Vincent Luk

Last Update | @October 12, 2021 10:38 AM

MD Notes

Heading 1

Heading 2

Heading 3

Heading 4

Heading 5

Heading 6

Divide Line - - -

Code Block ``` language ```

```
document.write('Hello World!');
```

Loosely typed language

A loosely typed language is a programming language that does not require a variable to be defined. For example, Perl is a loosely typed language, you can declare a variable, but it doesn't require you to classify the type of variable. In the example below, the first line declares the $test variable that can be used as an integer or string.

```
var message = 'Hello World';  // Right!
message = 30; //variable is now integer.
var 2age = 30; // Wrong! The variable name cannot start with a number.
```

Strongly typed language

A programming language that requires a variable to be defined, and the variable it is. For example, C is a strongly typed language. When declaring the variable, you must also specify the variable type.

```
int num = 3;
double num1= 1.5;
String name = "vincent";
```

JavaScript

💡 **Primitive: In JavaScript, all types other than Object are immutable (the value cannot be changed). We call these types of value as “primitive values”.**

💡 **All global variables are the attribute of the object window.**

```
<script type="text/javascript">
    var x = 100 ;
    alert( window.x );//100 will pop up.
    alert(x);
</script>
```

Variable Shadowing

In JavaScript, variables with the same name can be specified at multiple layers of nested scope. In such case local variables gain priority over global variables. If you declare a local variable and a global variable with the same name, the local variable will take precedence when you use it inside a function. This type of behavior is called shadowing.

Scope (Var / let / Const)

So, in javaScript, variables declared with var or created by function declarations in non-strict mode do not have block scope.

```
var x = 1;
{
  var x = 2;
}
console.log(x); // logs 2
```

By contrast, identifiers declared with **let** and **const** do have block scope.

```
let x = 1;
{
  let x = 2;
}
console.log(x); // logs 1

const c = 1;
{
  const c = 2;
}
console.log(c); // logs 1
```

String

In JavaScript, a string can be indicated with **double quotes (") or a single quote (')**, so the following two strings are both valid:

```
var firstName = 'Hello';
var lastName = "World";
```

- `\n`: Linefeed
- `\t`: Tab
- `\b`: Backspace
- `\r`: Carriage return
- `\\`: Slash（\）
- `\'`: Single quote（'）
- `\"`: Double quotes（"）

includes() method – string search

includes() method is used to determine whether a string is included in another string and return true or false as the case may be. This method is case sensitive. For example:

```
'Blue Whale'.includes('blue'); // false (case sensitive)
'Blue Whale'.includes('Blue'); // true
```

Different between substr() & substring()

```
//substr (start, length)
var str = 'abcdefghij';
str.substr(0,3); // 'abc'
str.substr(3,3); // 'def'
str.substr(3); // 'defghij'

//substr (Start, End)
var str = 'abcdefghij';
str.substring(0,3); // 'abc'
str.substring(3,3); // ''  ( There is no character between 3 to 3.)
str.substring(3); // 'defghij'
str.substring(2,3); // 'c'
```

Array

Map

```
// Arrow function
map((element) => { ... })
map((element, index) => { ... })
map((element, index, array) => { ... })

// Callback function
map(callbackFn)
map(callbackFn, thisArg)

// Inline callback function
map(function callbackFn(element) { ... })
map(function callbackFn(element, index) { ... })
```

```
map(function callbackFn(element, index, array){ ... })
map(function callbackFn(element, index, array) { ... }, thisArg)
```

Reduce

Reduce all the array items into one output

=== & ==

Three equal signs means strict equivalence, not only in value but also in data type. However, two equal signs represent equivalence in value only, not in data type.

```
alert('0' === 0); // false  Strict equivalence. ’0‘ is a string, while 0 is a number, so it is not strictly equal.
alert('0' == 0); // true  Not strict equivalence. ’0‘ will be converted into 0 through implicit conversion. By comparison, 0==0, so true wi
```

Git

Git add .

Git commit -m "msg"

Git push branch master

Git remote -v

git remote rm origin

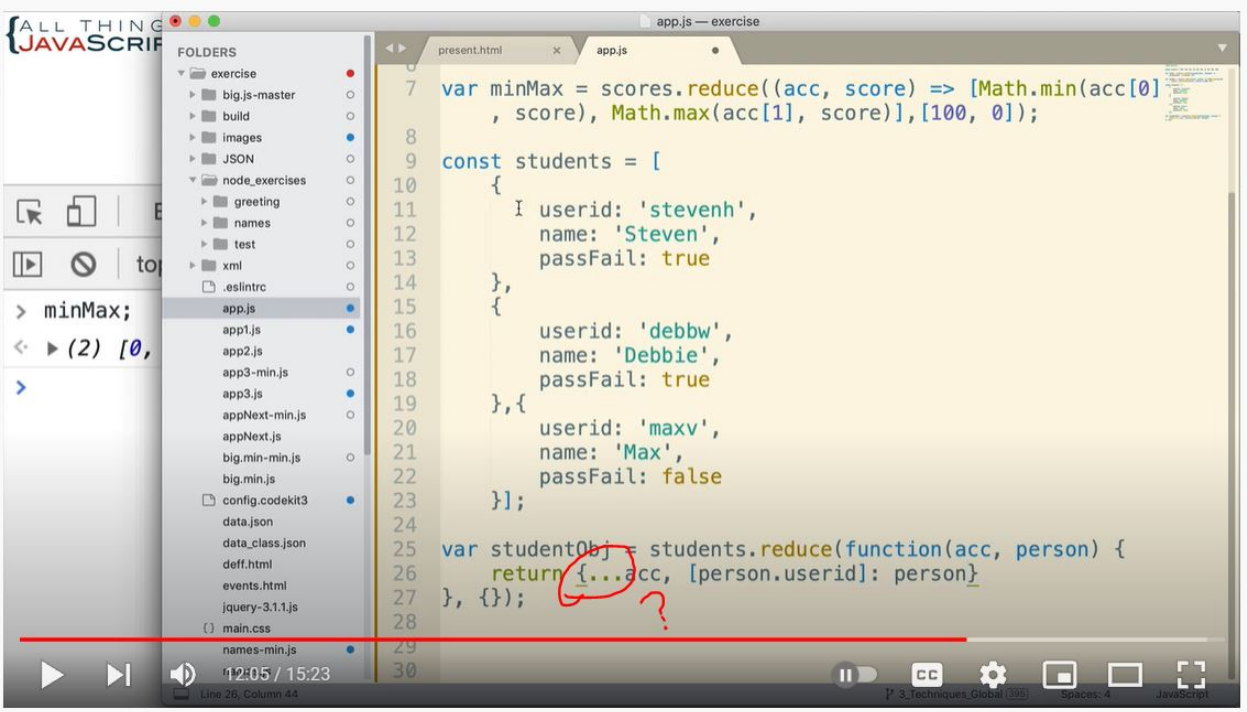
Command Line

Zip folder

Compress-Archive .\JavaScript-String-english-2019-10-11-9-32-19-0\ ex5.zip

Questions :

Q1 what is the name of ... ?



#javascript #AllThingsJavaScriptLLC
The Magic of the reduce Array Method