var just = () {

return new Date();

};

// another form

var just = () => new Date();

console.log(just);

--

Rest operator is 3 dots’ …’

It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions. In this path you will learn the basics of JavaScript as well as more advanced topics such as promises, asynchronous programming, proxies and reflection.