1. **JavaScript Tutorial for Beginners: Learn JavaScript in 1 Hour**

<https://www.youtube.com/watch?v=W6NZfCO5SIk>

# Object-oriented Programming in JavaScript: Made Super Simple | Mosh

<https://www.youtube.com/watch?v=PFmuCDHHpwk&t=2222s>

# JavaScript: Understanding the Weird Parts - The First 3.5 Hours

<https://www.youtube.com/watch?v=Bv_5Zv5c-Ts>

# JavaScript Tutorial for Beginners 2021 (New Version Available - See Description) – Make a To-Do List

<https://www.youtube.com/watch?v=DqaTKBU9TZk&t=8s>

# JavaScript Concepts You HAVE TO KNOW

# An intro to some topics: <https://www.youtube.com/watch?v=a00NRSFgHsY>

# Array Methods:

# <https://www.youtube.com/watch?v=OMoxLUxW7Wc>

McDonald video: <https://www.youtube.com/watch?v=t3Gi1kab-qQ&t=28s>

AD 1)

# What is JavaScript? A leading Programming language.

# What can you do with it? Web/Mobile Apps, Real-time Networkding Apps, Command-line Tools and Games.

# Where does JavScript code run? In Browsers and in NodeJS (for back-end).

# JavaScript vs. ECMAScript? ECMA script is a standards specification, JavaScript is a programming language that confirms to this specification.

# Use Chrome Developer tools (in a Browser) and test JavaScript commands there (using the Console tab). Fx. *console.log(‘Hello World’);*

# Setup your development environment: Editor, fx. Microsoft Visual Studio Code.

# Install Nodejs from nodejs.org (choose the LTS version). We will primarily use Nodejs to install 3rd party libraries.

# START LEARNING JAVASCRIPT:

# Create your first index.html file in a folder of your own choosing.

# In the Extensions tab, look for and install *live server*. Maybe you need to restart VS Code after the installation.

# Now you can right-click the index.html file and select “Open with Live Server”, which starts a local server and shows your index.html file in a browser.

# THE SCRIPT ELEMENT -> in the Head section or in the Body section?

# EXTERNAL JAVASCRIPT FILE -> reference in *src attribute of* SCRIPT ELEMENT.

# STATEMENTS -> Actions to be carried out.

# TERMINATE STATEMENTS WITH A SEMI COLON “;”.

# A STRING -> is a sequence of characters.

# COMMENTS -> // to the end of the line. OR, the well-known /\* \*/

# SEPARATION OF CONCERNS.

# CREATING AN INDEX.JS FILE. Reference in SCRIPT-TAG via *src* attribute.

# TESTING A .JS FILE IN NODE -> *node index.js (press Enter) in CMD env.*

# NO MORE NODEJS IN THIS COURSE…

# VARIABLES:

# Used to store data temporarily.

# It is a location/address with a name and a value.

# Keywords to declare vaiables: *var*, *let* and *const*.

# Best practice: avoid using the *var* keyword.

# undefined is both a type and a value. It is the default value of a variable.

# Below; 2 JS statements to declare a variable and output it:

# *let name = ‘Mosh’;* (you can use single quotes or double quotes).

# *console.log(name);*

# Some naming rules for variables:

# cannot be reserved keywords

# they should be meaningful

# cannot start with a number

# cannot contain space or hyphen (-).

# are case-sensitive.

# CAMEL CASE /PASCAL CASE.

# CONSTANTS -> keyword: *const*. Do not allow for re-assignment. Use with Objects and Arrays or ordinary variables that are not supposed to change.

# PRIMITIVES/VALUE TYPES:

# string, boolean, number, undefined, null and symbol.

# REFERENCE TYPES:

# functions, objects and arrays.

# STATIC LANGUAGE (Statically-typed) AND DYNAMIC LANGUAGE (Dyn. Typed).

# TYPEOF -> function to show the type of a variable, f.x.:

# *typeof name*

# OBJECT -> is a Reference Type: objects, arrays and functions are objects.

# OBJECT LITERAL -> *let person = {};*

# OBJECT “PERSON”: (objects are key/values pairs. Keys are attributes.).

# Fx. *let person = {*

# *name: ‘Mosh’,*

# age: 30

# }

# WORKING WITH OBJECTS -> 2 ways: *Dot Notation* and *Bracket Notation*.

# Dot Notation: *console.log(person.name);*

# Dot Notation: *person.name = ‘John’;*

# Bracket Notation: *console.log(person[“name”]);*

# Bracket Notation: *person[‘name’] = ‘Mary’;*

# ARRAYS -> lists of values/objects. They are dynamic in size and content.

# Fx. *let selectedColors = [];* // An Array Literal… (an empty array).

# Fx. *let selectedColors = [‘red’, ‘blue’];*

# Output first element: *console.log(selectedColors[0]);*

# Add element: *selectedColors[2] = ‘green’;*

# Add element of different type: *selectedColors[3] = 1;*

# Length of an Array: *console.log(selectedColors.length):*

# TECHNICALLY AN ARRAY IS AN OBJECT!

# FUNCTION-> A Fundamental building block of JavaScript.

# A Named set of statements that perform a task or calculate a value.

# Fx. *function greet() {*

# *console.log(‘Hello World’);*

# *}*