**Introduction and Basics of JavaScript | Part 1 -** [Denis listiadi](https://denislistiadi.medium.com/?source=post_page-----6c99792e0e25--------------------------------) Mar 29, 2022

**JavaScript History**

JavaScript or often abbreviated as JS was first created in 1995. A programmer at Netscape named Brandan Eich made a scripting programming language to run on the Netscape Navigator Browser.

Previously the name was Mocha, then changed to LiveScript, and finally to JavaScript.

JavaScript is made so that the website pages that we have created using HTML and CSS can be more interactive

JavaScript has now defeated Java Applets and Flash as programming languages to make web pages more interactive, this is due to the ease of the language and also by default now all browsers can run JavaScript without having to install additional applications such as Java Applets and Adobe Flash Player.

**JavaScript on Server**

Initially, JavaScript was mostly used to run on the client-side (Browser). But lately, since NodeJS technology came out that can be used to run JavaScript without a browser, now finally JavaScript is also widely used to create applications on the Server.

Because of this, finally, JavaScript is now known as the FullStack programming language (Backend and Frontend) because it can be used to create Backend applications and Frontend applications.

**JavaScript and ECMAScript**

Because JavaScript is now almost adopted by browser applications, finally a standard is made called ECMAScript. The organization that standardizes ECMAScript is ECMA International.

Now with standardization, we can make sure that our JavaScript program code can run in all browsers, because browsers that want to support JavaScript, must follow the ECMAScript standard.

Now that ECMAScript and JavaScript are the same, it can now be said that ECMAScript and JavaScript are two names for the same programming language.

**JavaScript vs Java**

Beginner programmers are often wrong about JavaScript and Java. Some think that Java and JavaScript are the same programming language, but they are different.

Java is another programming language, with nothing to do with JavaScript  
Although the name has the word “Java” in but these two programming languages are completely different, there is no connection at all

**Development Tools**

When we learn JavaScript, we need to prepare some software to help with development.

Browser, this is for sure, because we need to run JavaScript program code using Browser. There are various browsers now, such as chrome, Mozilla firefox, edge, opera, safari, etc

Text Editor or Integrated Development Environment is used to create JavaScript program code. Like VS Code, Atom, Sublime, WebStorm, etc.

**Creating JavaScript Code**

There are several ways to code JavaScript. Can be directly in the HTML file or you can use a .js file (an extension for JavaScript), then include it in the HTML file. In this practice course, we will use HTML directly to make it easy to code the program

**Script JavaScript in HTML**

Graphical user interface, text

Description automatically generated

**Include Script JavaScript**

Logo

Description automatically generated with medium confidence

**Semicolon in JavaScript (;)**

JavaScript is similar to the C/C++ programming language, were at the end of each program code statement, we need to add ; (semicolon)

However, in JavaScript the ; (semicolon) is not mandatory, so we can add or not.

It is highly recommended to be consistent, if you want to use a semicolon, use it everywhere, if not, don’t use it everywhere. I prefer to use a semicolon

**Comments in JavaScript ( // or /\* )**

Comments are program codes the codes will not be executed when read. Comments are usually used as additional information or hints. In JavaScript, we can add comment code

Text

Description automatically generated

**Data types in JavaScript**

**Number data type**

JavaScript only supports one number data type, where the number data type in JavaScript can be an integer or a decimal number. Unlike in other programming languages which usually distinguish between integer and decimal number data types, in JavaScript everything is unified

Text

Description automatically generated

**Number Notation**

JavaScript supports number notation, the default is base 10, JavaScript also supports binary, hexadecimal, and octal

* Hexadecimal : 0xFF
* Binary : 0b10101
* Octal : 0o10

**Boolean Data Type**

The boolean data type is a data type that contains truth data, meaning that there are only two data, true and false (yes or no). True is represented by the true keyword, and false is represented by the false keyword. Although the boolean data type is very simple, the boolean data type will be widely used in various programming features.

Text

Description automatically generated

**String Data Type**

The string or text data type is a data type that contains an empty set or more characters. Previously we have used strings when creating a hello world program. In JavaScript, to create data of type string, we need to use “ (double quote) or ‘ (one quote) before and after the content of the text.

Text

Description automatically generated

string data can also be concatenated using plus (+)

<script>  
 document.writeln("HelloWorld" + "Hello World again");  
</script>

**Escape Sequence**

JavaScript supports escape sequences in strings. Escape sequences are special characters, such as ENTER, TAB, “ (double quotes), etc. The following is an example of an escape sequence supported by JavaScript in string data.

* **\n** for ENTER
* **\t** for TAB
* **\’** for ‘
* **\”** for “
* **\\** for \

Text

Description automatically generated

**Variable**

A Variable is a place to store data. By storing data in a variable, we can use it again by mentioning the name of the variable. To create a variable in JavaScript, we can use the var keyword followed by the variable name.

JavaScript is a dynamic language, meaning that variables in JavaScript are not fixed, they must use one data type, we can change the data type in the same variable.

<script>  
 var fullName  
</script>

**Changing Value in Variable**

After the variable is declared, we can change the value in the variable. To change it, we can use the variable name command followed by the = (equals) sign and then followed by its value.

Text

Description automatically generated

**Creating Direct Variables With Value**

In JavaScript, we can also declare a variable, directly with the contents of its value. The way we can use the keyword var is followed by the name of the variable, then followed by the = (equals) sign, and followed by the value.

Text

Description automatically generated with low confidence

***Variable and Value:****From the code above, fullName is the variable, and Denis Listiadi is the value*

**Accessing Variables**

One of the advantages of using variables is that they can be reused. This will make it easier when you need the same data many times. To access a variable, we just need to mention the name of the variable

Graphical user interface, text, application

Description automatically generated

**let and const keywords**

Before 2015, the keyword to create a variable could only use the var keyword. However, since the 2015 ECMAScript version, new keywords were introduced to create variables, let and const.

JavaScript is now no longer recommended to use the var keyword to create variables, but replaced with let, this is because there are problems with the initial design of var (we will discuss separately).

**Variable let**

let fullName = "Denis Listiadi";

**Variable const**

The let keyword is like the var keyword, where the data in the variable can be changed as we like. While the const keyword is different, when a variable has been filled in the const variable, the value of the variable cannot be changed anymore. This kind of variable is sometimes called constant.

A screenshot of a computer

Description automatically generated with medium confidence

**Mathematical Operators**

JavaScript supports various Mathematical operators for the Number data type, such as:

* Arithmetic Operators
* Augmented Assignments Operators
* Unary Operators
* And others

**Arithmetic Operators**

A screenshot of a computer screen

Description automatically generated

For example in the implementation of the code

A screenshot of a computer

Description automatically generated with medium confidence

**Augmented Assignments Operators**

Table

Description automatically generated

For example in the implementation of the code

A screenshot of a computer

Description automatically generated with low confidence

**Operator Unary**

* **+ :**Indicates a positive value
* **- :**Indicates a negative value
* **++ :**Increment, increase 1 point
* **- - :**Decrement, decrease 1 point

For example in the implementation of the code

Text

Description automatically generated

**Comparison Operators**

A comparison operation is an operation to compare two pieces of data. A comparison operation is an operation that returns a boolean value (true or false). If the result of the operation is true, then the value is true  
If the result of the operation is false, then the value is false

* **==**isequal to
* **===**is equal value and equal type
* **!=**is not equal
* **!==**is not equal value or equal type
* **>**is greater than
* **<**is less than
* **≥**is greater than or equal to
* **≤**is less than or equal to

For example in the implementation of the code

Text

Description automatically generated

**Logic Operators**

Logical operators are operators for two boolean data. The result of the logical operator is a boolean again.

* **&&**is AND
* **||**is OR
* **!**is NOT

**Operator AND ( && )**

Table

Description automatically generated

**Operator OR ( || )**

Table

Description automatically generated

**Operator Unary ( ! )**

Table

Description automatically generated

For example in the implementation of the code

Text

Description automatically generated

**Since it’s too long, it will be continued in part 2 here**

[JavaScript: Introduction and Basics of JavaScript | Part 2](https://denislistiadi.medium.com/javascript-introduction-and-basics-of-javascript-part-2-772e7473aa4d)