**Introduction and Basics of JavaScript | Part 3 -** [Denis listiadi](https://denislistiadi.medium.com/?source=post_page-----8abf61f717f7--------------------------------) Mar 30, 2022

Before continuing with basic JavaScript, if you haven’t read part 2, please read it first at [Introduction and Basics of JavaScript | Part 2](https://denislistiadi.medium.com/javascript-introduction-and-basics-of-javascript-part-2-772e7473aa4d). Without further ado, let’s get straight to the basics of javascript.

**Switch Expression**

Sometimes we just need to use simple conditions in the if statement, such as using only comparison ==

A Switch is the same branching statement as if, but it’s simpler to make. The conditions in the switch statement are for comparison purposes only ==

Text

Description automatically generated

**Operator typeof**

typeof is an operator that we can use to see the data type of a value or variable. Because JavaScript is a dynamic language, so sometimes we need to check the data type of a value or variable using the typeof operator. The result of the typeof operator is the string data type.

Table

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

**In Operator**

In operator is an operator that can be used to check whether a property is in an object or not. If the property is in the object, then the result is true, otherwise, the result is false. Not only in objects, but it can also be used to check the index in Array.

Text

Description automatically generated

***Warning:****In operator will only check whether the property or index exists or not. So even if the value of the property or index is undefined or null, it will still be considered there*

Text

Description automatically generated

**Ternary Operator**

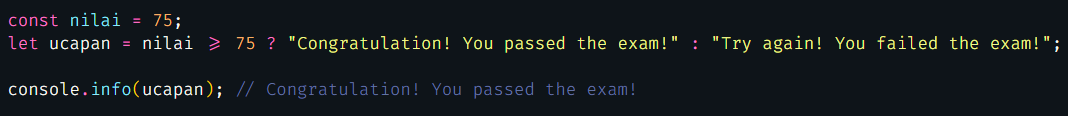
The ternary operator is the simplest operator of the if statement. The ternary operator consists of conditions that are evaluated, if it returns true then the first value is taken, if false, then the second value is taken.

Let’s just give an example when we don’t use the ternary operator below

A screenshot of a computer

Description automatically generated with medium confidence

We’re going to need a lot of lines of code to make the contents of the speech variable match this condition. But if we use the ternary operator, it only takes one line, please see the code below



**Nullish Coalescing Operator**

The nulllist value is null and undefined. Nullist coalescing operator (??) is an operator similar to the ternary operator, the difference is in the condition if the value is null or undefined, then the default value is taken. Let’s just give an example when we don’t use the Nullist coalescing operator below

Text

Description automatically generated

We will need many lines of code to make the contents of the variables match the conditions. But if we use the nullish coalescing operator, it only takes one line, please see the code below

Text

Description automatically generated

**Optional Chaining**

Optional chaining operator (?) is an operator that is used to secure when we want to access the property of an object from nullish data.

If we try to access the properties of an object from nullish data without using the optional chaining operator, an error will occur

Graphical user interface, text

Description automatically generated



So that there are no errors, we must first check. Now to check we can use an if expression or optional chaining. First, let’s look at the one using the if expression below:

Text

Description automatically generated

And below is using optional chaining with the same case

Text

Description automatically generated with medium confidence

**Falsy dan Truthy**

**Data Falsy**

Falsy or sometimes written falsey, s a value which when in a boolean context, it is considered false. This is one of the unique features of JavaScript, which is useful, but sometimes confusing too.

So in JavaScript, the condition can not only be boolean, but outside of boolean can also be, but we must know that some data is falsy, or is considered false.

Table

Description automatically generated

**Data Truthy**

Truthy is the opposite of false, where the data is considered true.  
It’s very simple to know that data is truthy, that is, data that is not false.

Text

Description automatically generated

**Logical Operators Non-Boolean**

We already know that the logical operators AND (&&) and OR (||) are used for two boolean data. However, in JavaScript, we can use logical AND and OR operators for non-boolean data types

**Operator OR (**|| **) in Non-Boolean**

The logical OR operator (||), reads from left to right.  
This operator will take the first truthy value.  
If none of them are truthy, then the last one will be taken

Text

Description automatically generated

**Operator AND (&&) in Non-Boolean**

The logical AND operator (&&), reads from left to right.  
This operator will take the first false value.  
If none of them has a false value, then the last one will be taken

Text

Description automatically generated

**For Loop**

For is one of the keywords that can be used to do a loop. The code block contained in the for loop will always be repeated as long as the for a condition is met

**For Loop Syntax**

for(init statement; condition; post statement){  
 *// loop block*   
}

* The init statement will be executed only once at the beginning before the loop
* Conditions will be checked in each iteration, if true the loop will be executed, if false the loop will stop
* The post statement will be executed every time at the end of the loop
* Init statement, Condition, and Post Statement are not required, if the condition is not filled, it means that the condition always evaluates to true

**Infinity Loop**

An infinity loop will occur if the conditions in the loop are true with nothing false. For more details, see the code example below

Text

Description automatically generated

From the result code above, we will get a “Hello World” alert continuously even though it has been clicked.

**Looping With Init Statement, Condition, and Post Statement**

A loop with this condition will stop if the loop condition evaluates to false. Please see the example below

Text

Description automatically generated

From the loop above, we will console from numbers 1 to 10

**While Loop**

While loop is a simpler version of the loop than the for a loop. In the while loop, there are only looping conditions, no init statement, and no post statement.

For more details, see the sample code below

Text

Description automatically generated

Maybe that’s my first post. For basic JavaScript, it will be posted later in part 4, in here [Introduction and Basics of JavaScript | Part 4](https://denislistiadi.medium.com/introduction-and-basics-of-javascript-part-4-41629ee1b588)