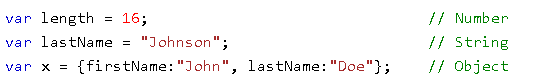
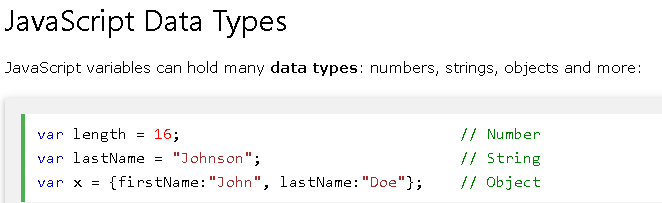
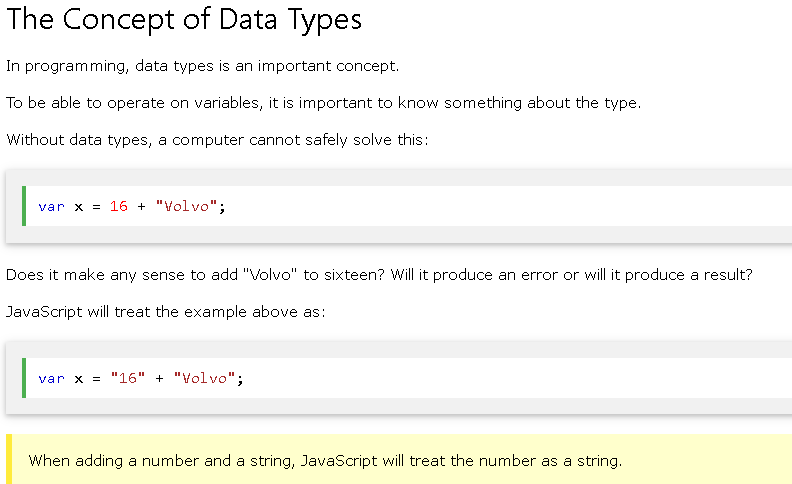
# **Data Types**

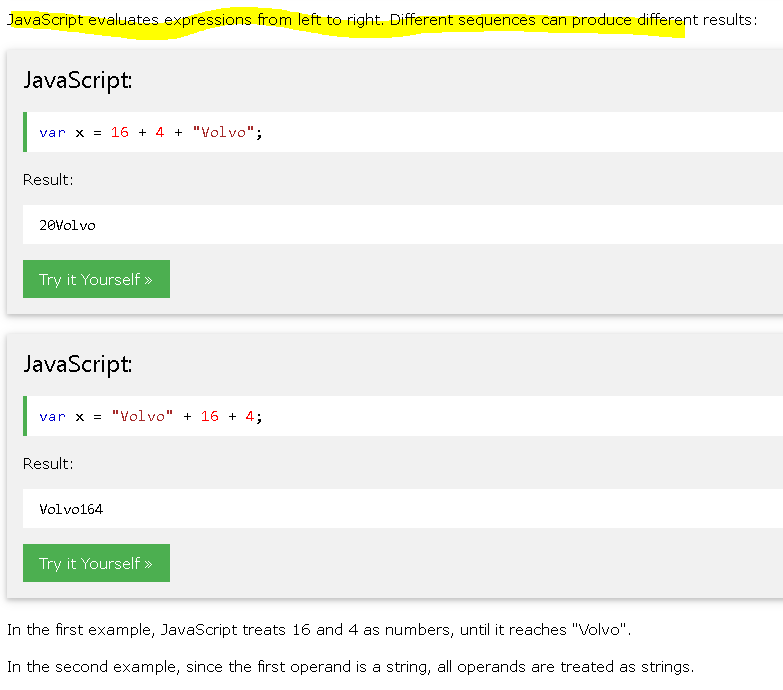
A variable in JavaScript can contain any data. A variable can at one moment be a string and at another be a number or object

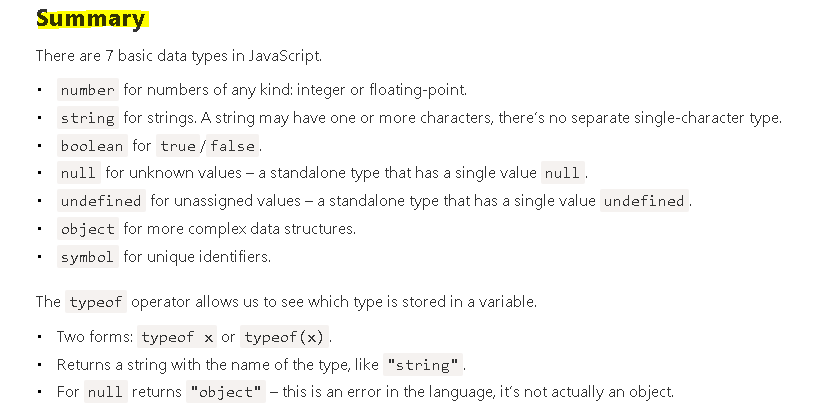


Programming languages that allow such things are called “dynamically typed”, meaning that there are data types, but variables are not bound to any of them.

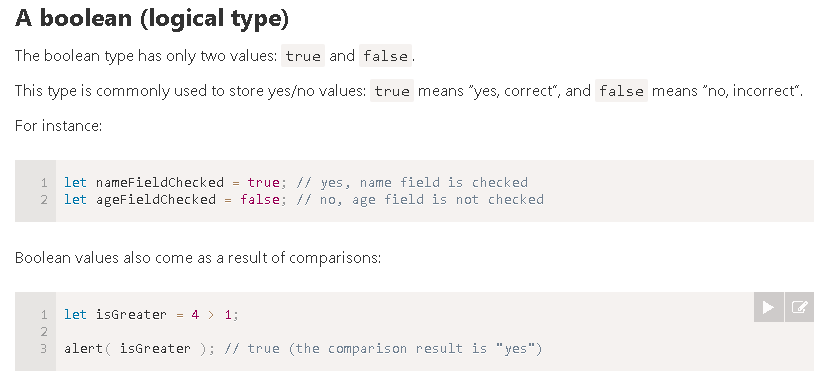




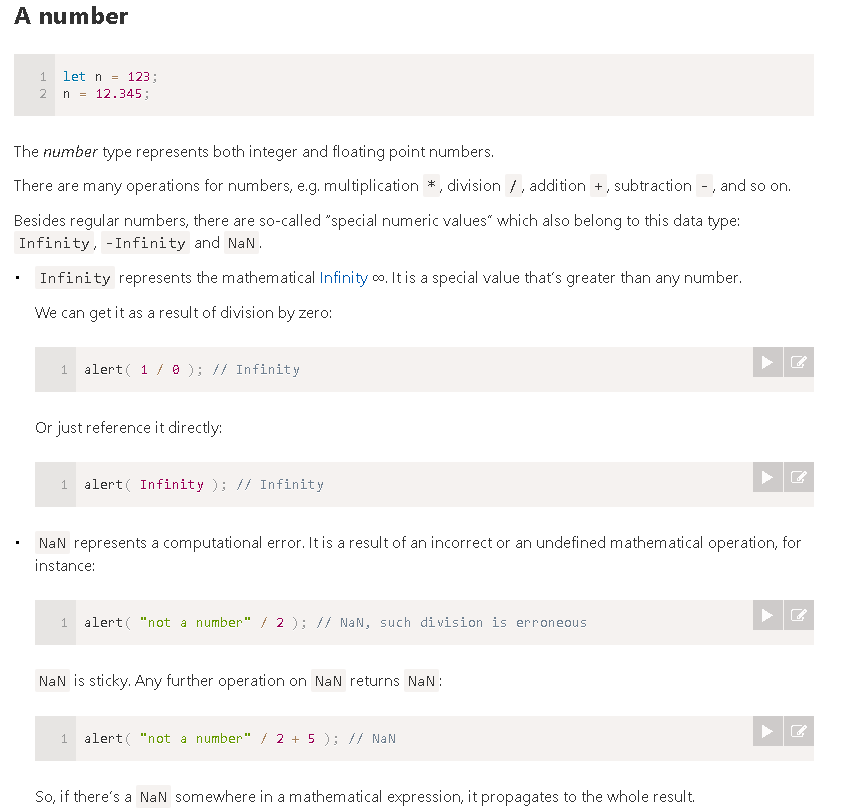


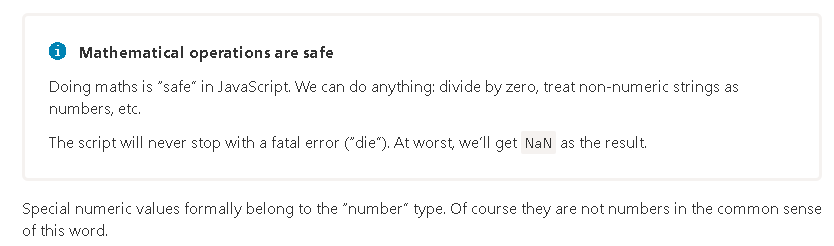


## **Boolean**



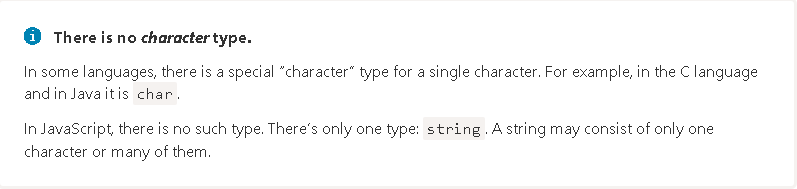
## **Number**

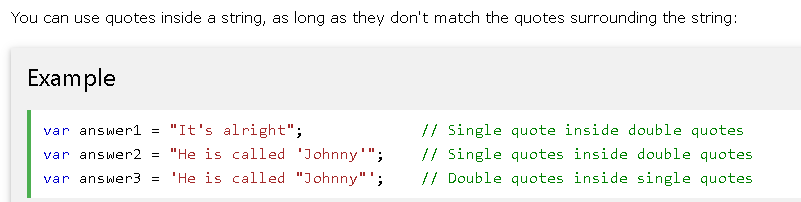




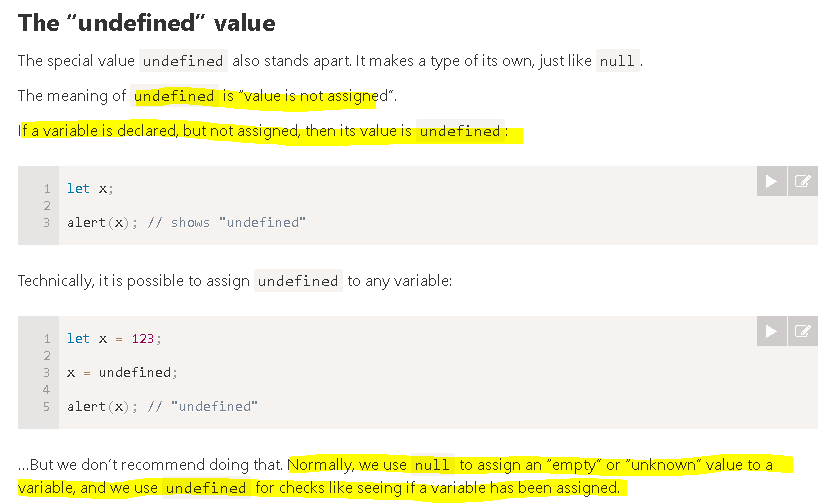
## **String**

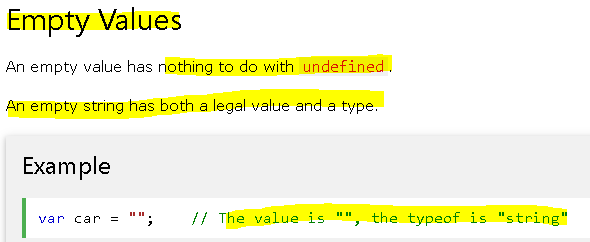




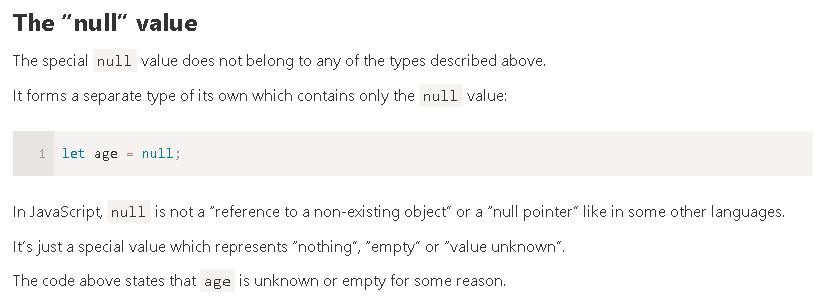


## **Undefined**

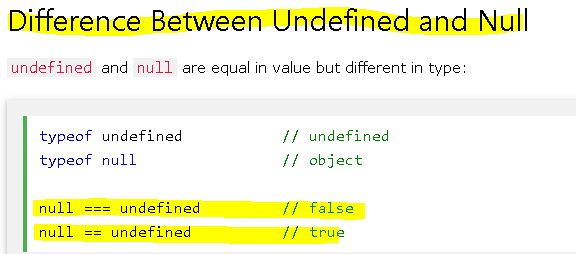




## **Null**



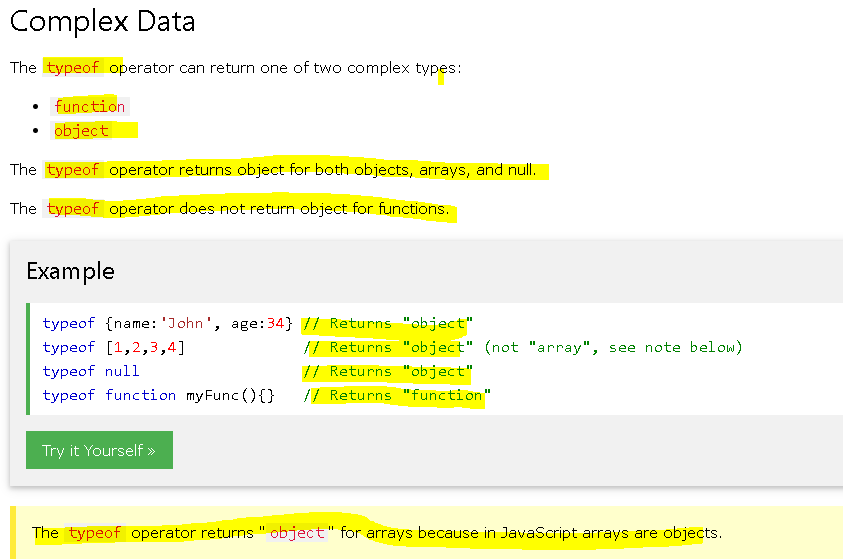




## **Object**

The object type is special.

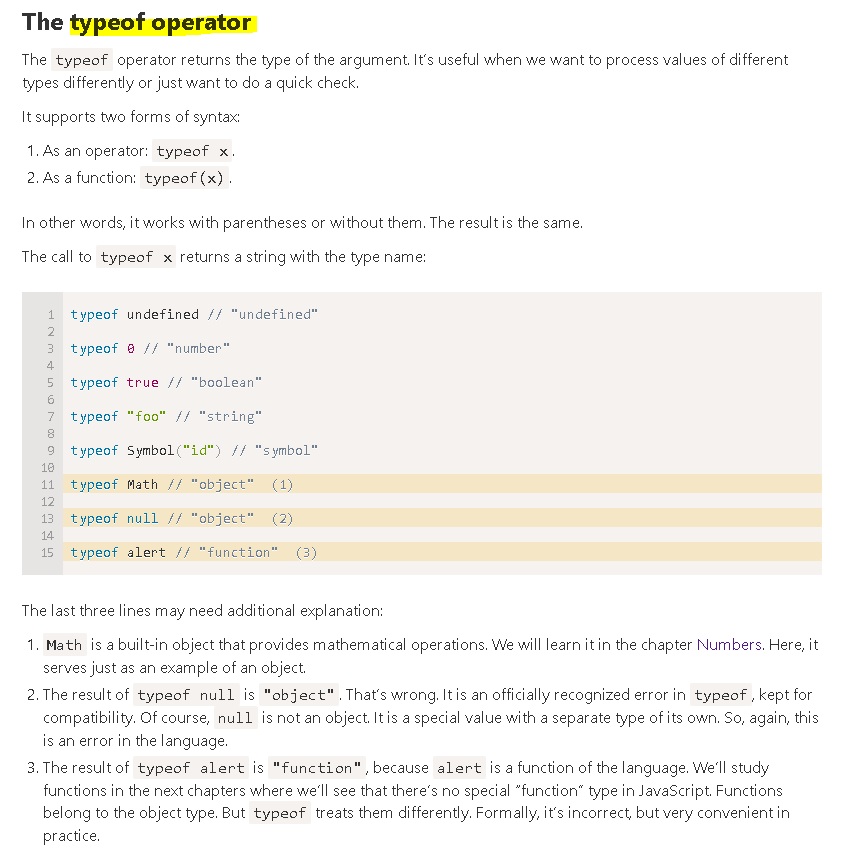
All other types are called “primitive” because their values can contain only a single thing (be it a string or a number or whatever). In contrast, objects are used to store collections of data and more complex entities.



## **Symbol**

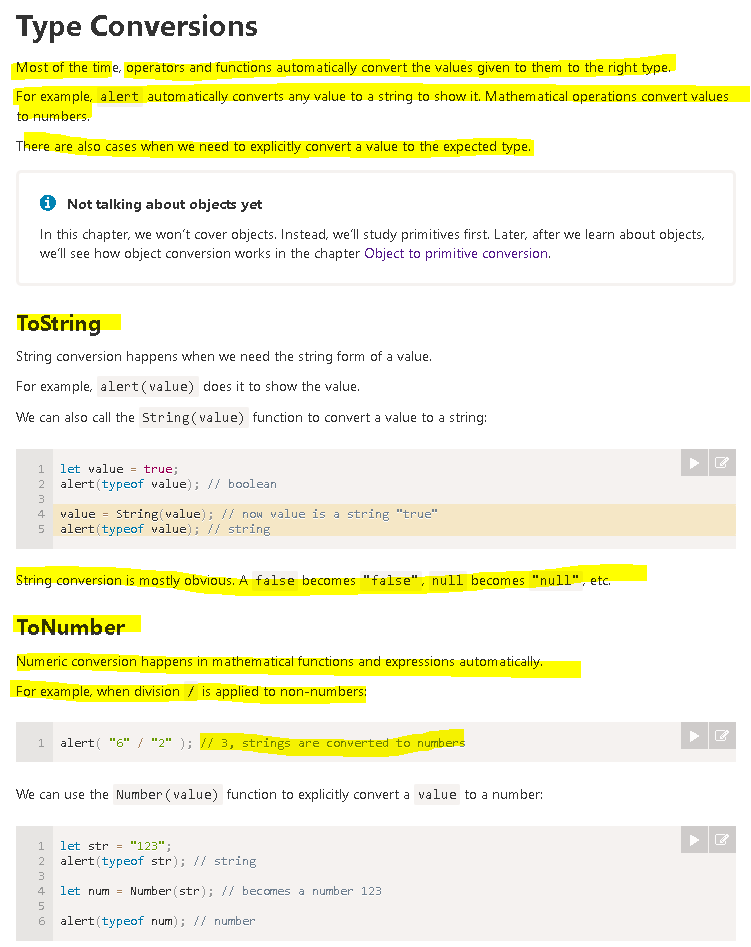
The symbol type is used to create unique identifiers for objects. We have to mention it here for completeness, but it’s better to study this type after objects.

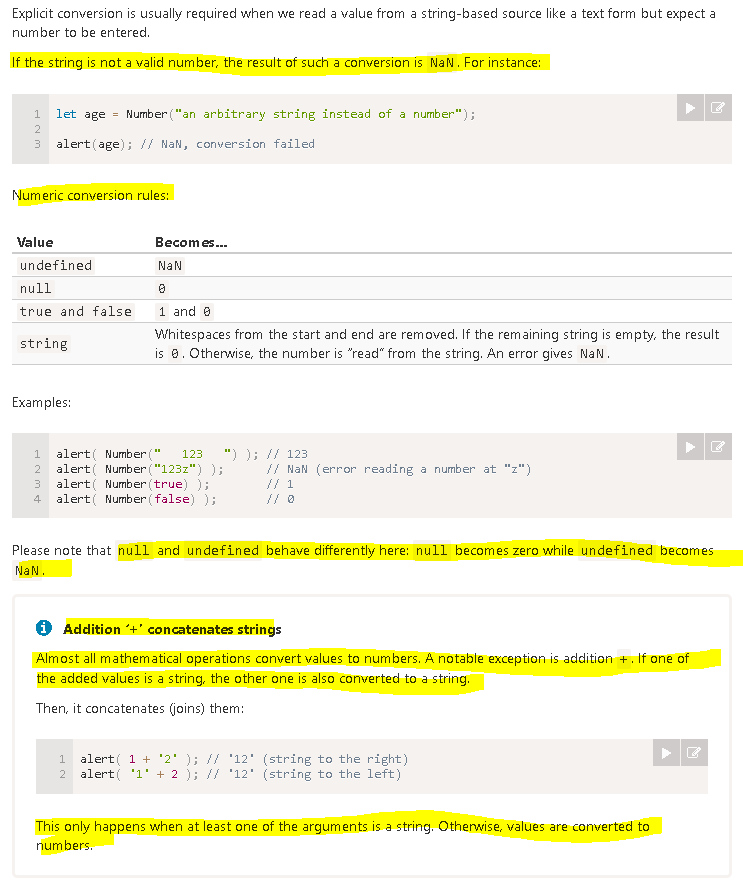
## **typeof Operator**

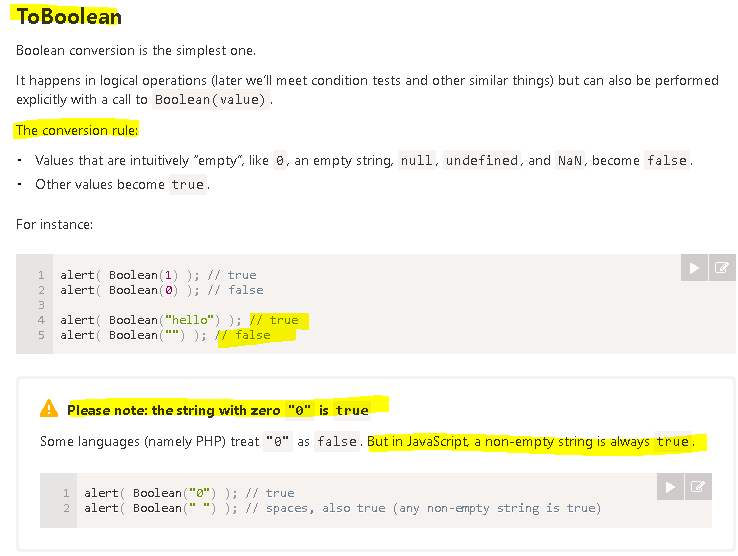


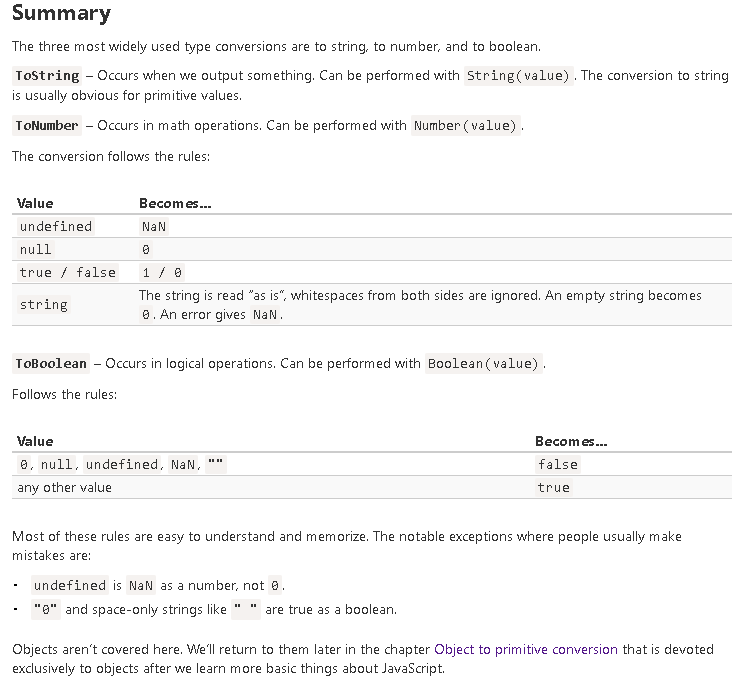
## **Summary**

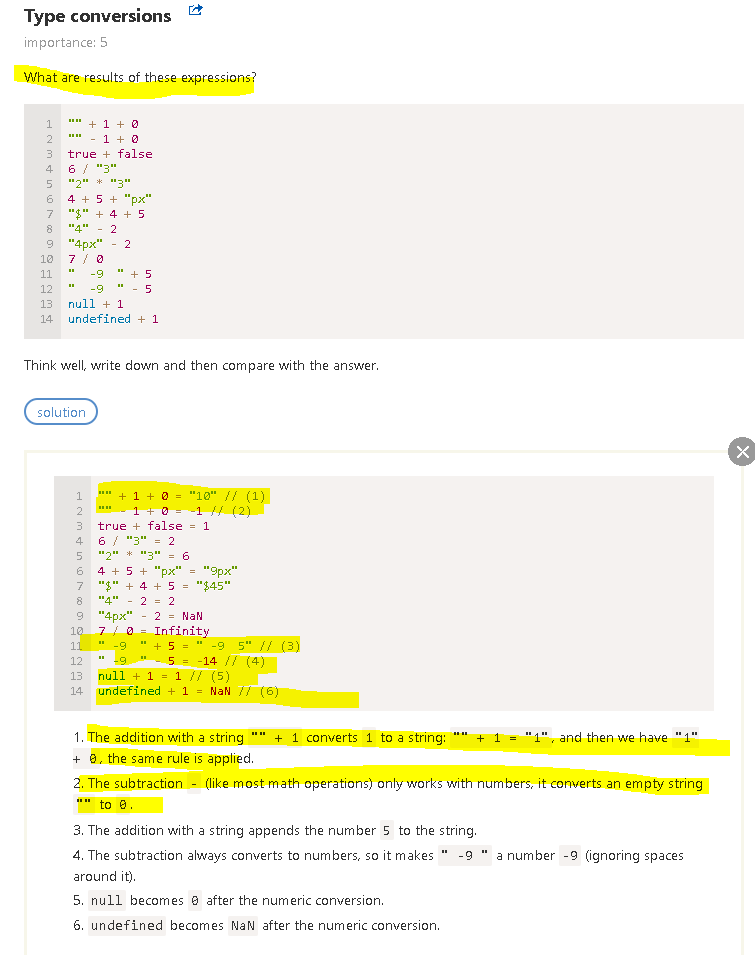
# **Type Conversions**



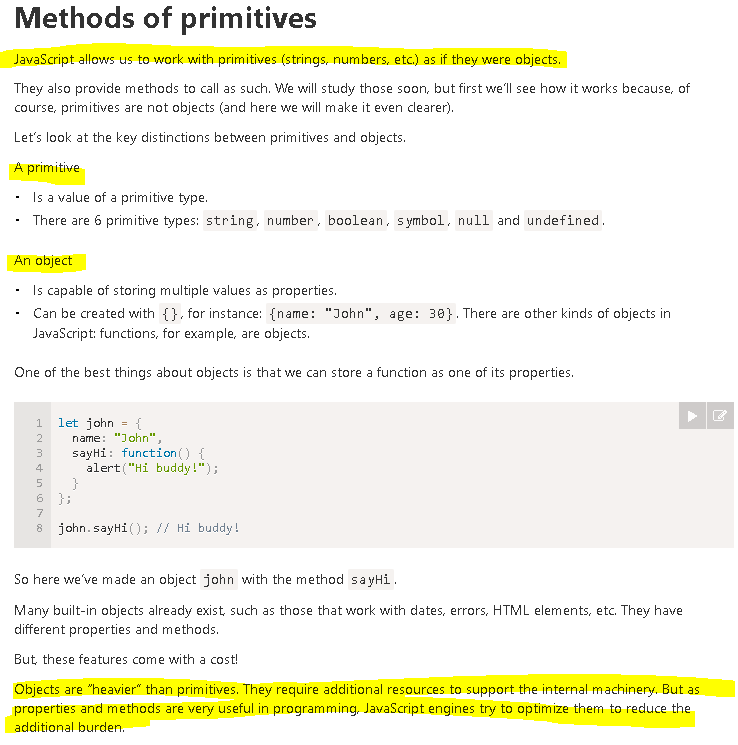


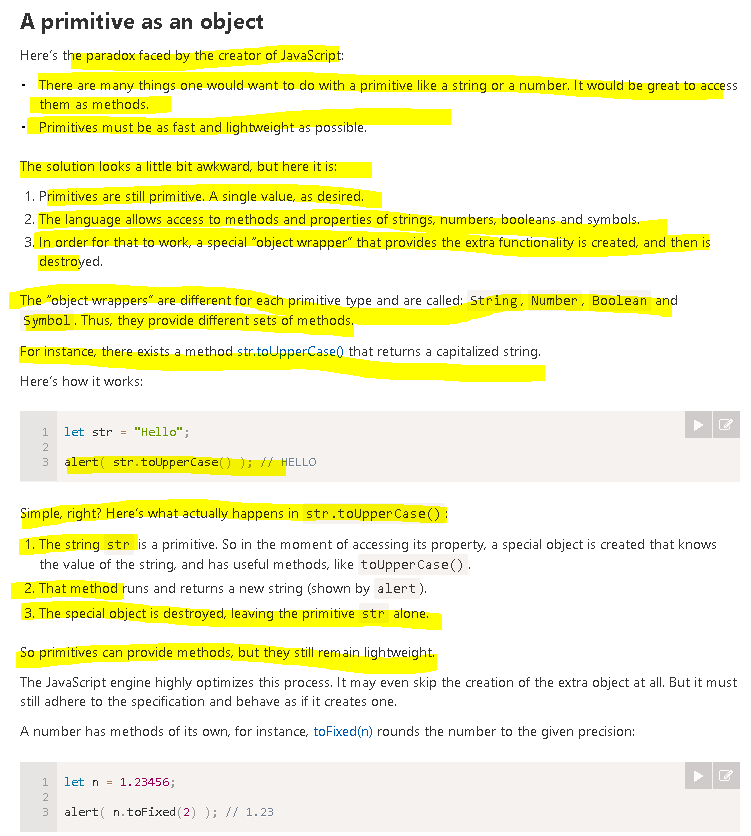




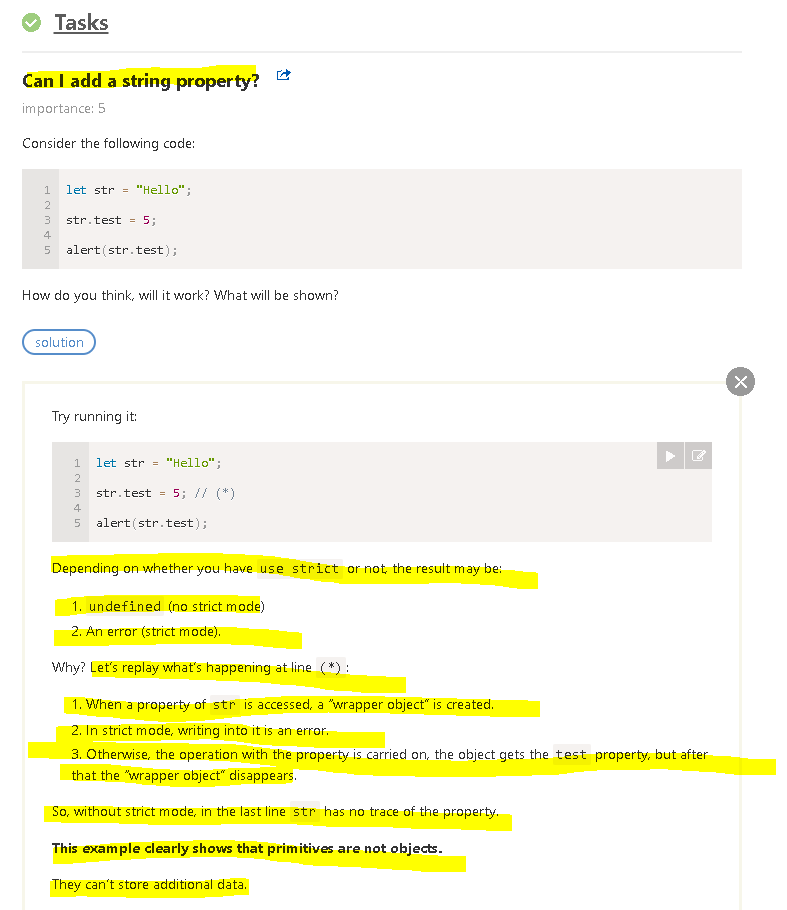


# **Methods of primitives**







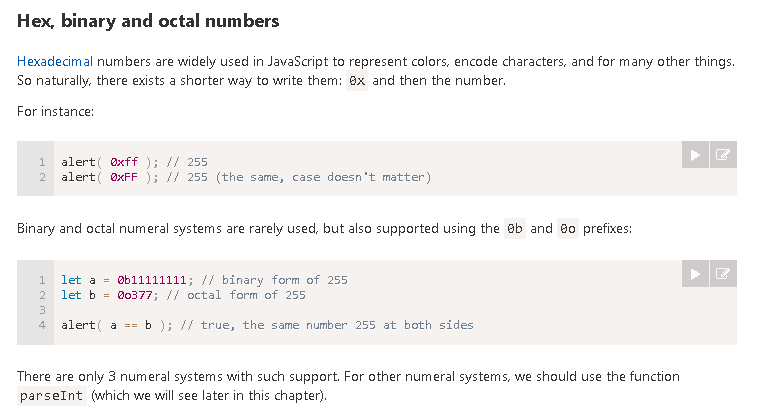


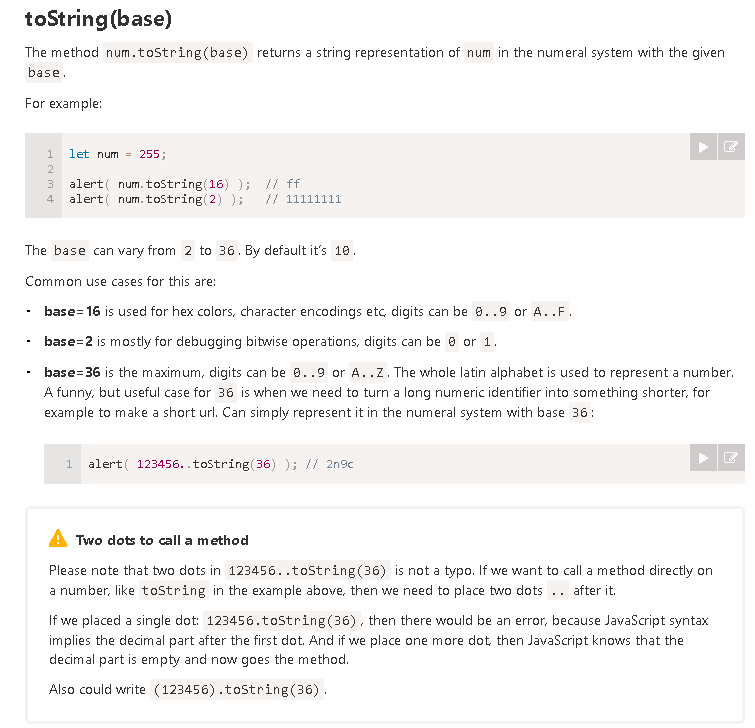
# **Numbers**

All numbers in JavaScript are stored in 64-bit format [IEEE-754](https://en.wikipedia.org/wiki/IEEE_754-2008_revision), also known as “double precision floating point numbers”.

Let’s recap and expand upon what we currently know about them.







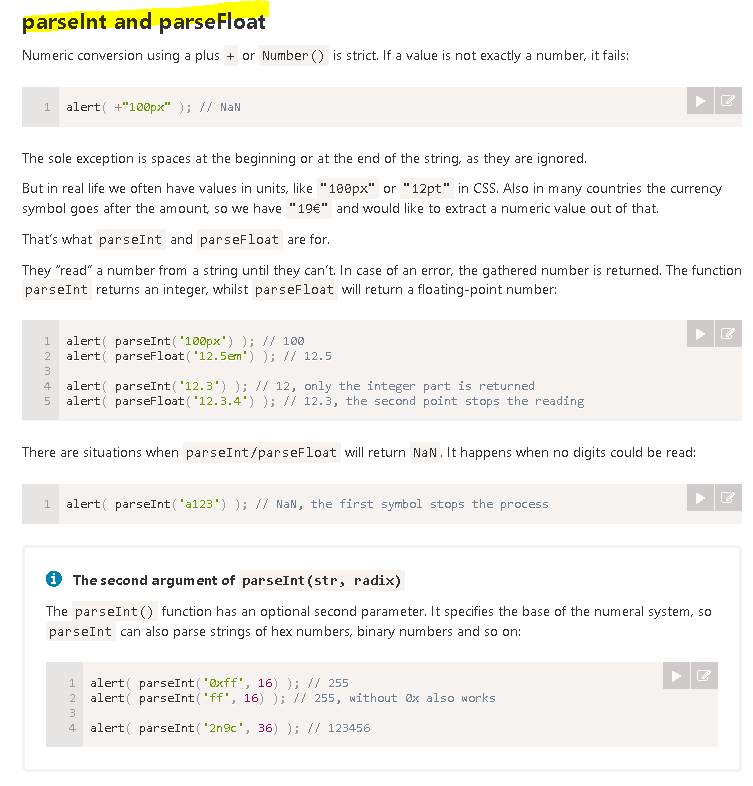


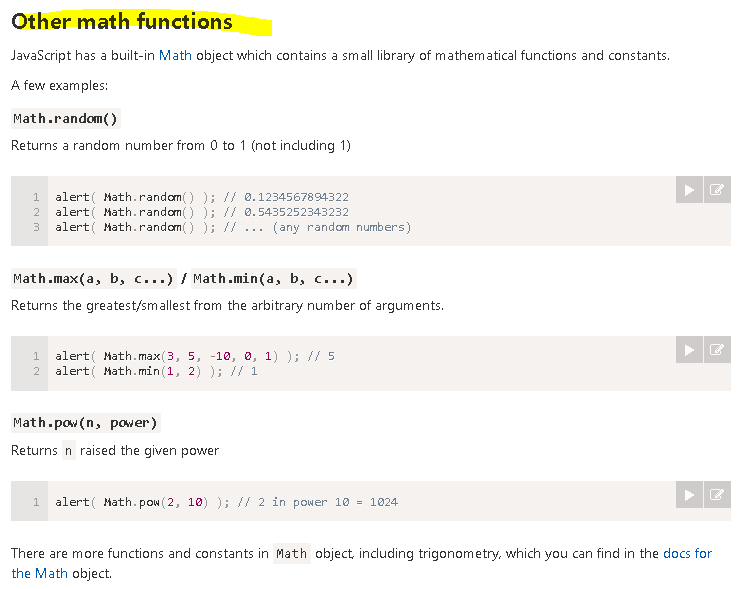


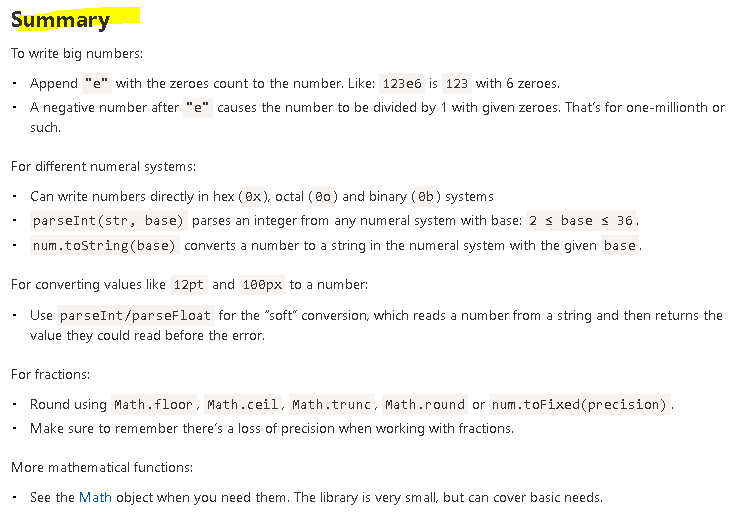




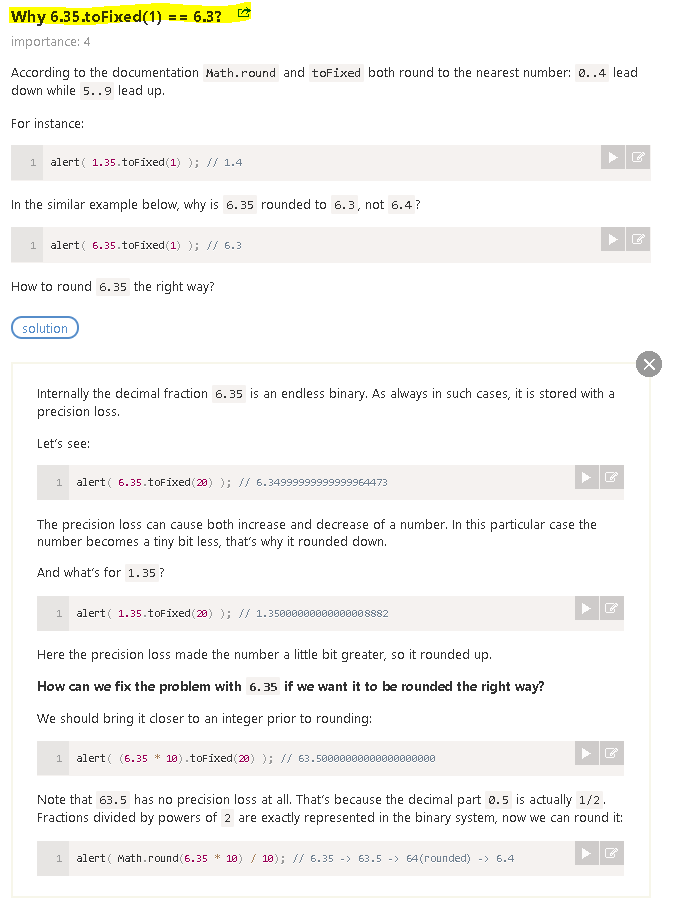


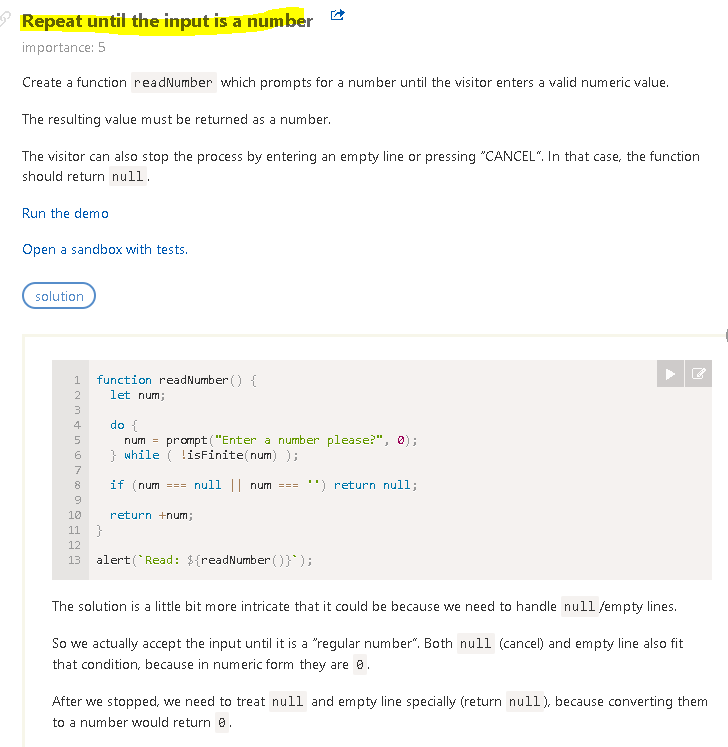


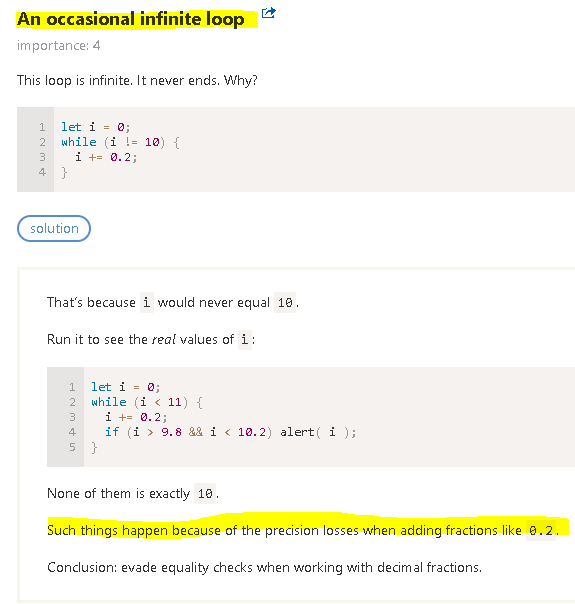


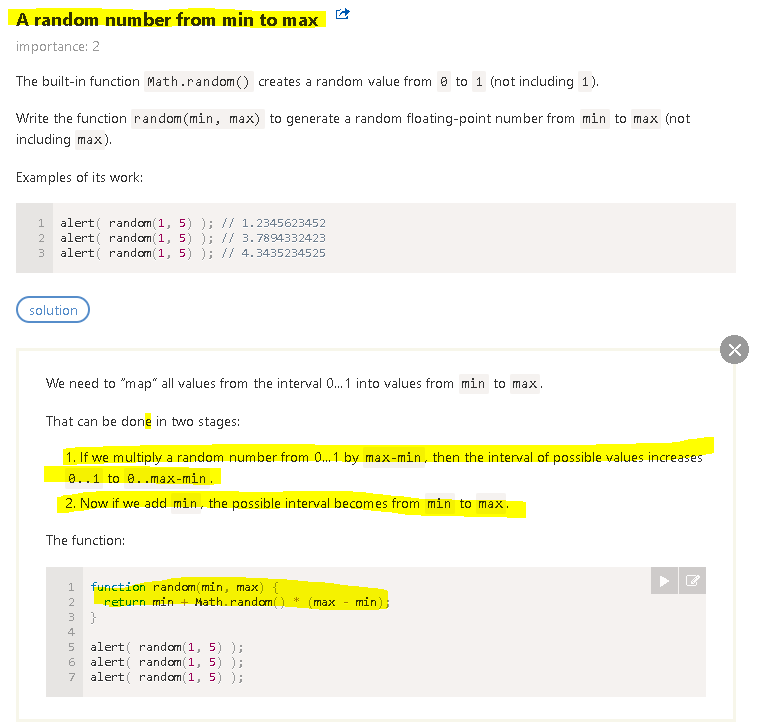


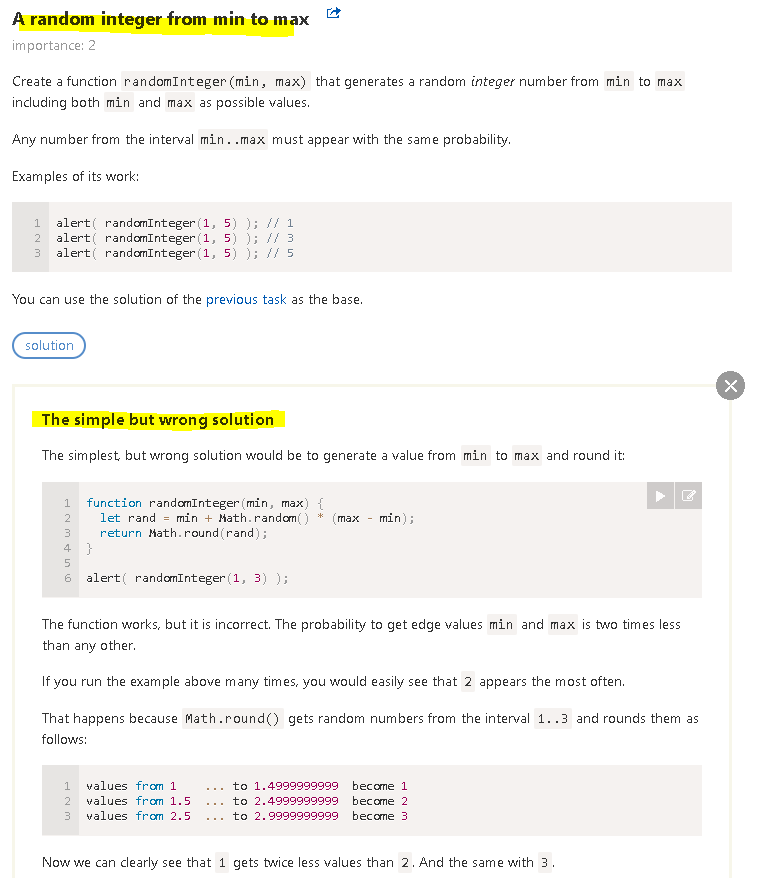
## **Coding Problem**







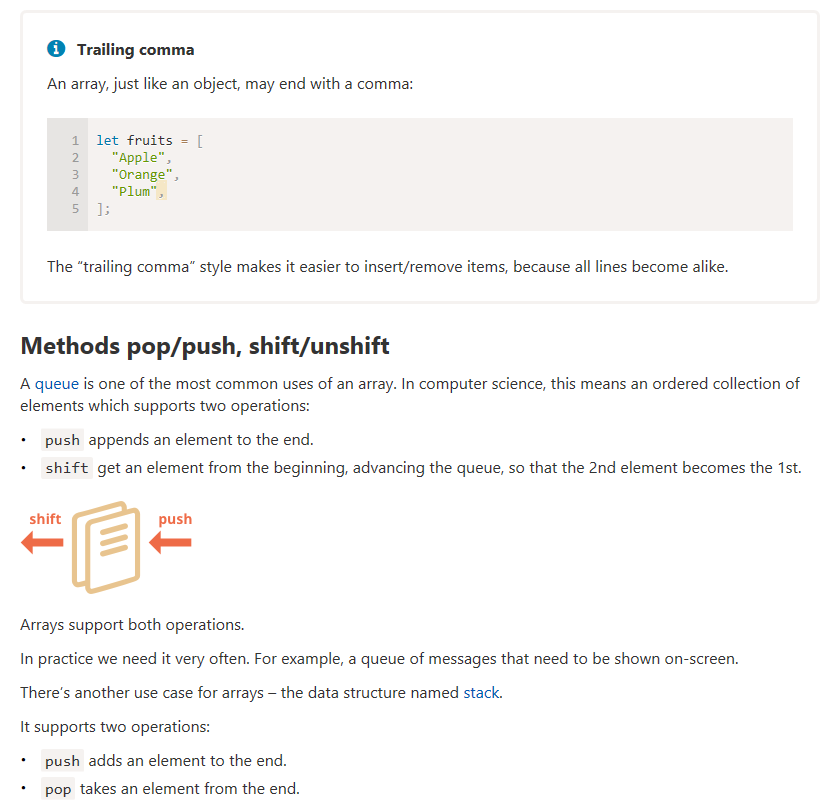


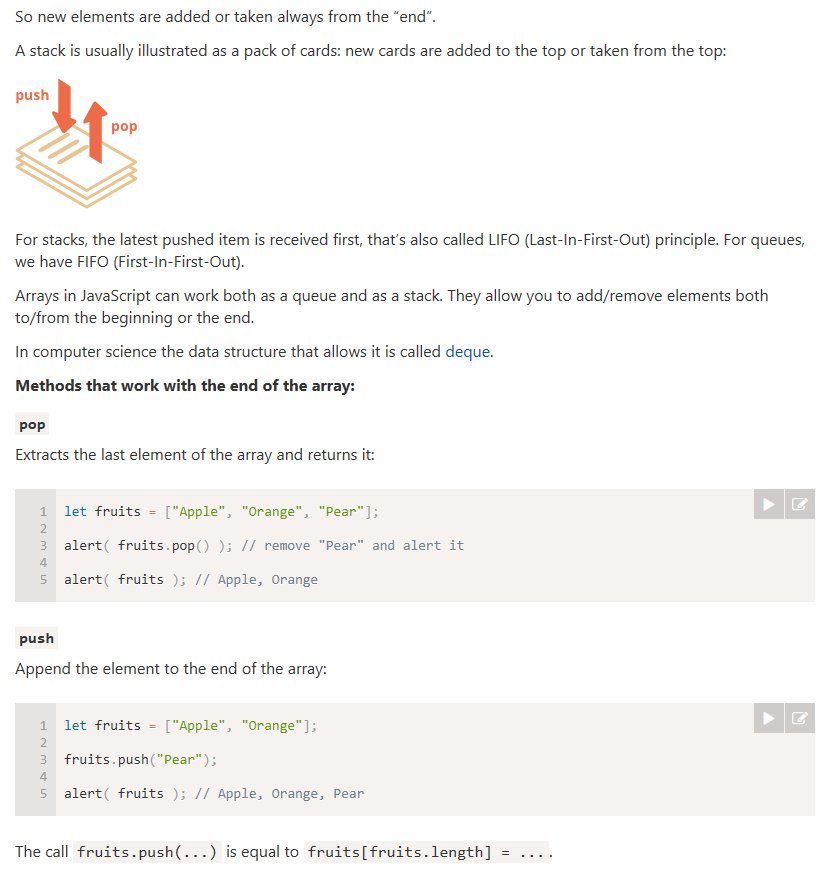




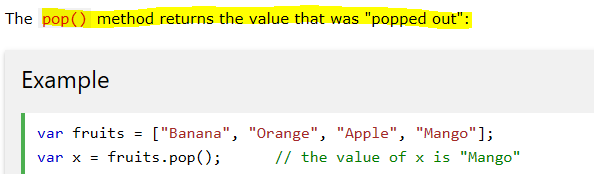
# **Arrays**

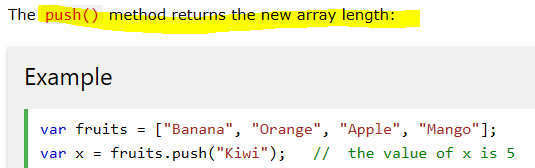




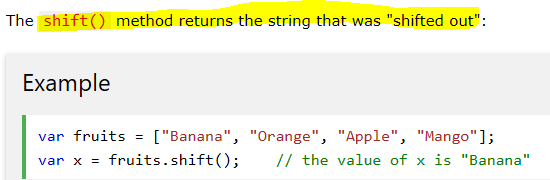


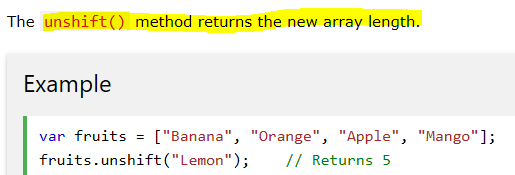


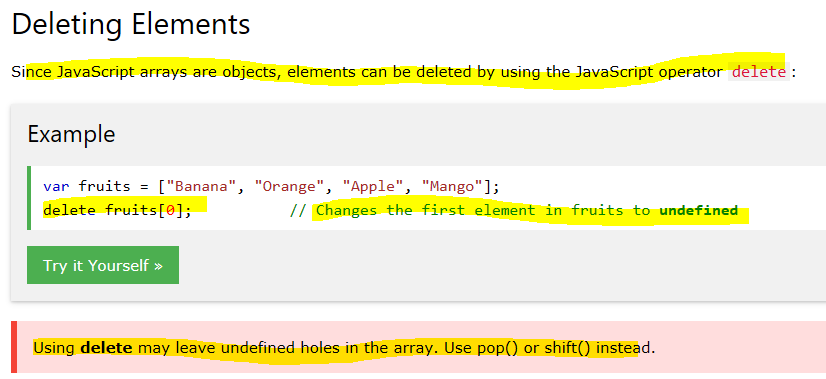


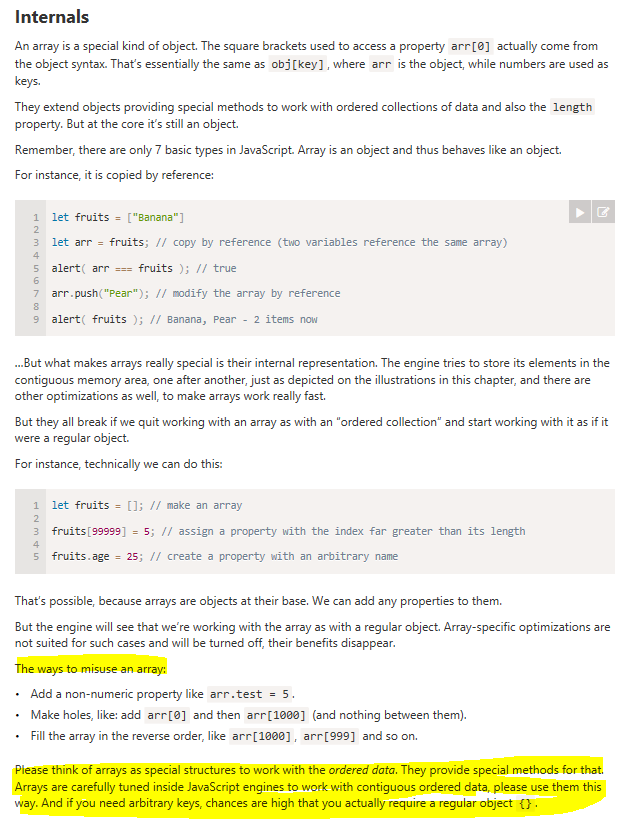


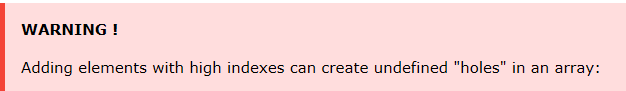


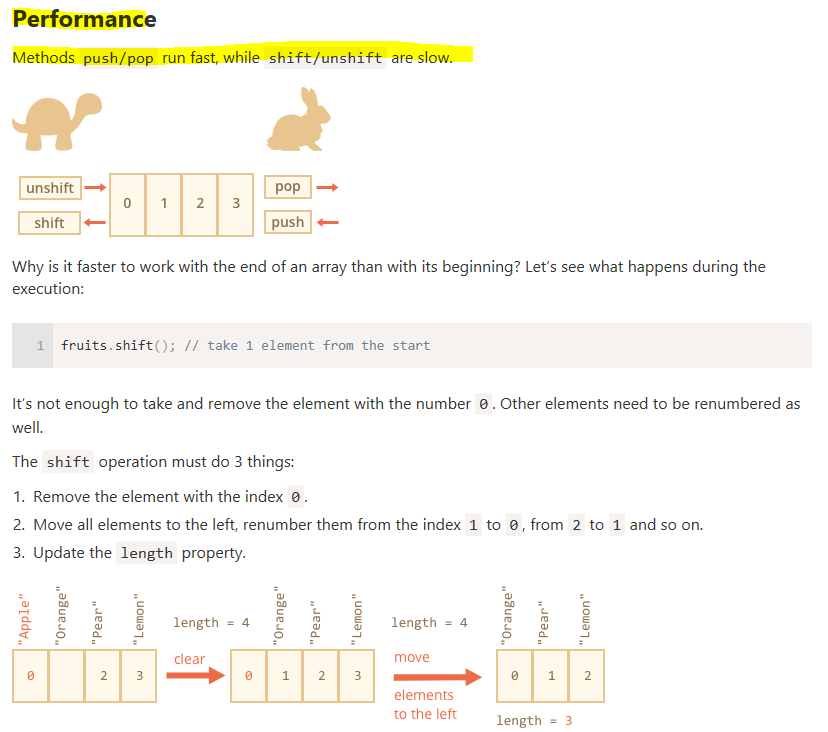


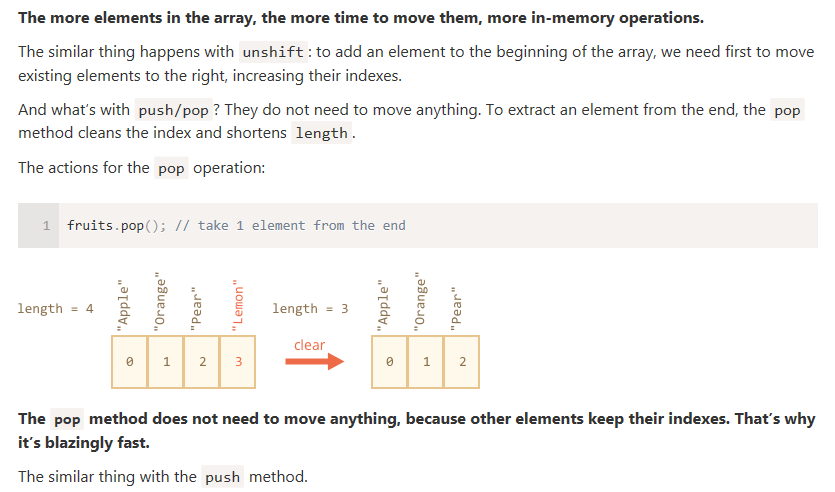


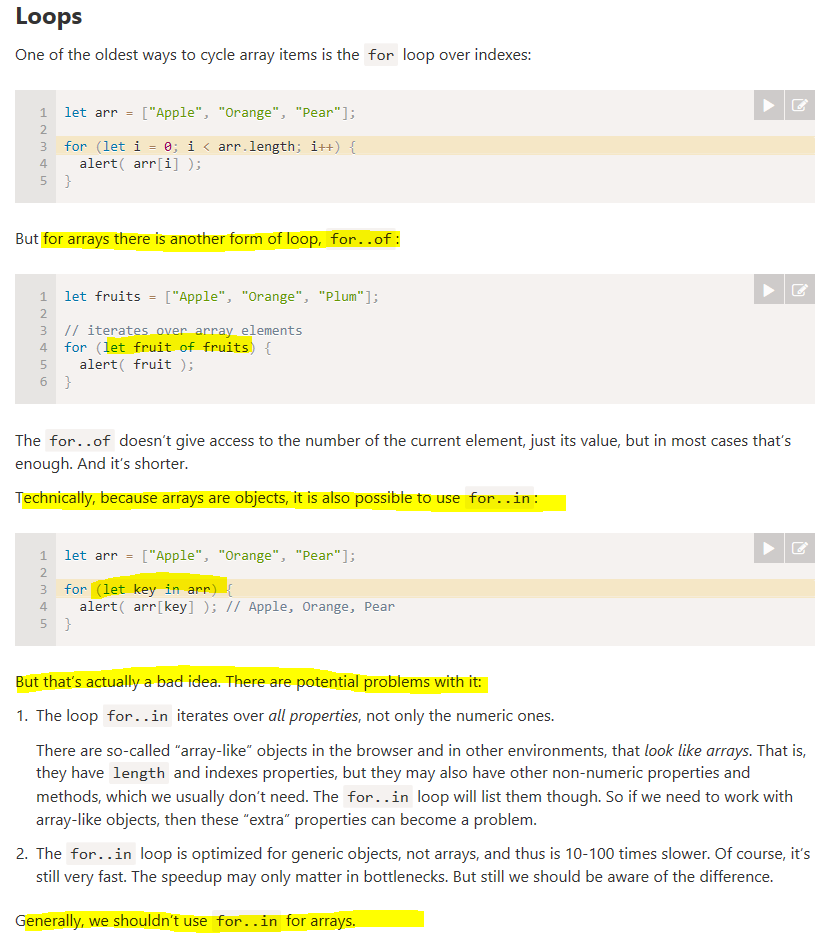


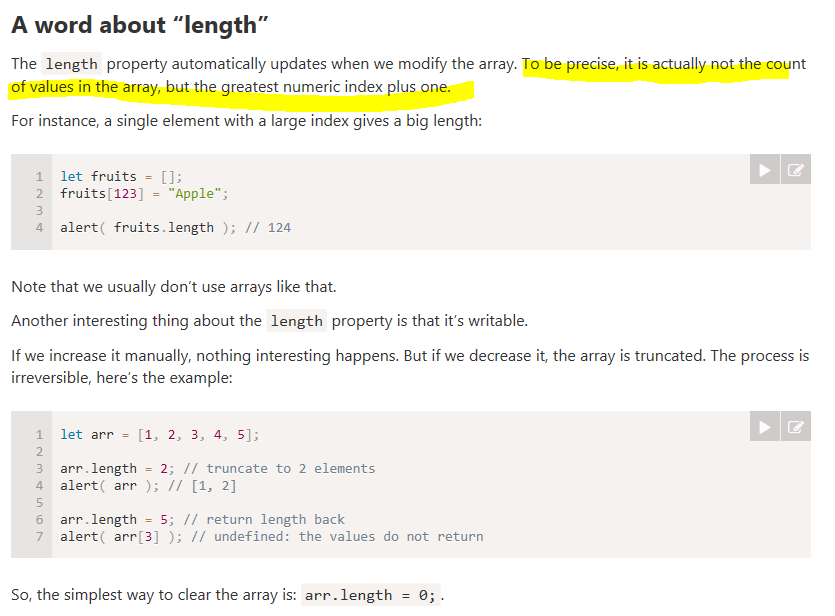




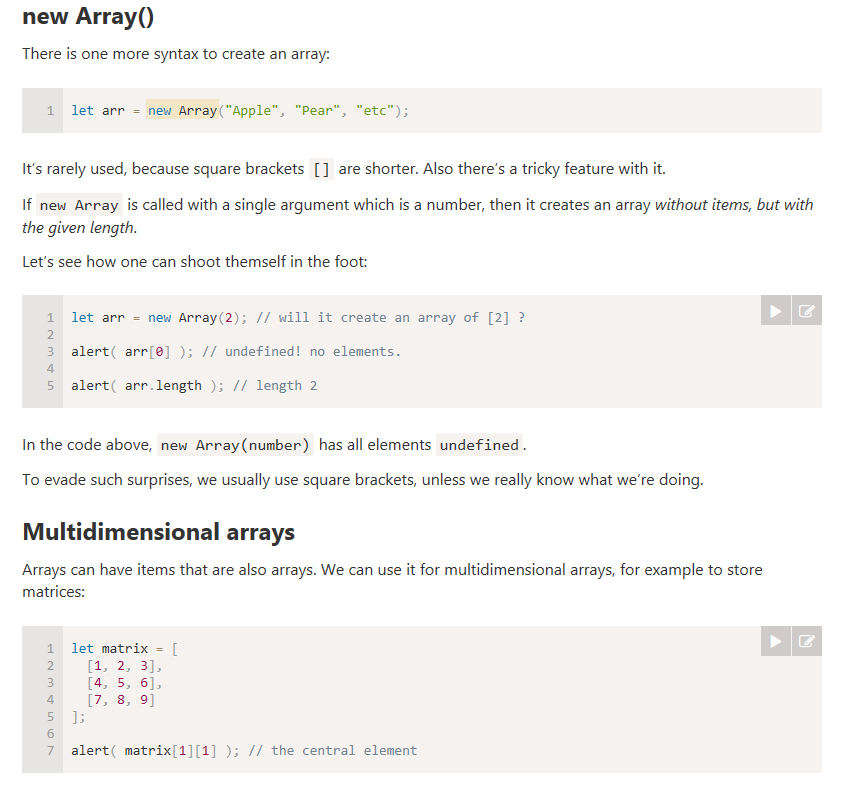






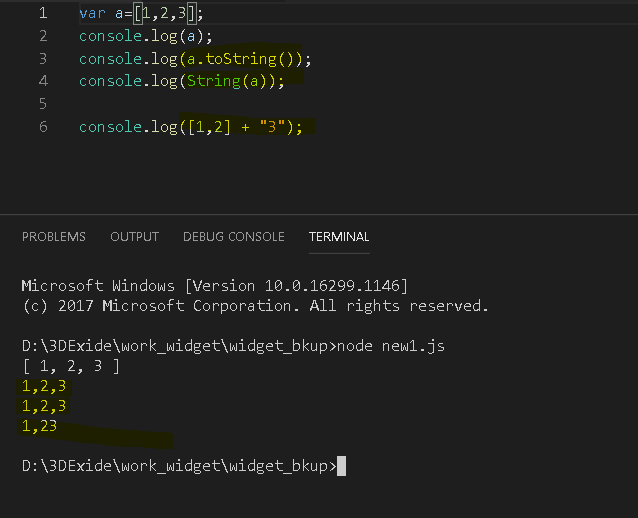


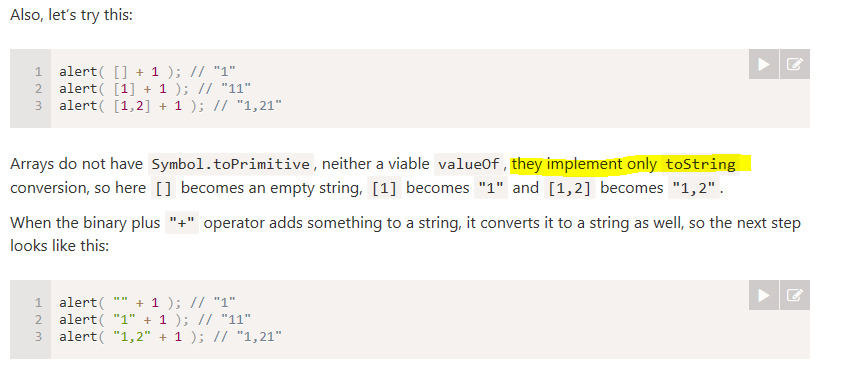




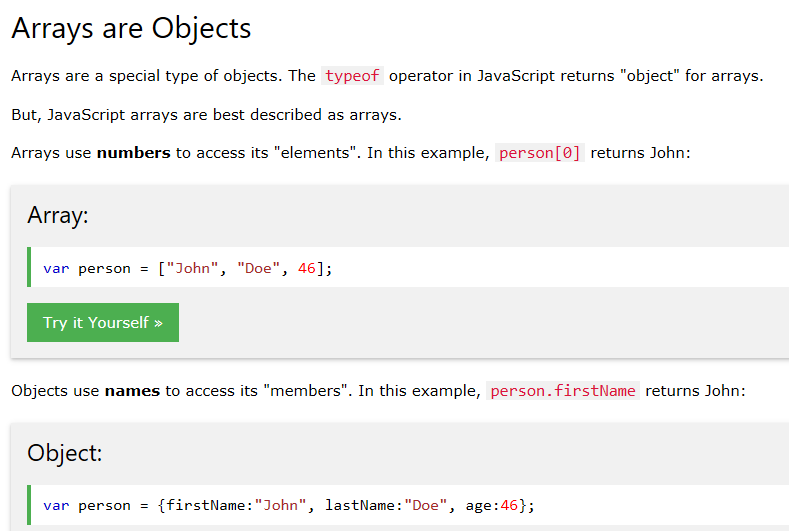
**[toString](https://javascript.info/array" \l "tostring)**

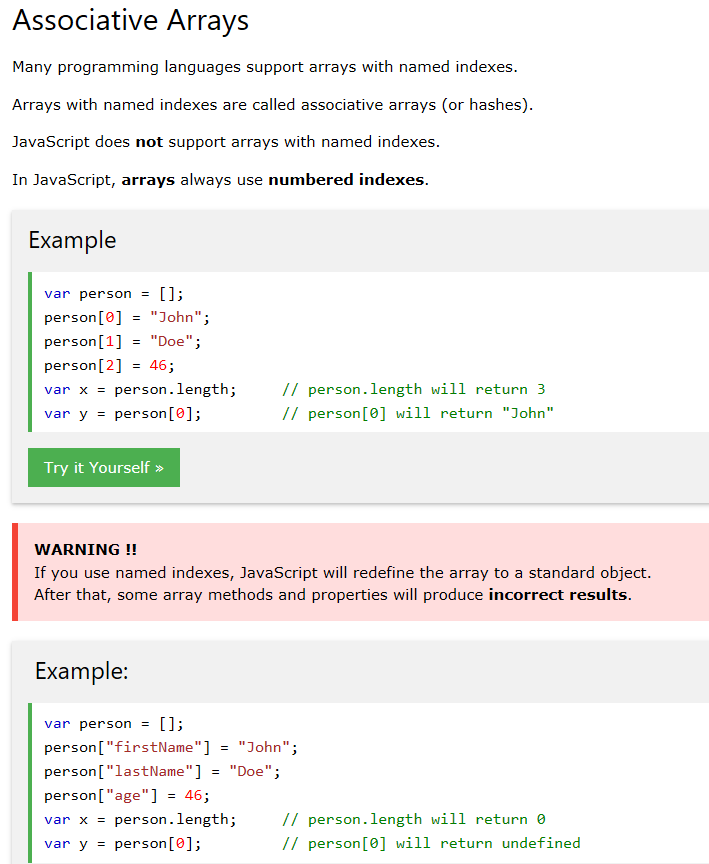
Arrays have their own implementation of toString method that returns a comma-separated list of elements.

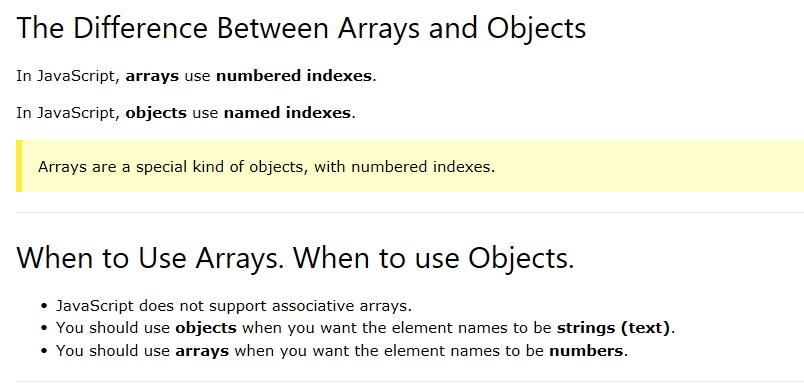
For instance: 

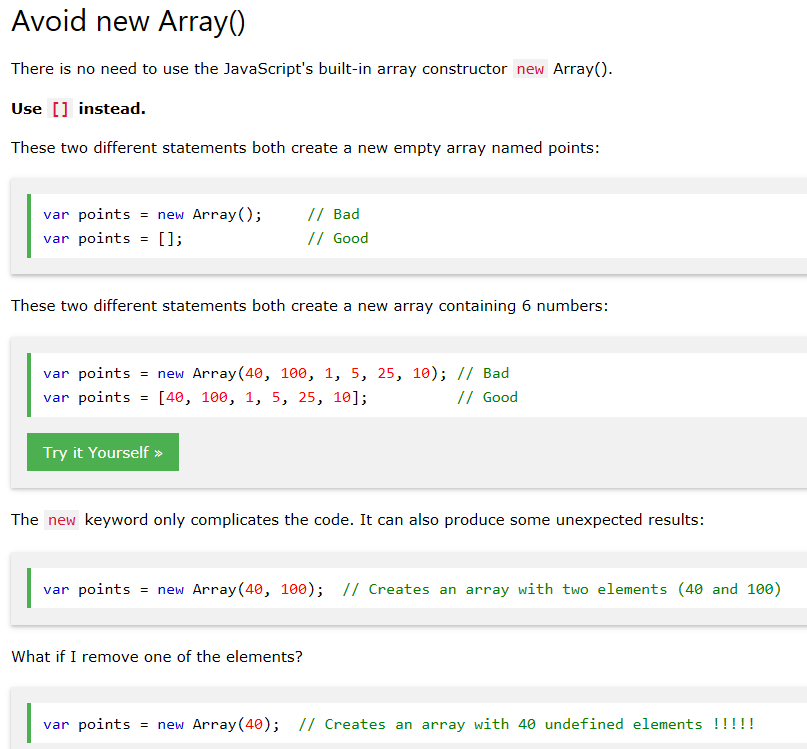




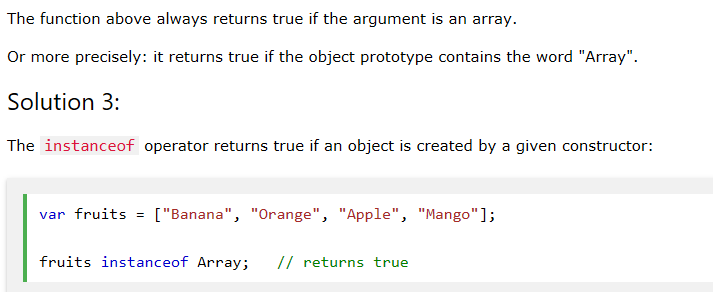


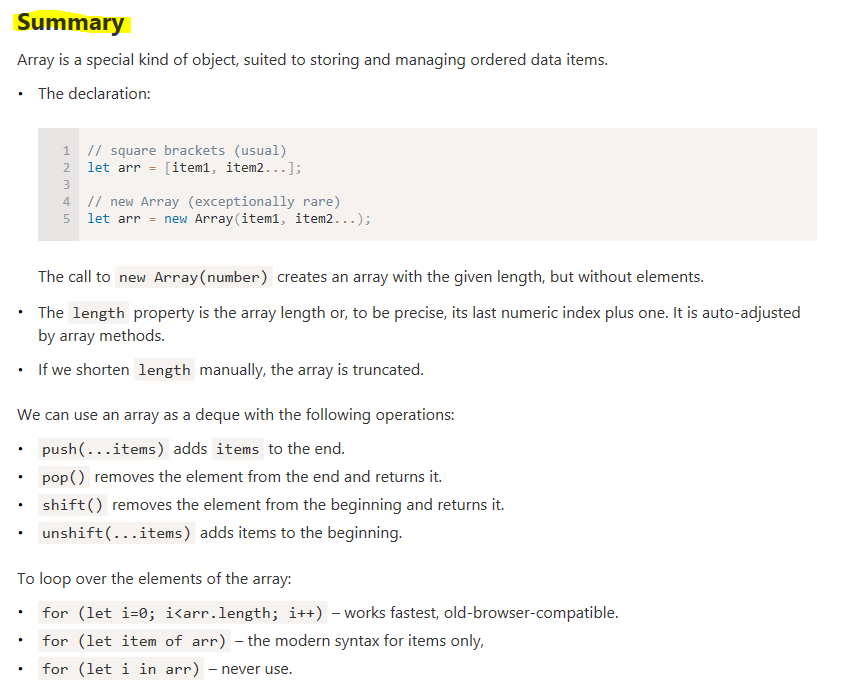






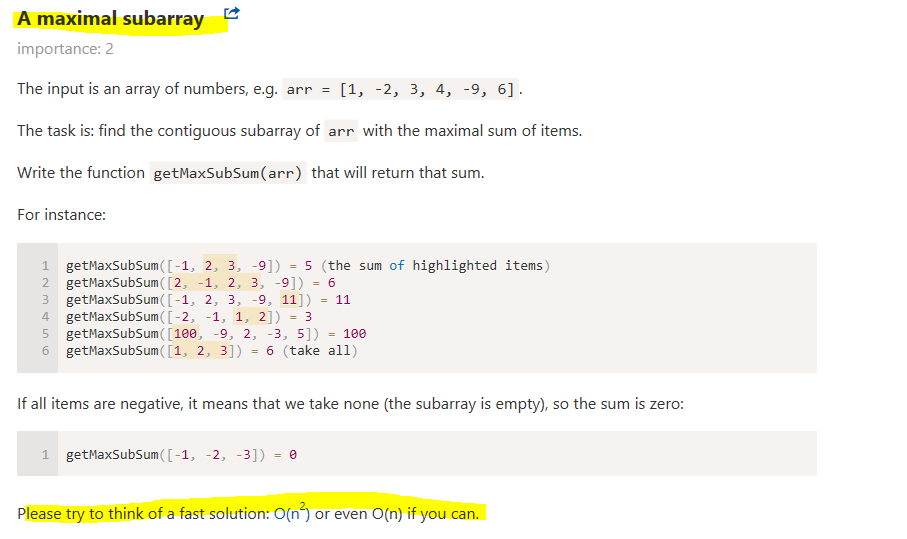




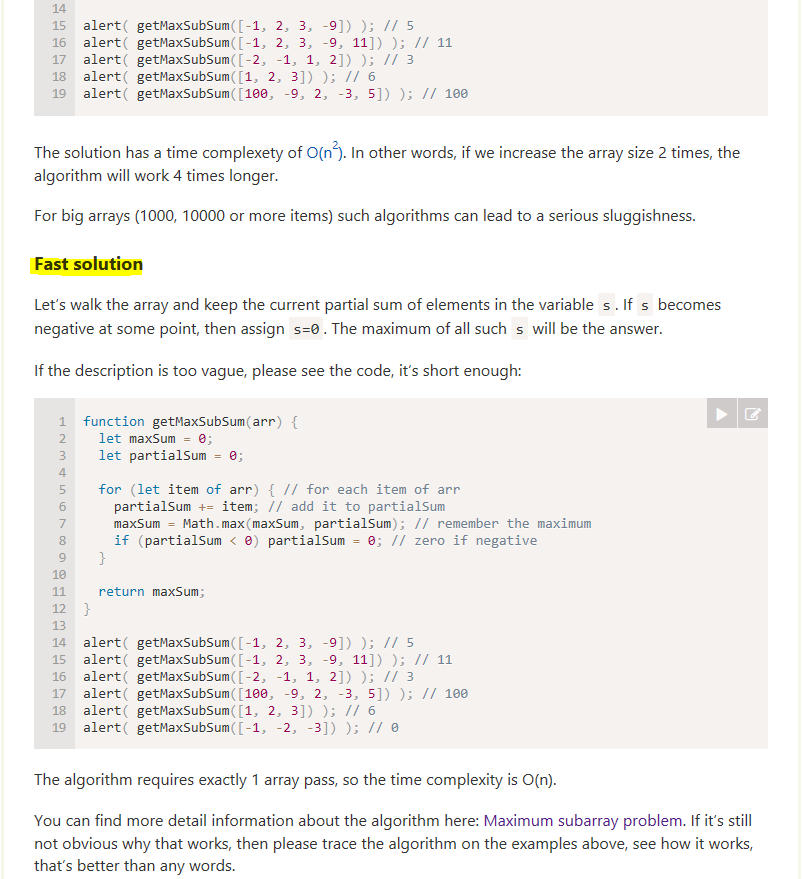


**Example -** 

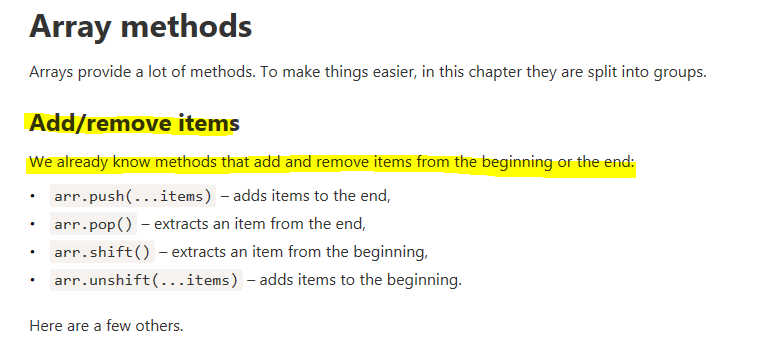


**Important Example -**





# **Array Methods**



## **Add/Remove items methods**

### **splice**



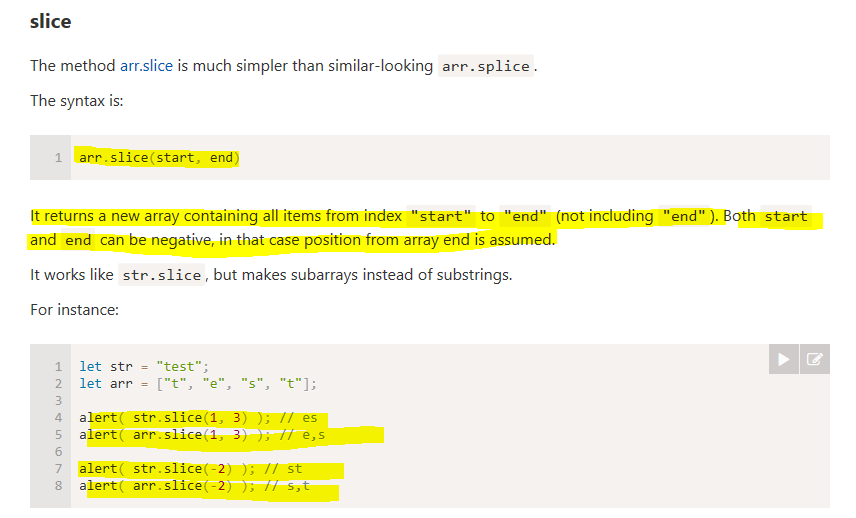


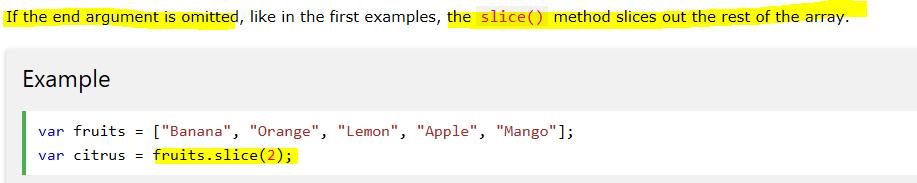


### **Slice**





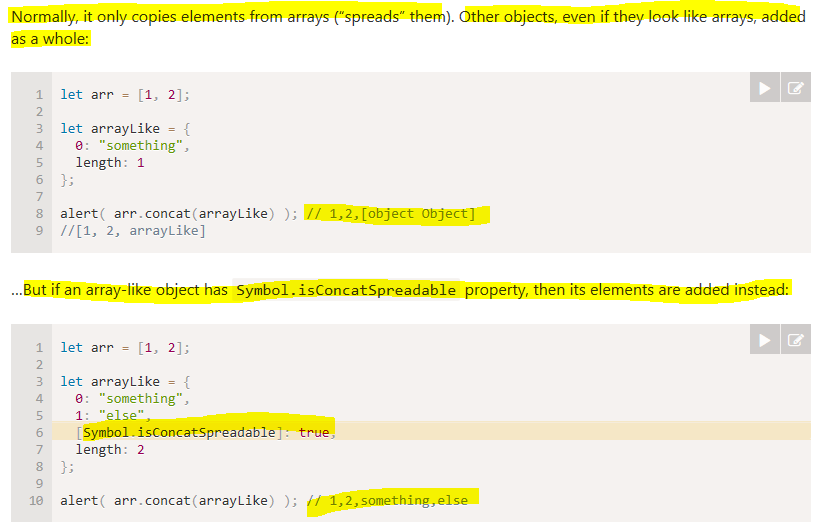




**Output -** 

### **Concat**

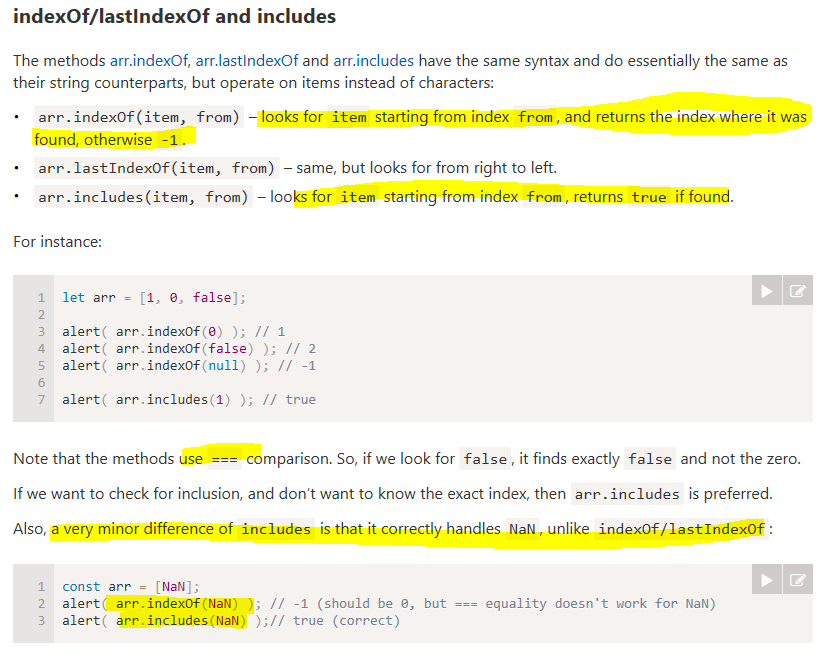




## **Iterate : forEach**

## **Searching in an Array**

### **indexOf/lastIndexOf/includes**





### **find/findIndex**

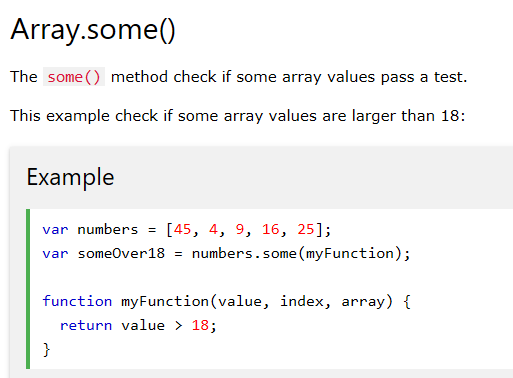


### **filter**

### **every**

output – false (allOver18 is false)

### **some**

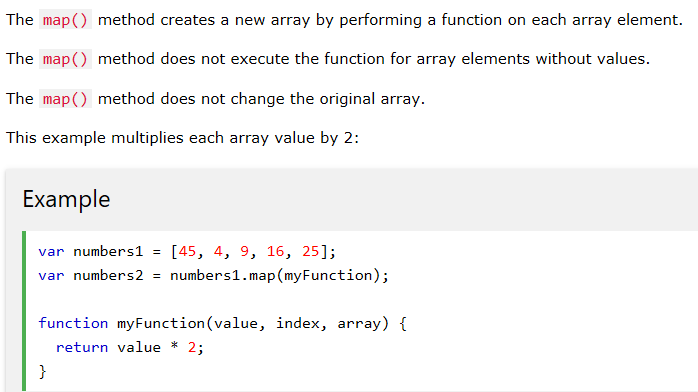


Output – true

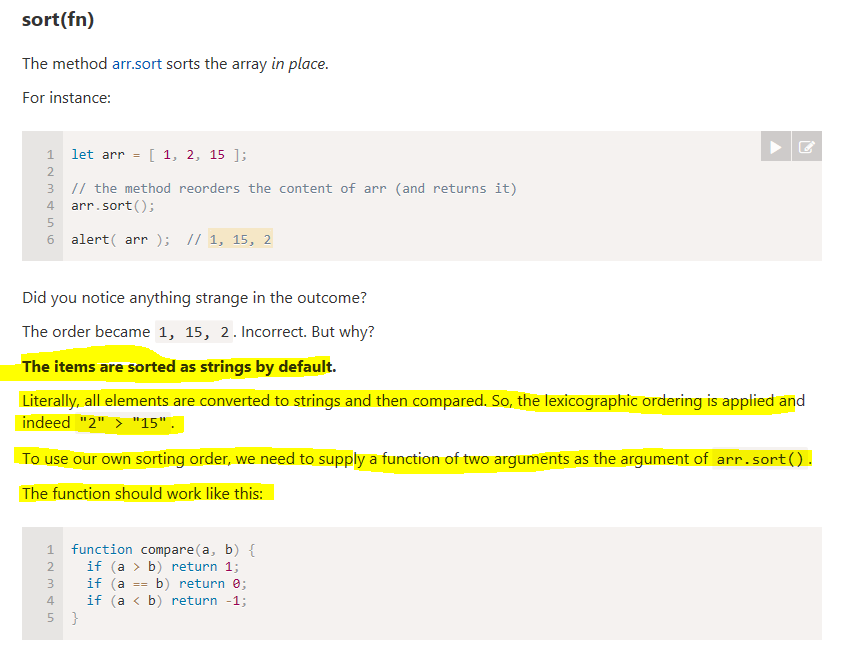
## **[Transform an array](https://javascript.info/array-methods" \l "transform-an-array)**

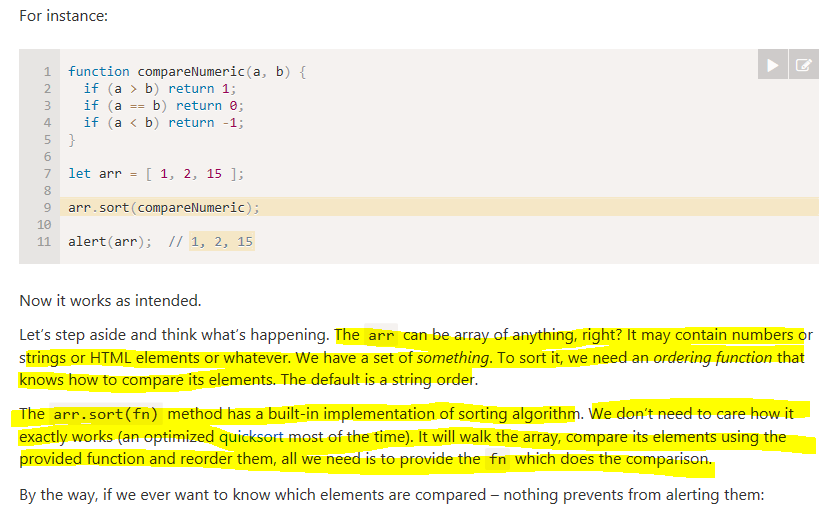
This section is about the methods transforming or reordering the array.

### **map**



### **Sort(fn)**



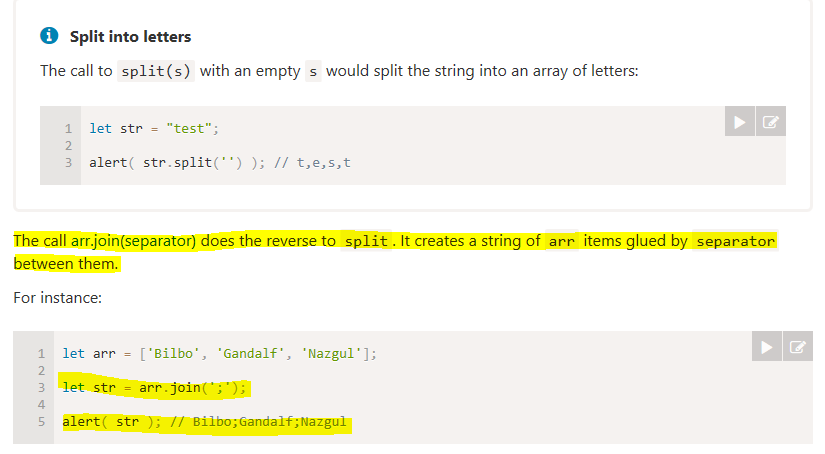


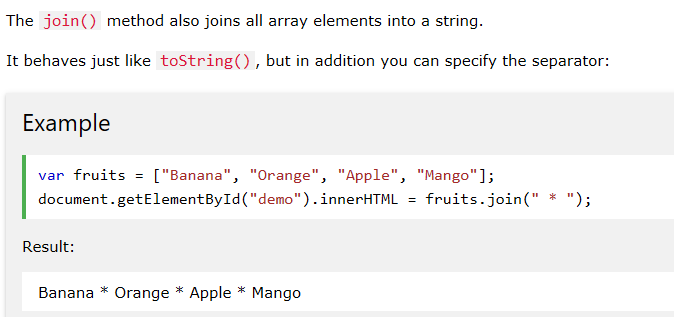


### **reverse**

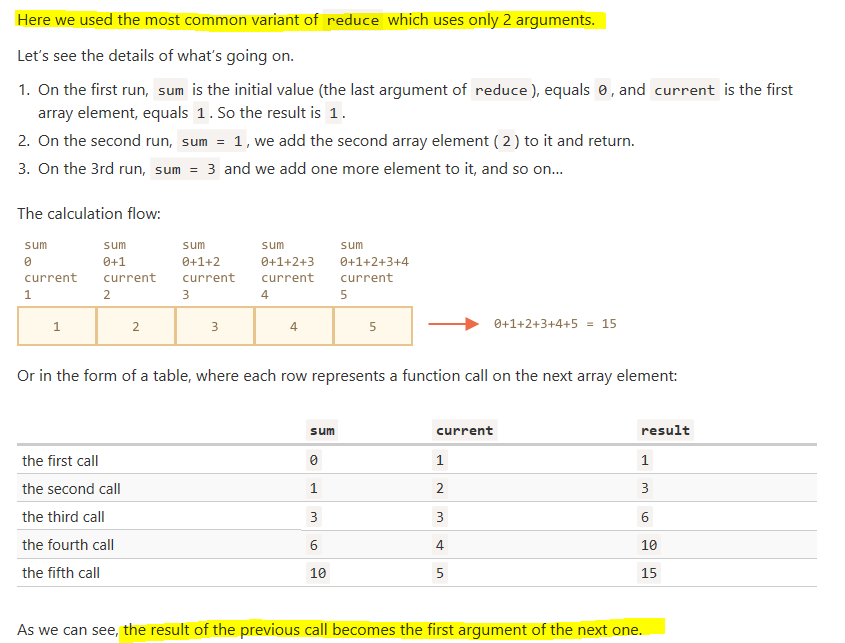
### **split and join**

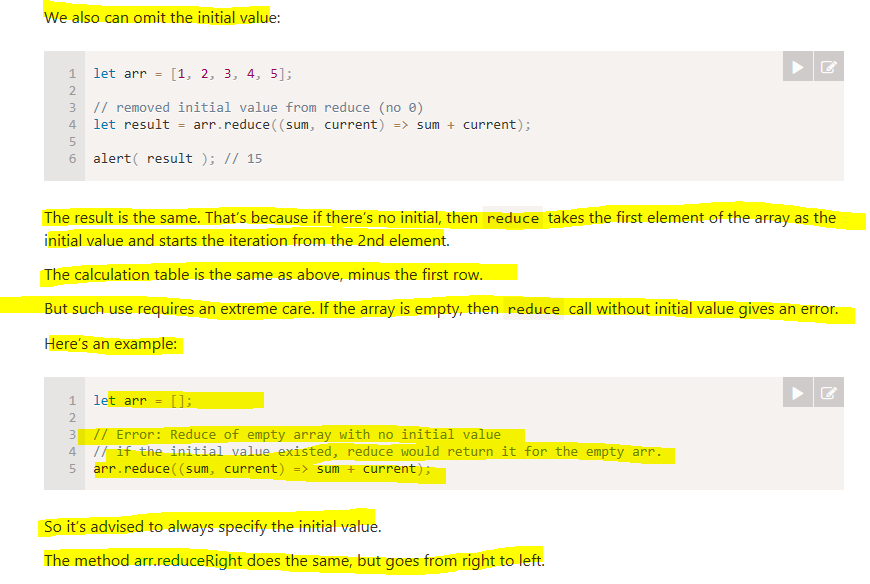


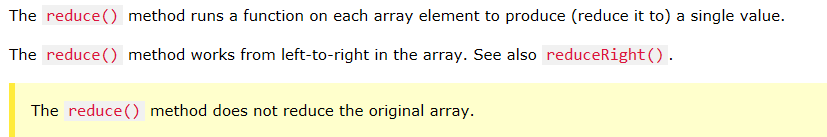




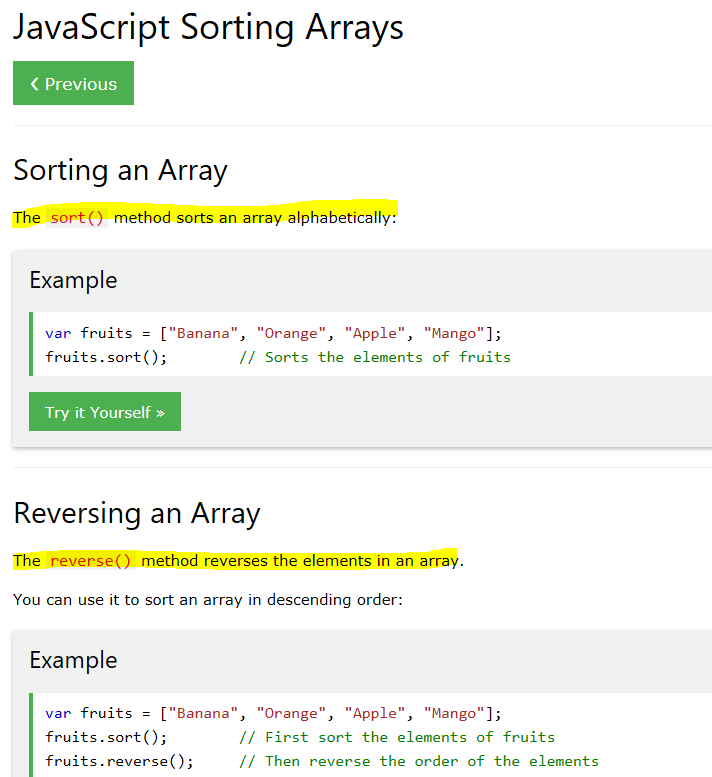
### **reduce/reduceRight**

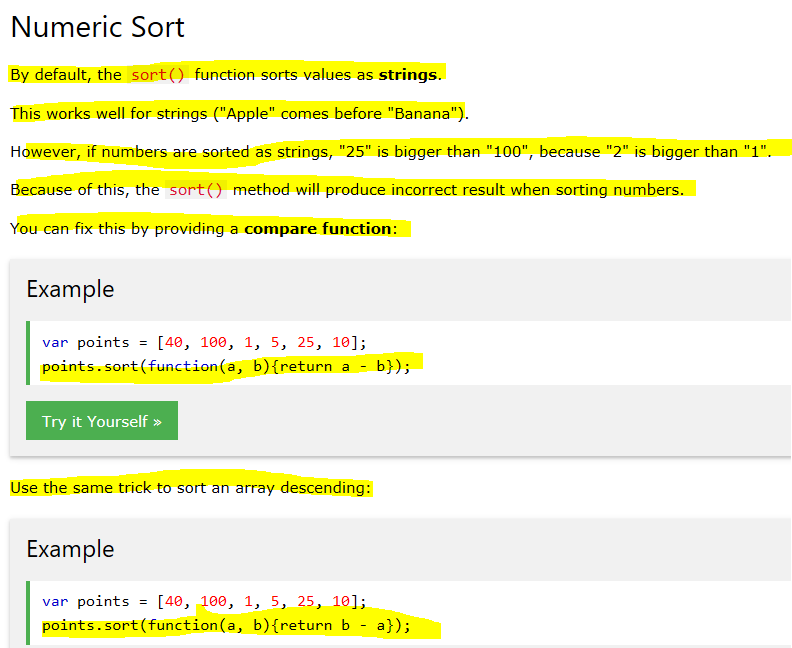


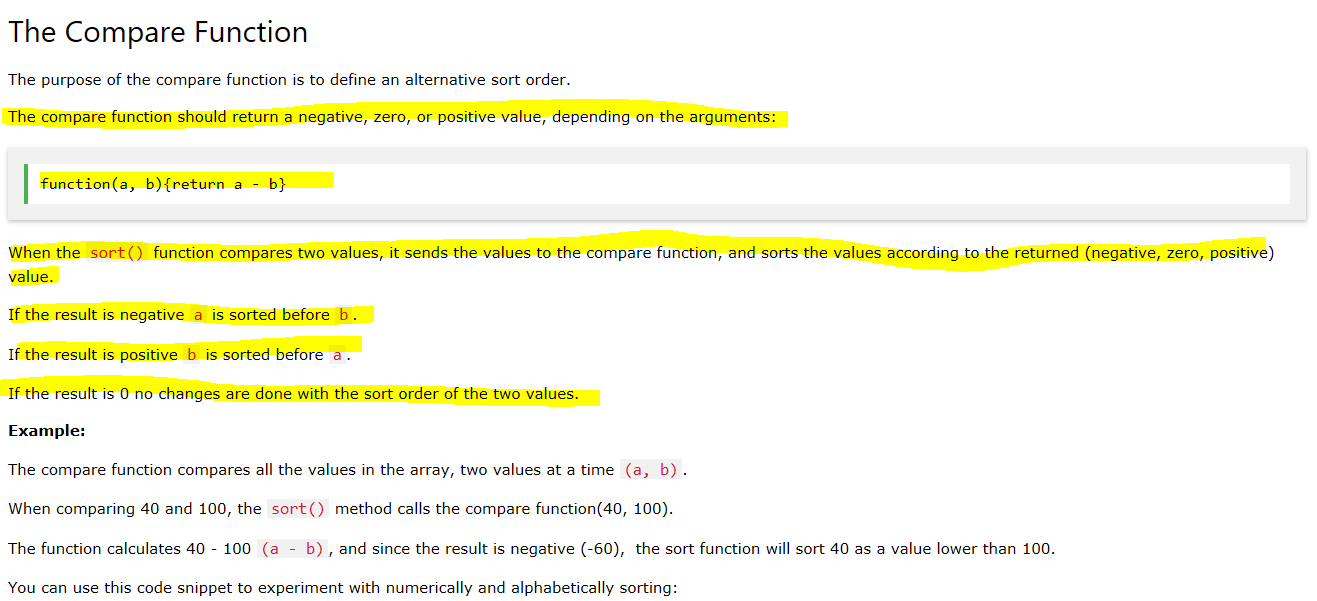


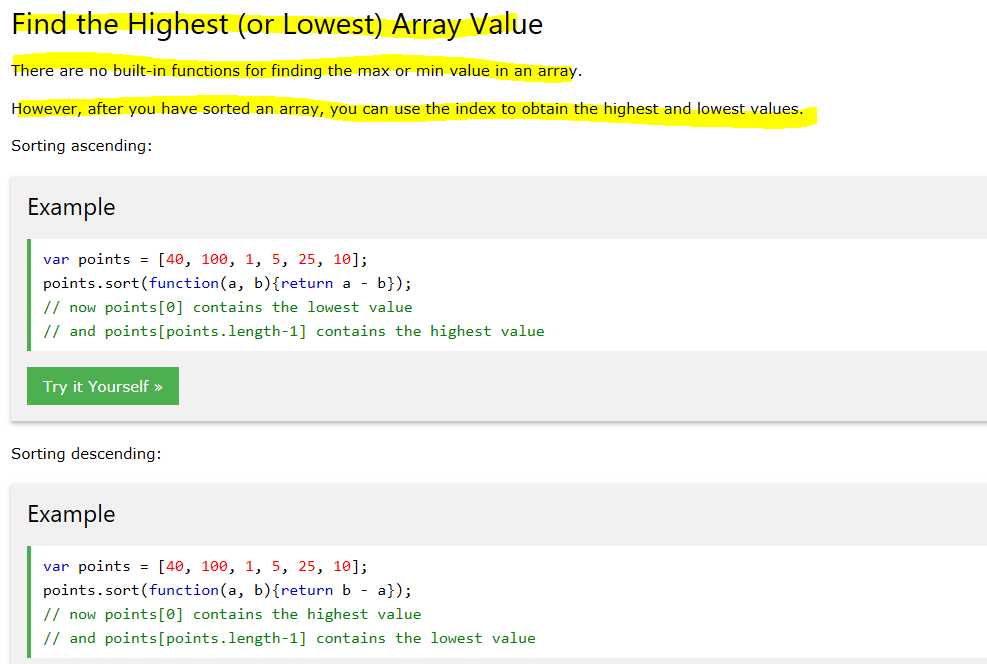


## **Array Sorting related**

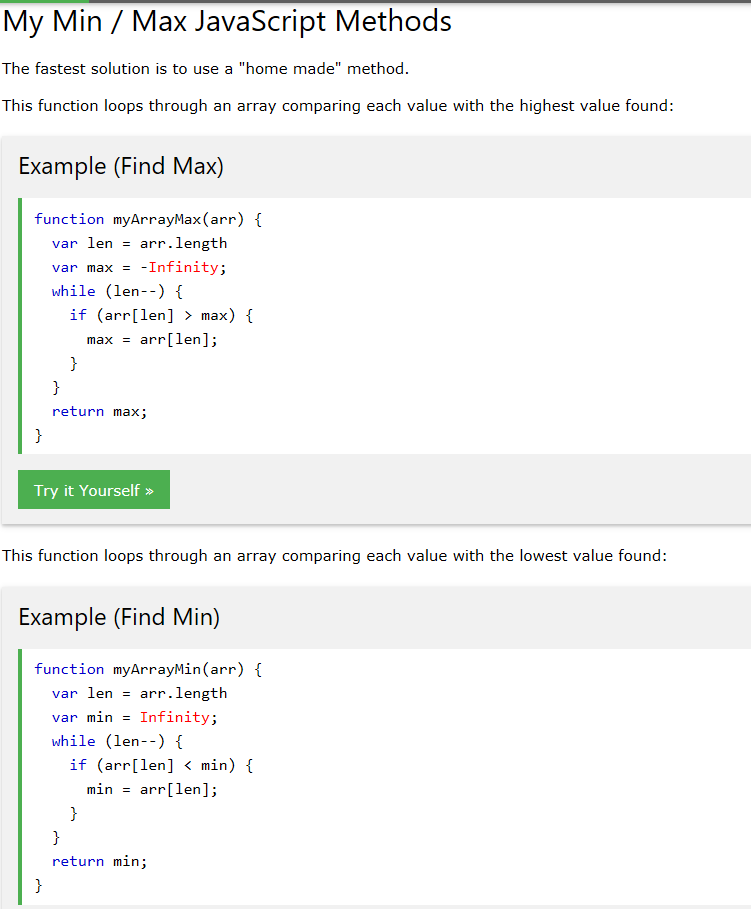




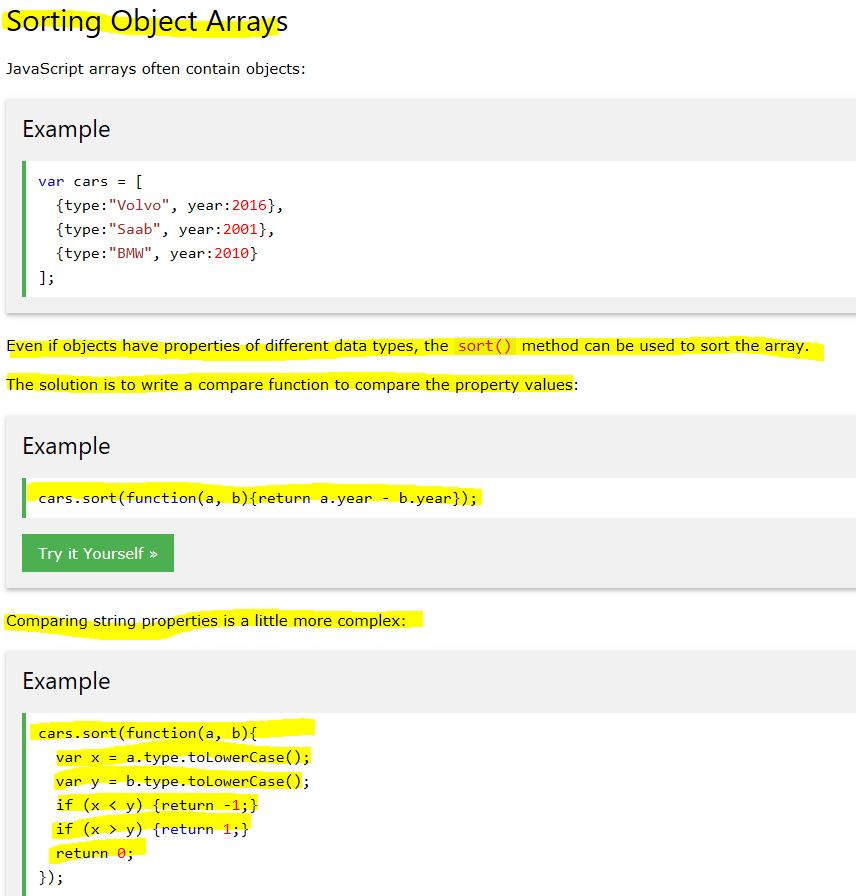




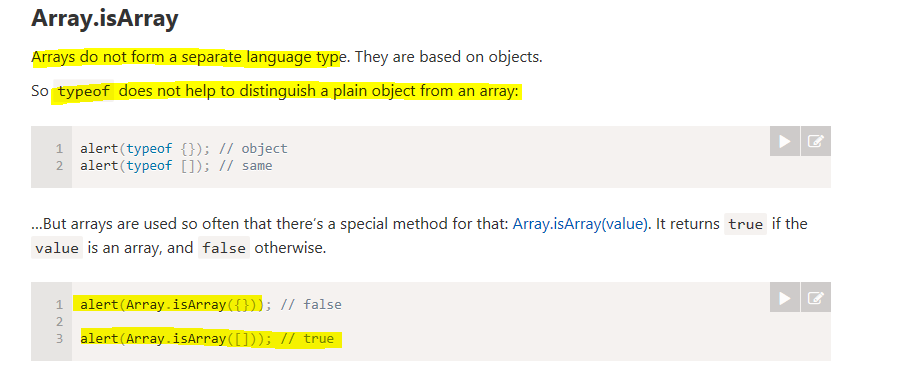




### **Sorting Objects Array**



## **Array.isArray()**

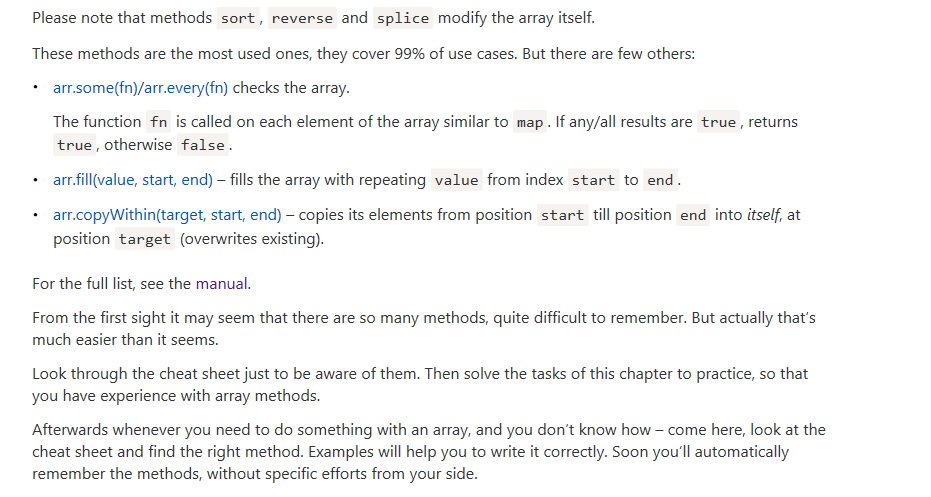


## **thisArg**



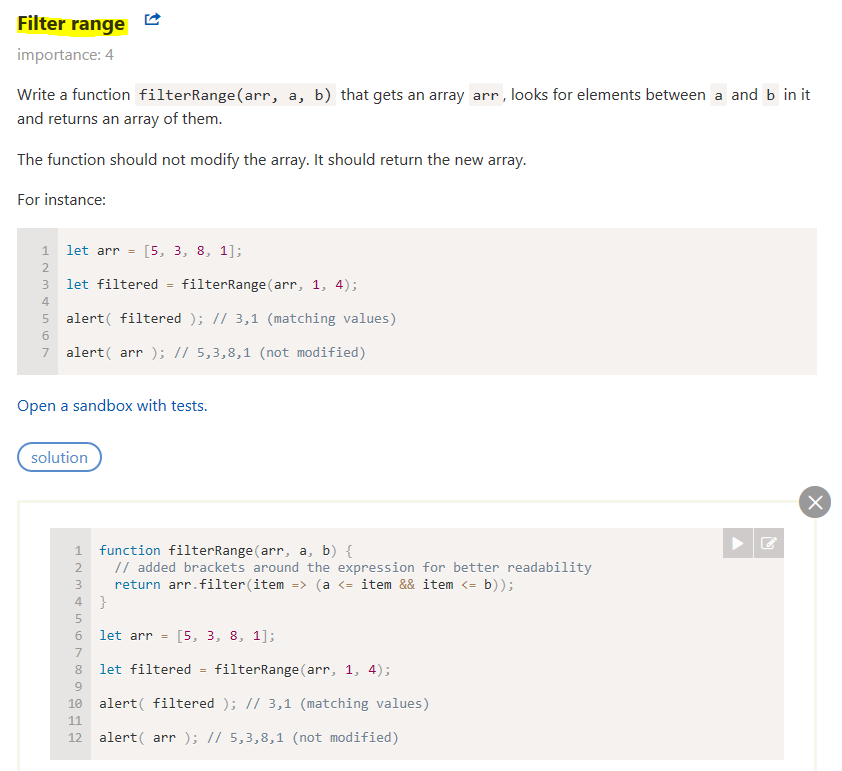
## **Summary**

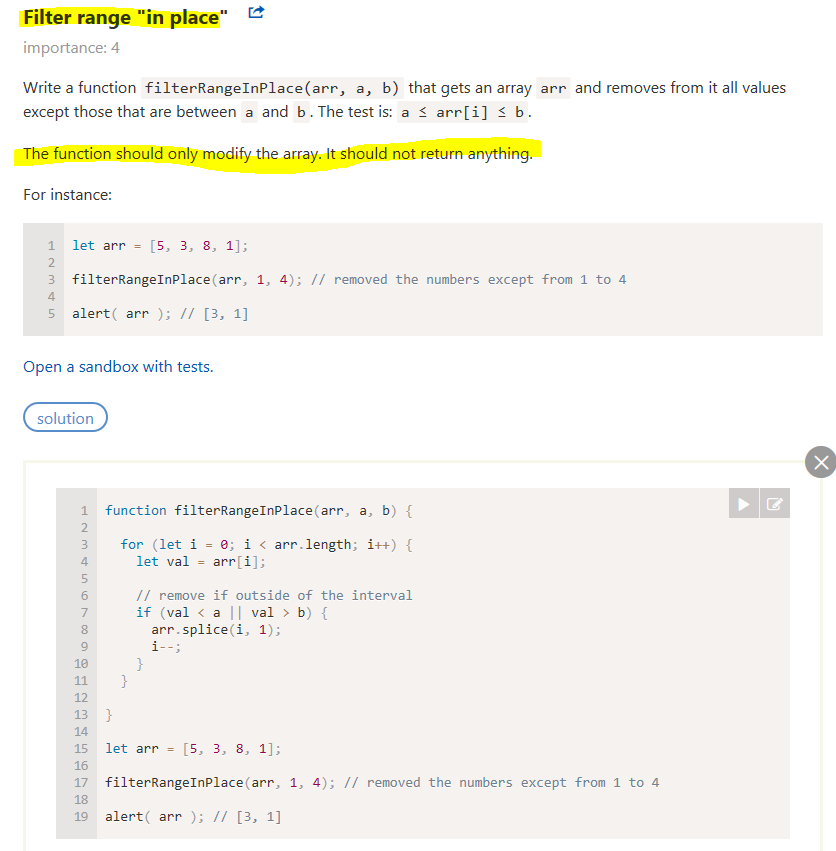


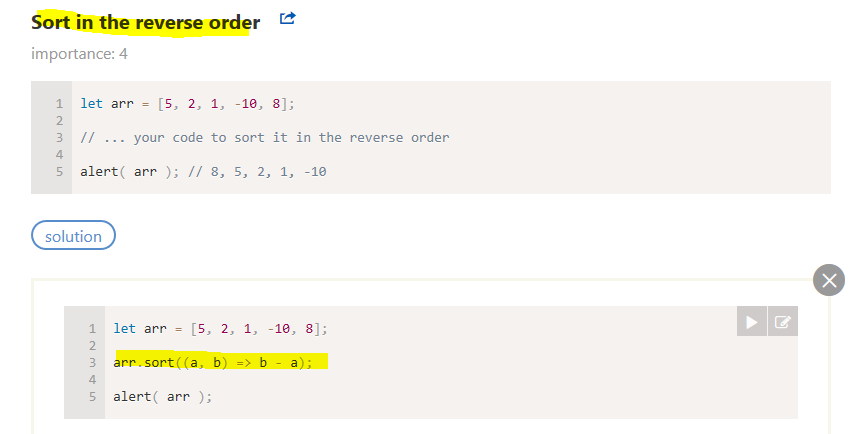


## **Important - Coding Problem**

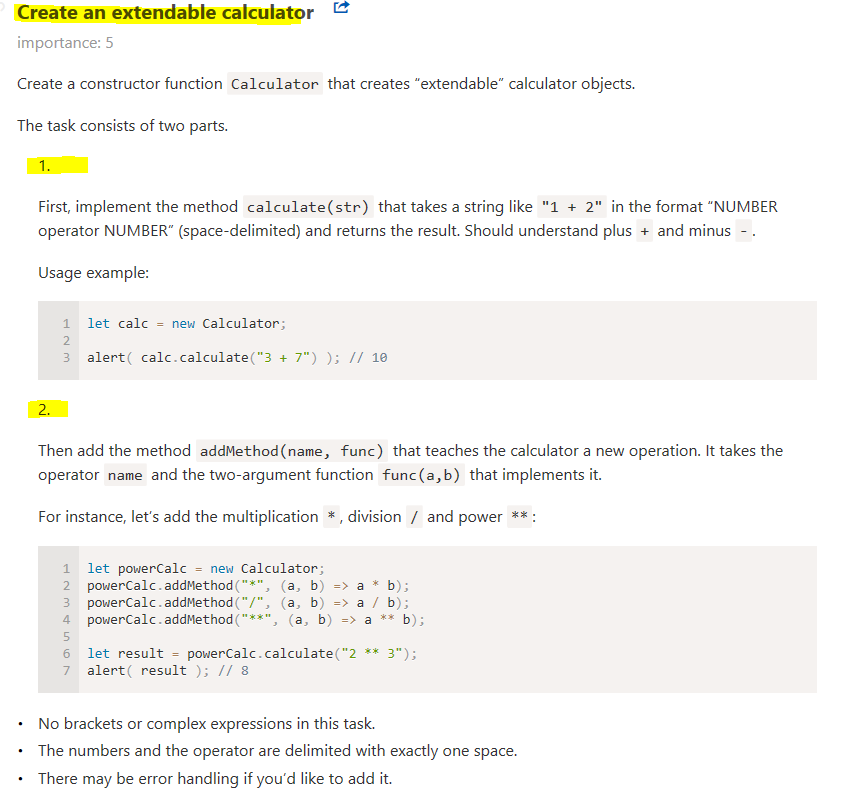


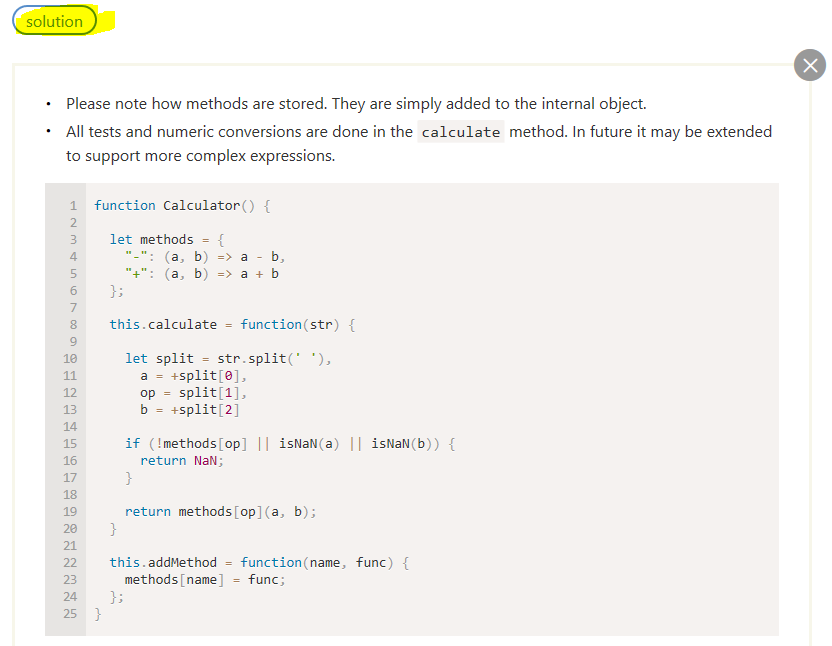




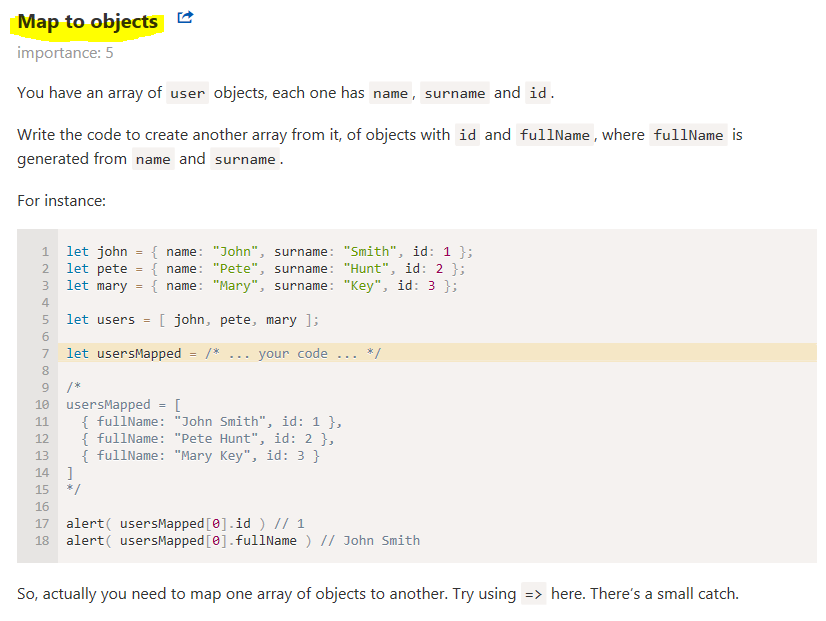




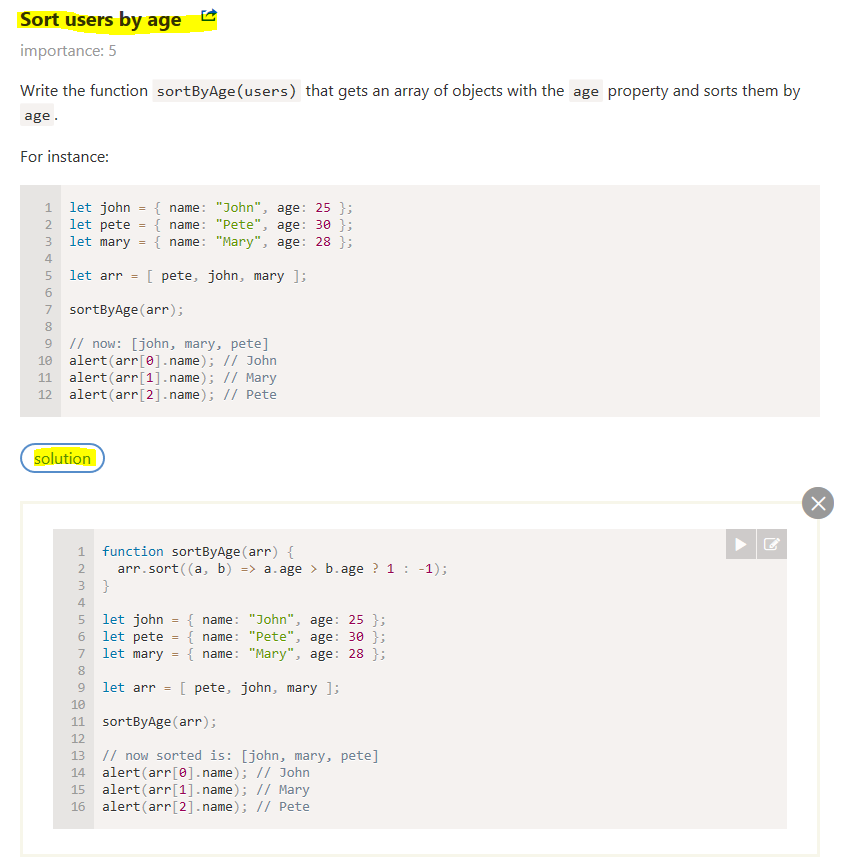
**Good Problem**

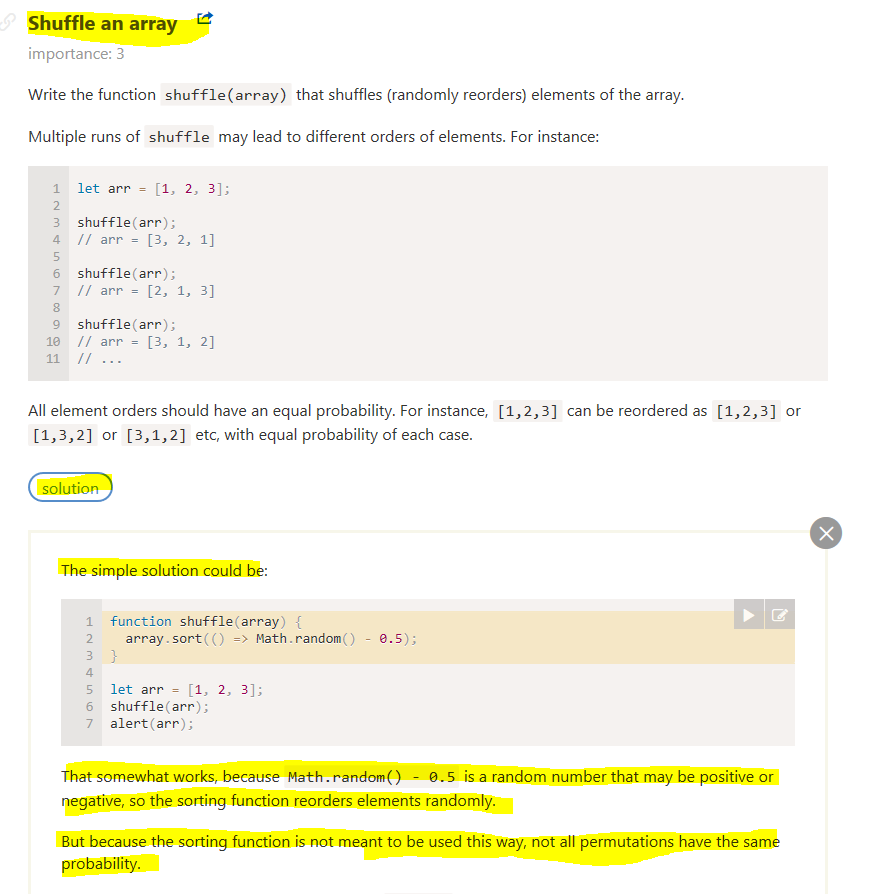


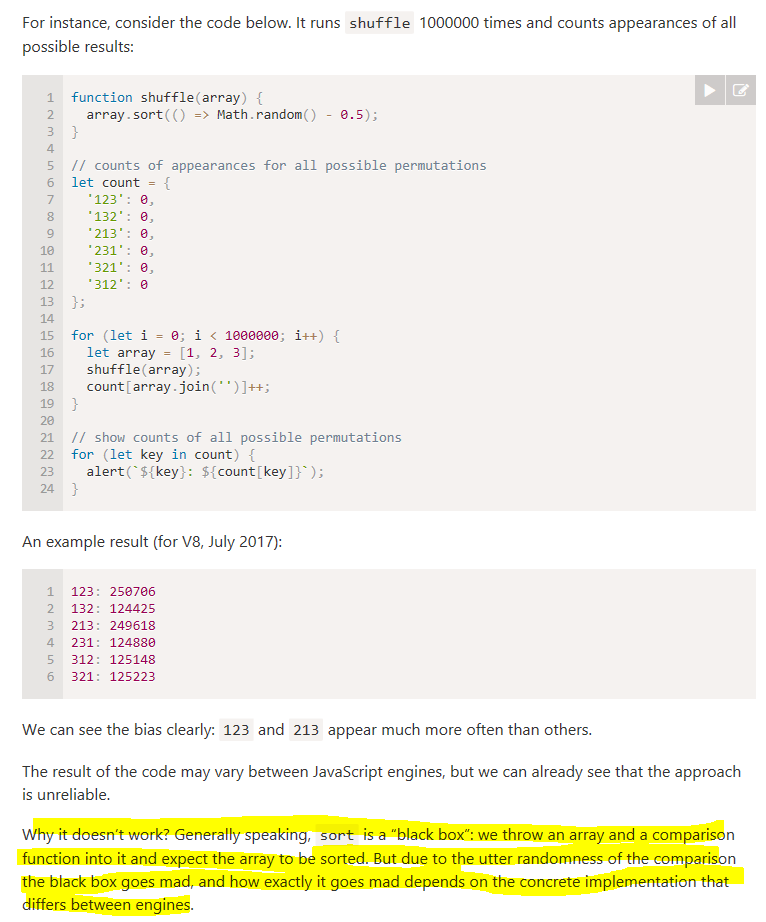


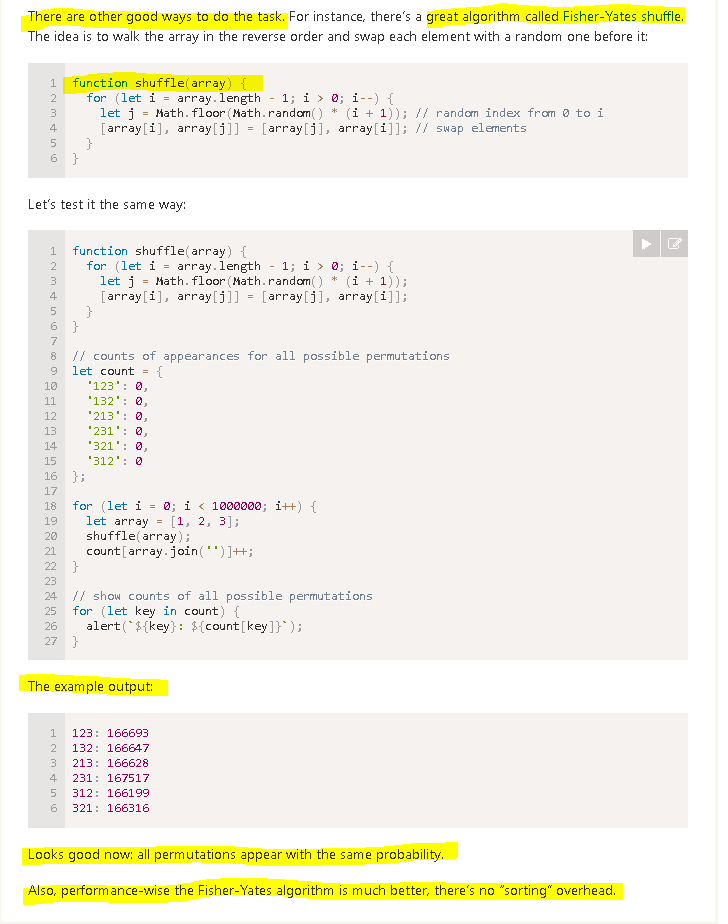
**Good Problem**



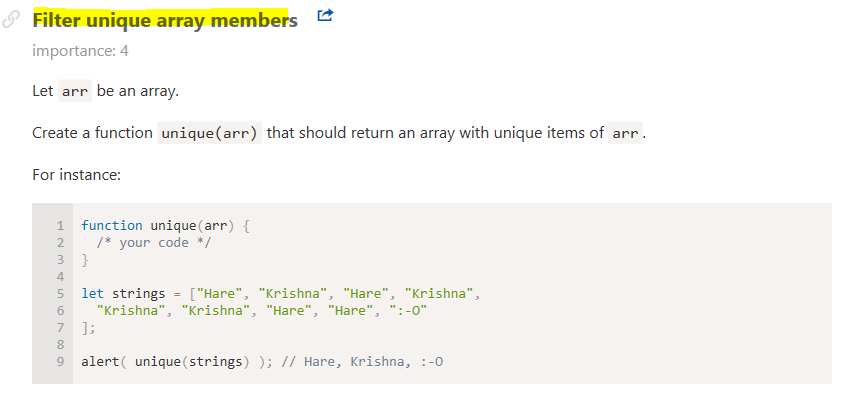










**Good Problem**

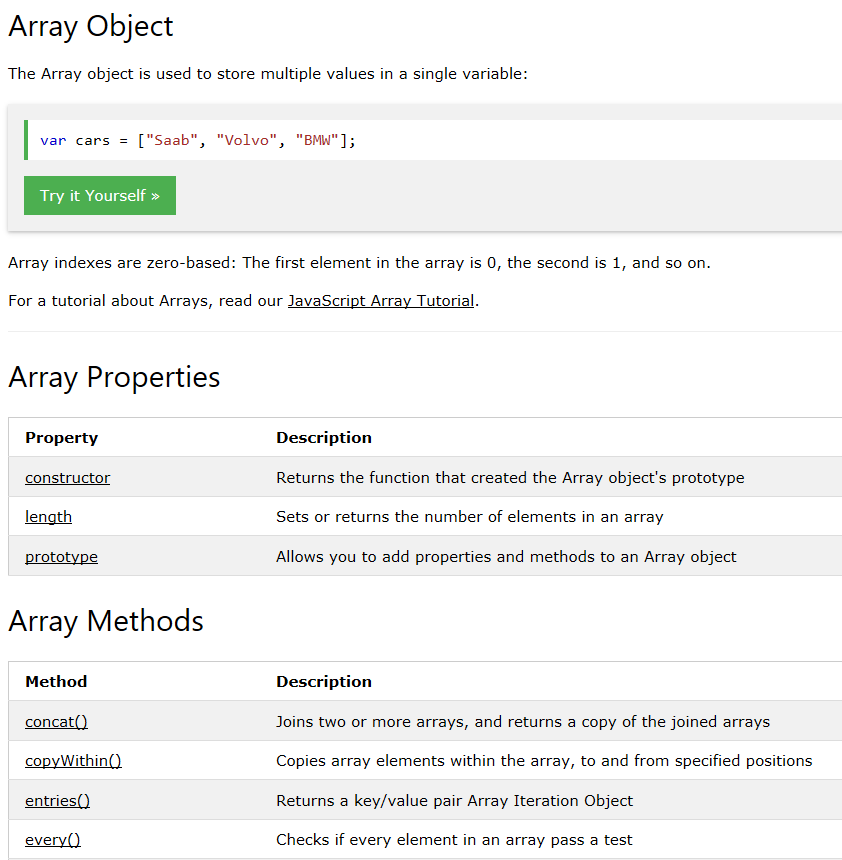


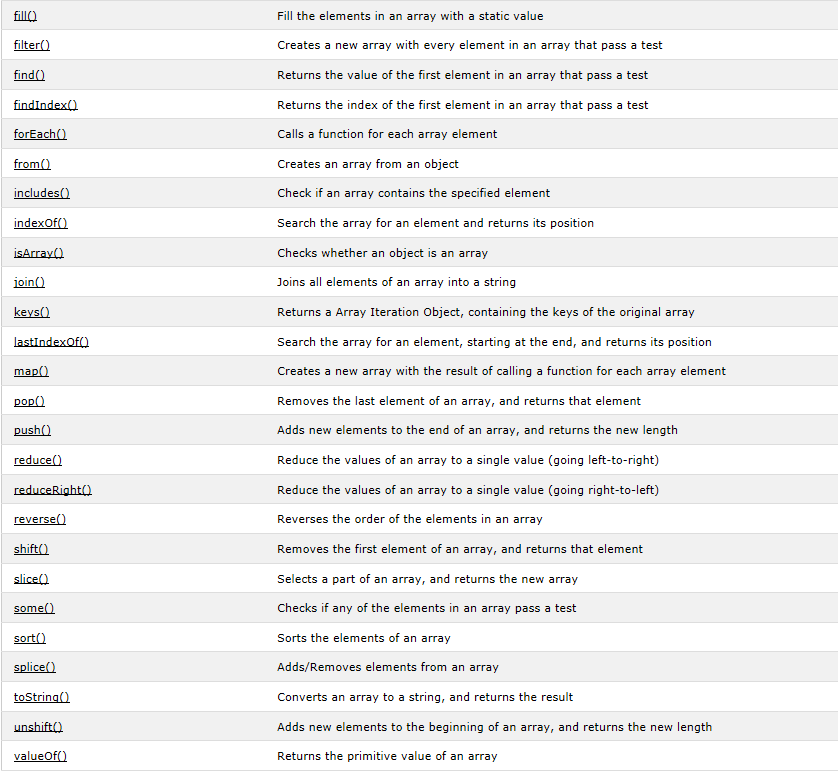
## **References**

<https://javascript.info/array-methods#transform-an-array>

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array>

<https://www.w3schools.com/jsref/jsref_obj_array.asp>

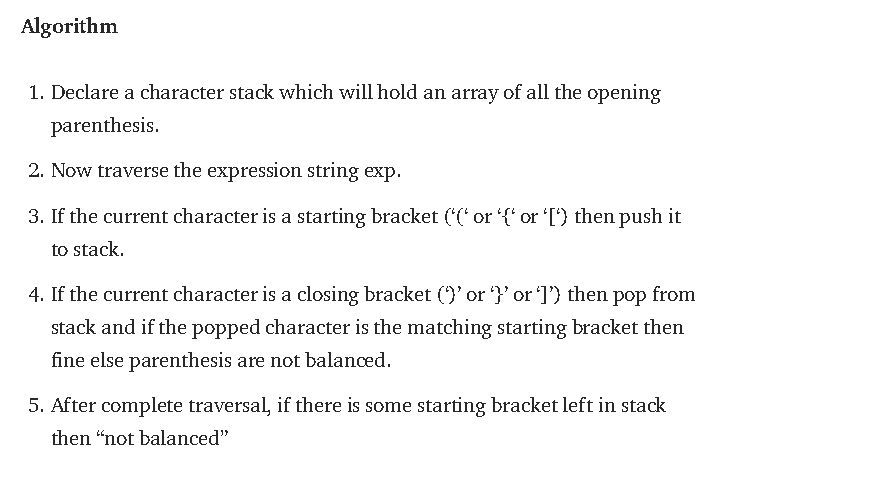
**Array reference in short (Cheat Sheet)**



# **Array Coding Problem**

Parenthesis Matching Problem in JavaScript

<https://medium.com/@paulrohan/parenthesis-matching-problem-in-javascript-the-hacking-school-hyd-7d7708278911>

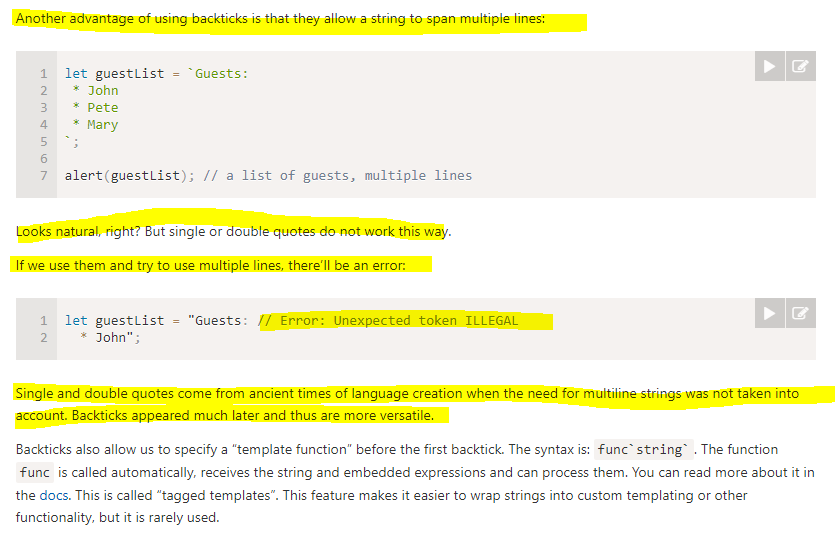


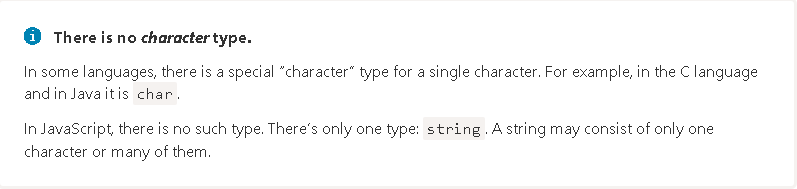


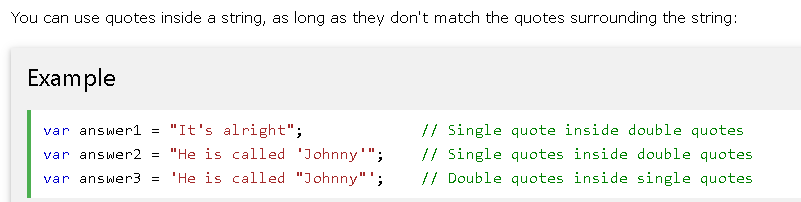
# **String**

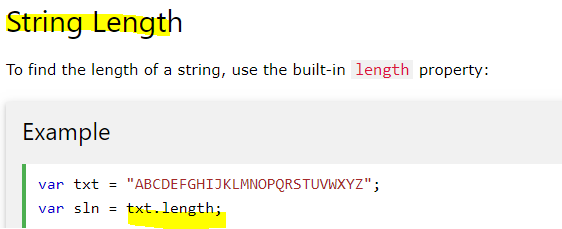




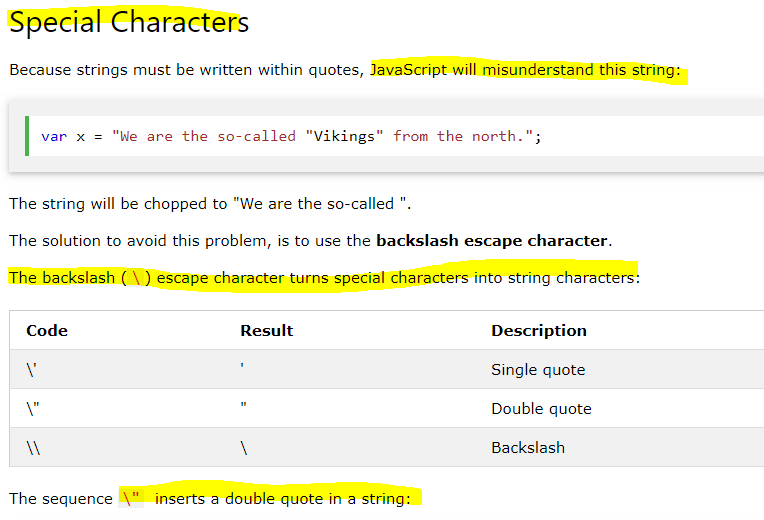


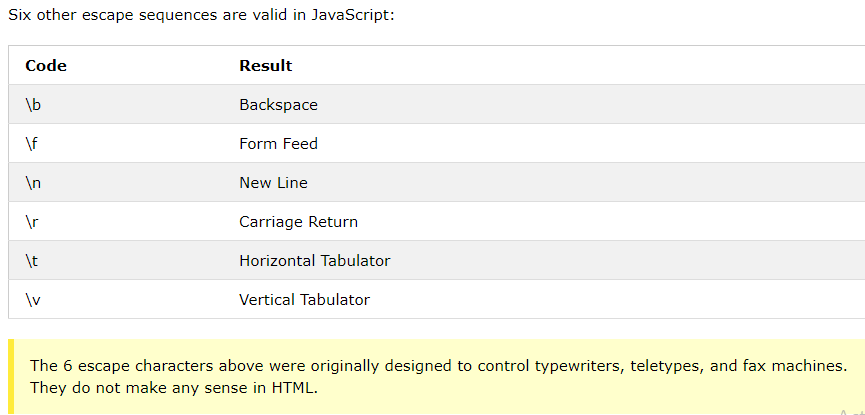


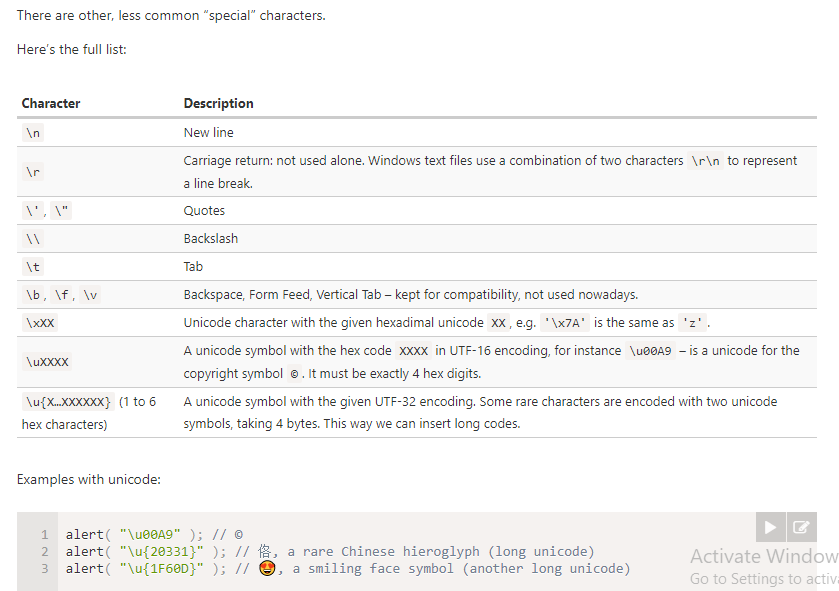


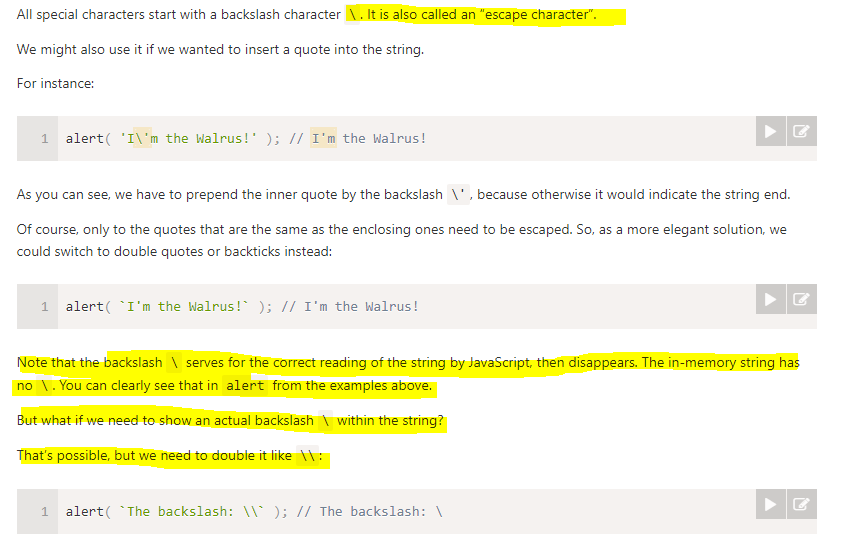


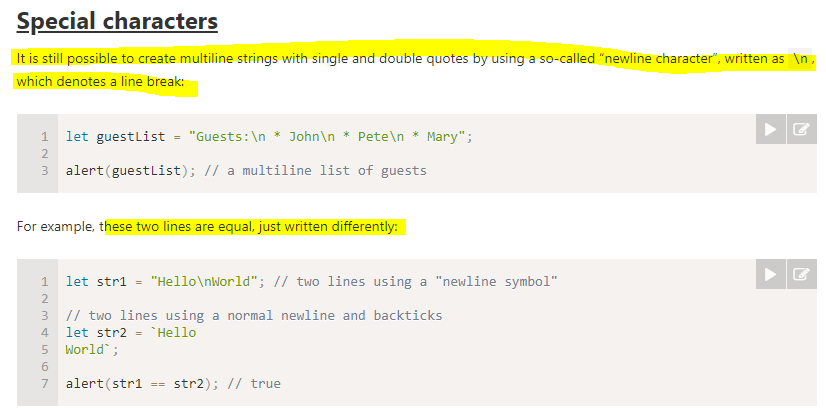


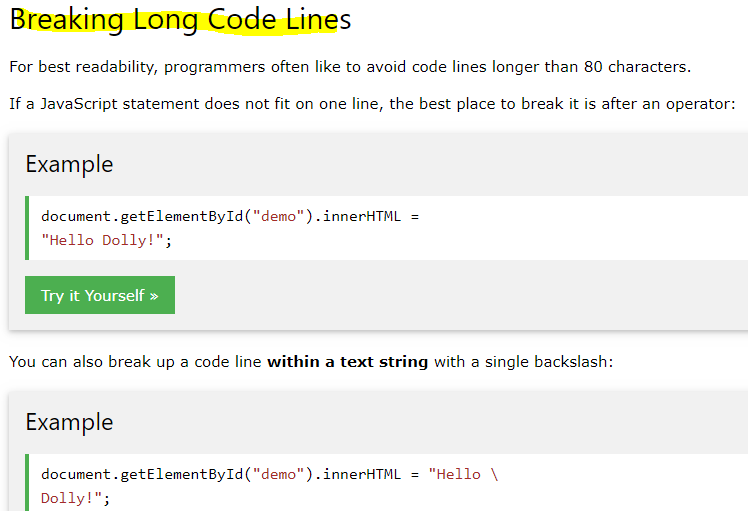


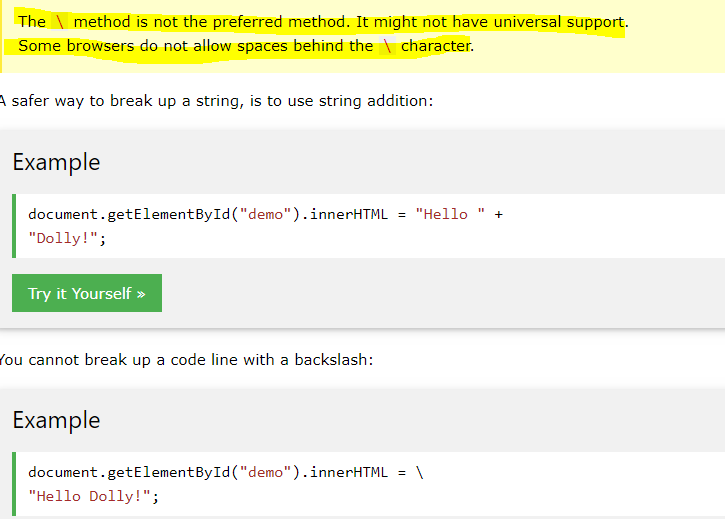


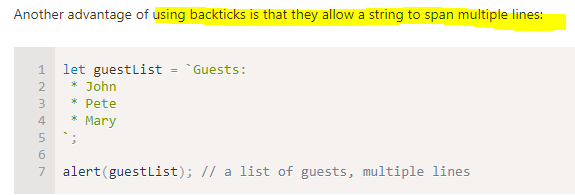


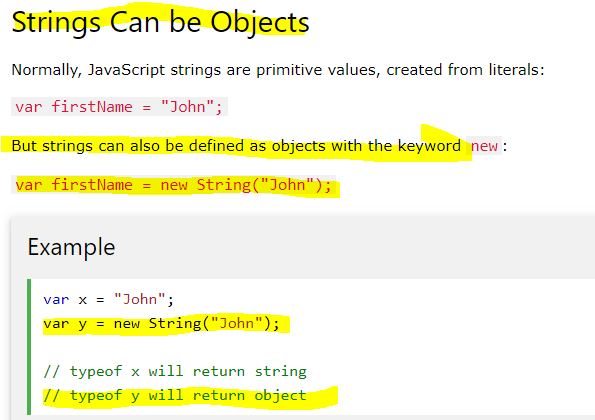


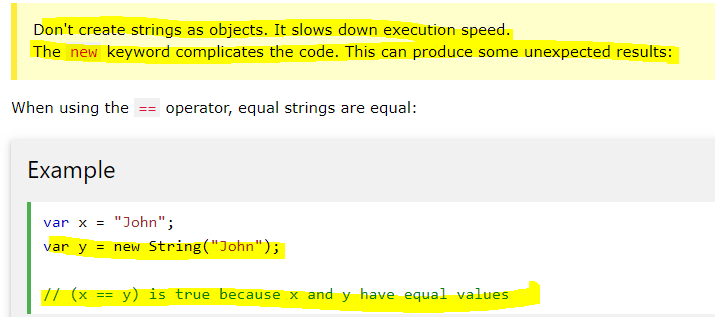


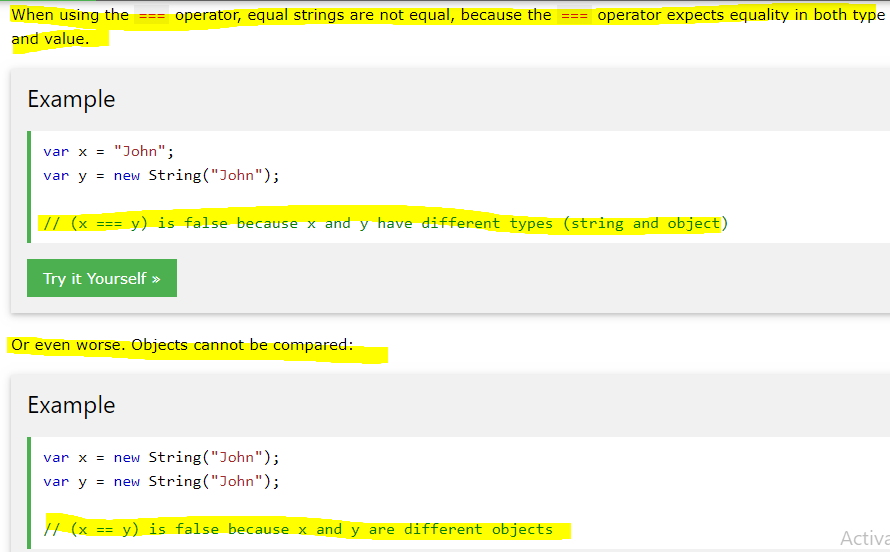


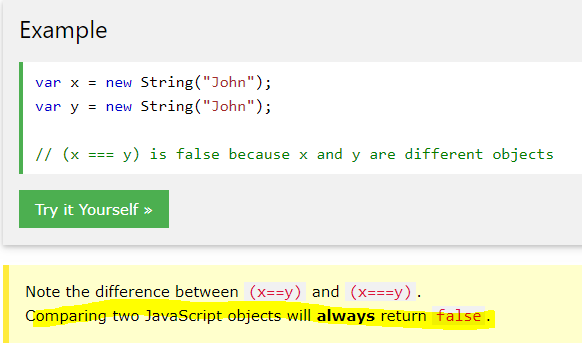


**Good way to breakup string in multi line -** 

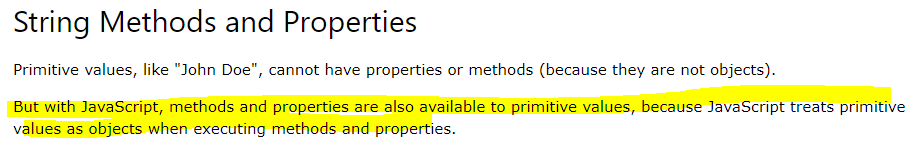


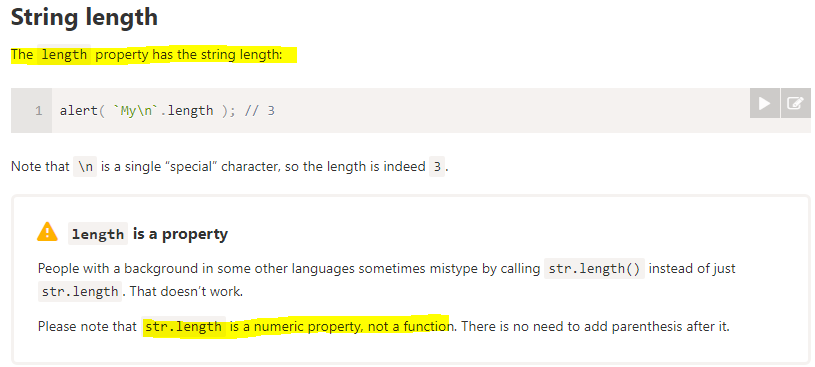




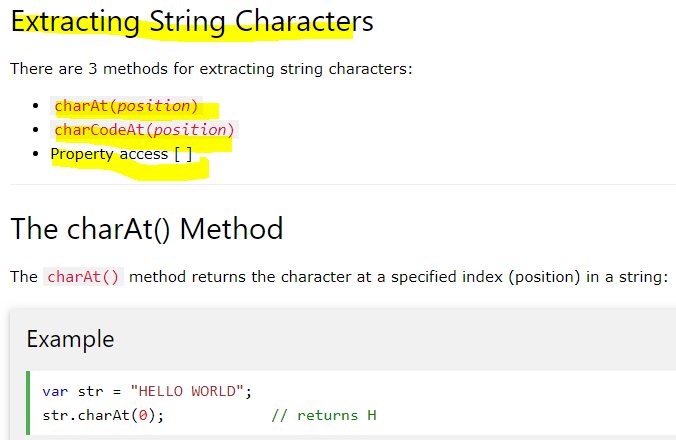


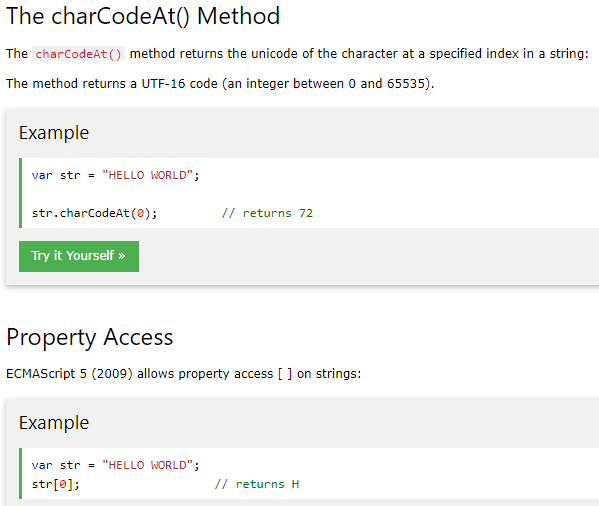
# **String Methods**

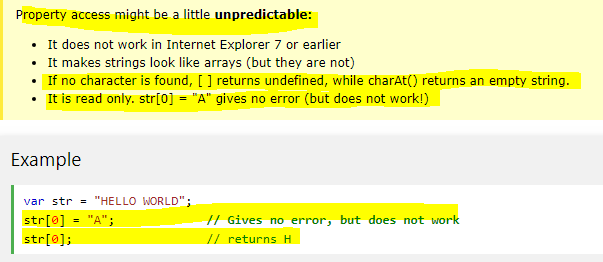


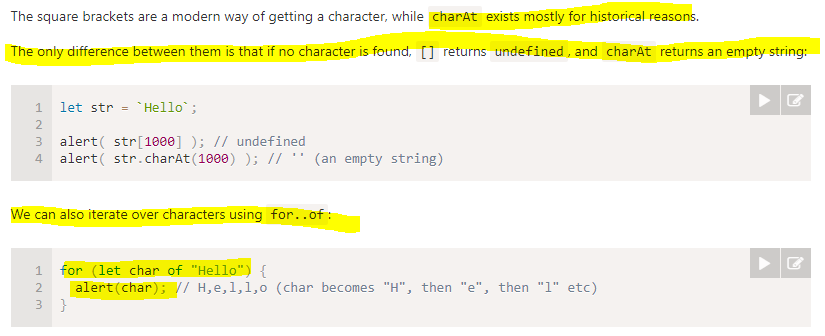


## **Accessing Characters**

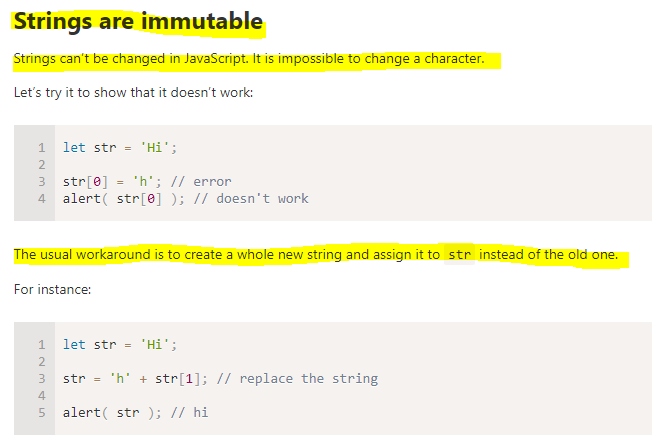




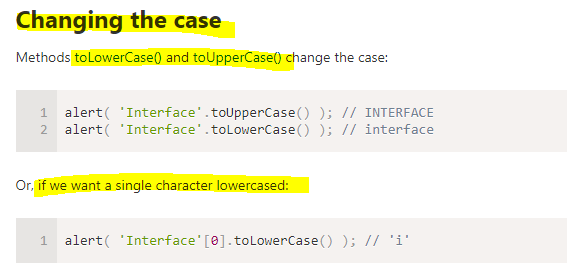




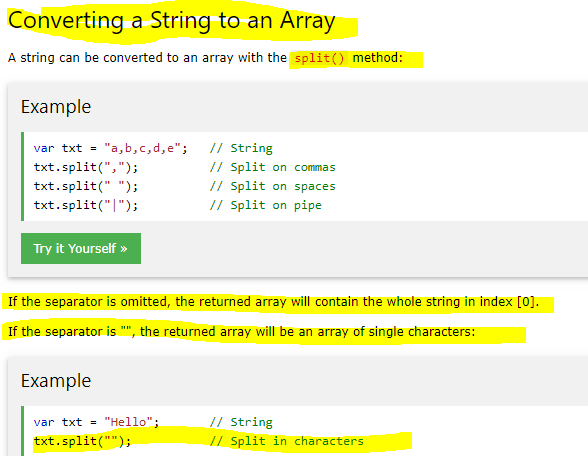
## **Strings are Immutable**

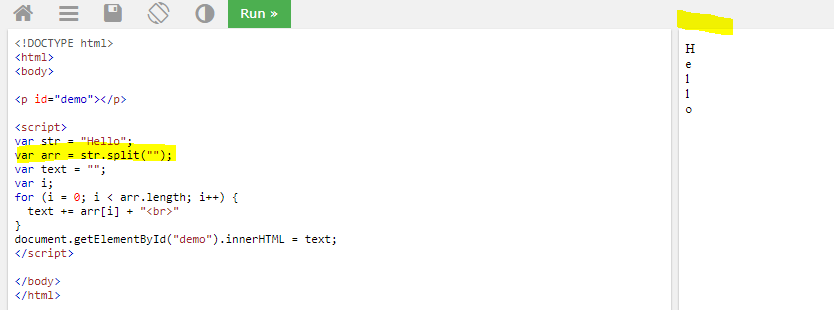


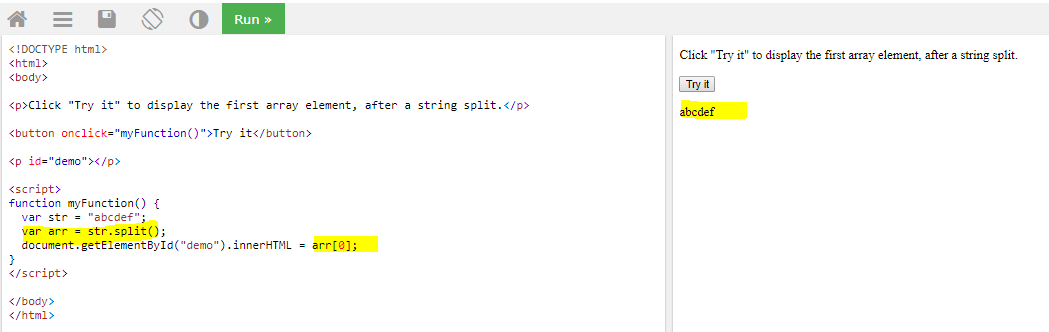
## **Changing the case (Uppercase or Lowercase)**



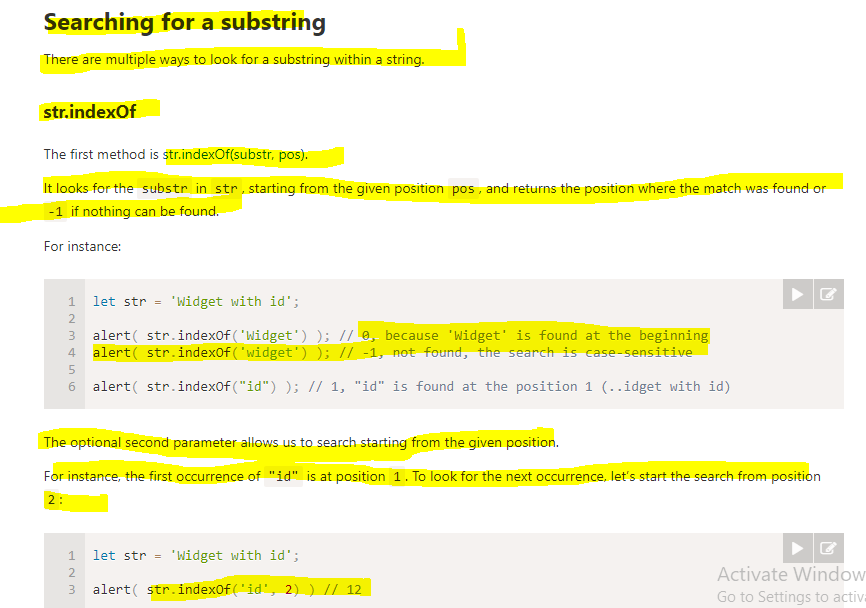
## **Converting String to Array – split method**

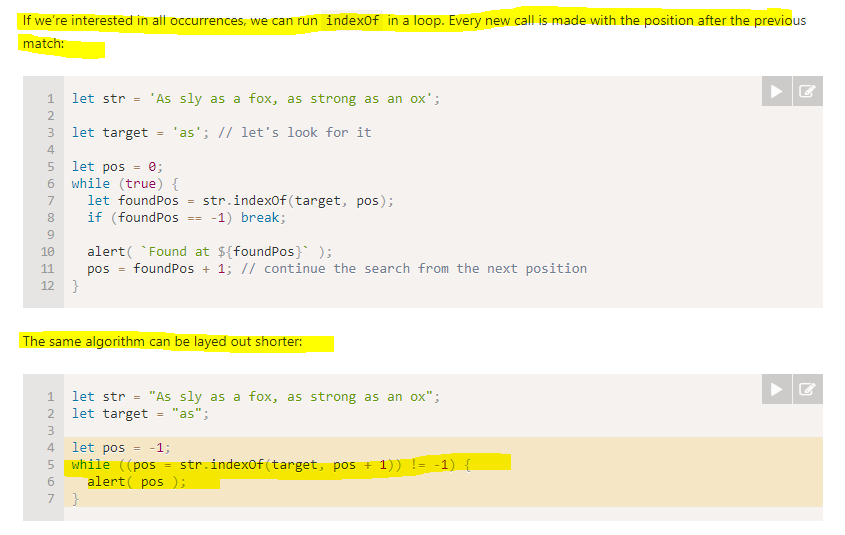




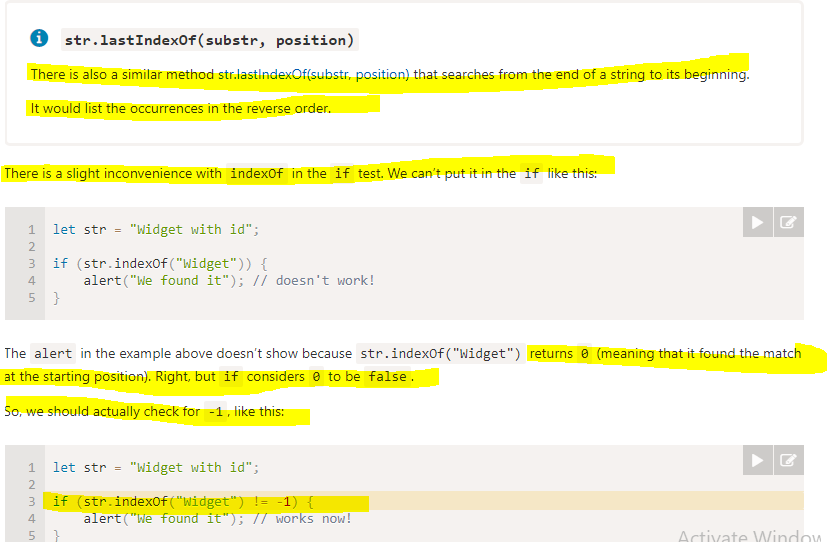


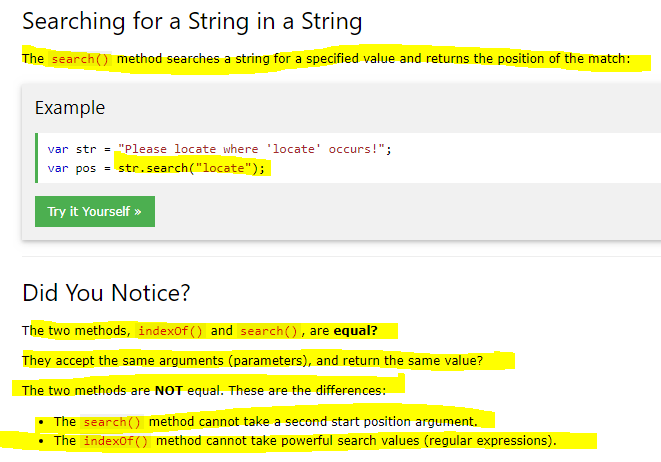
## **Searching for Substring**

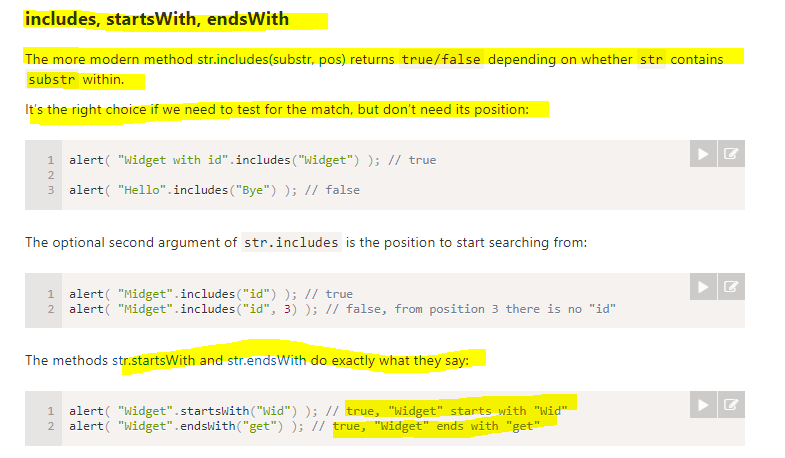


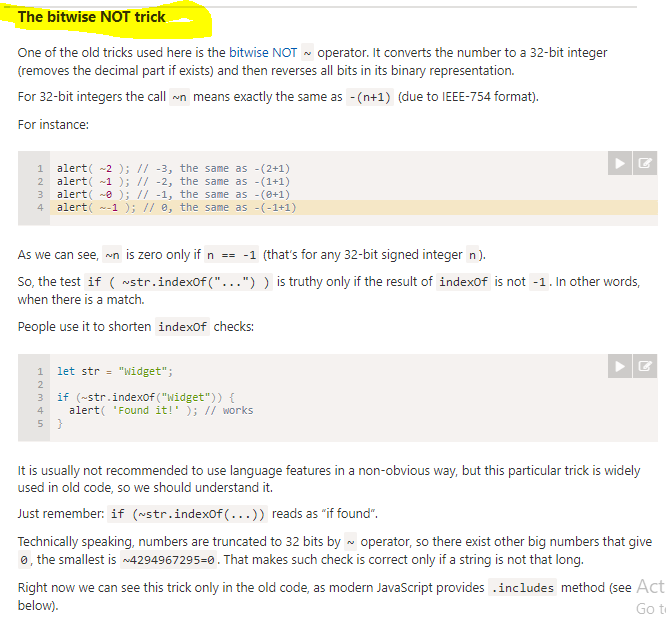




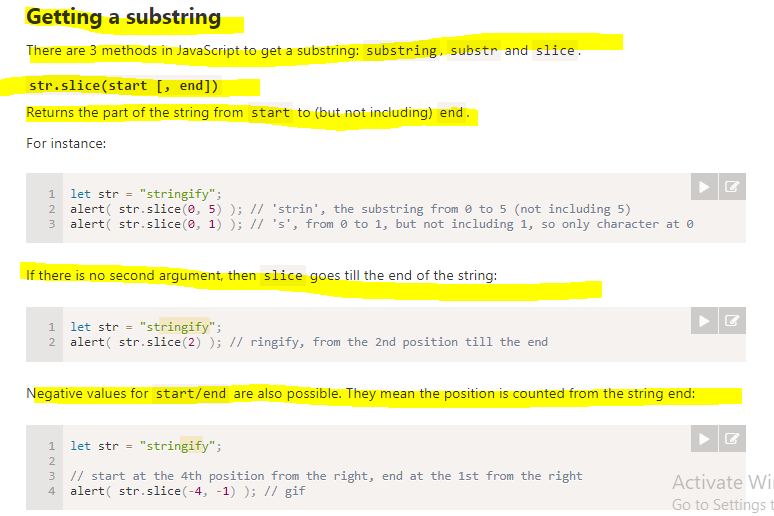








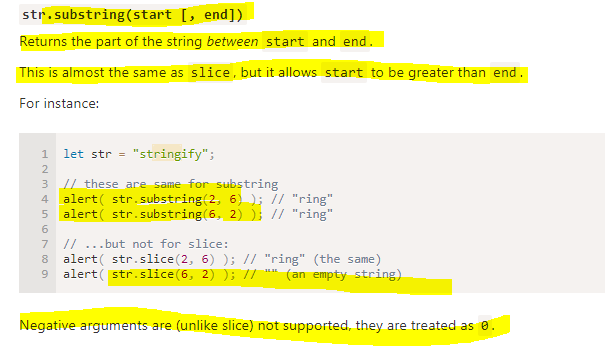
## **Extracting/Getting a substring**

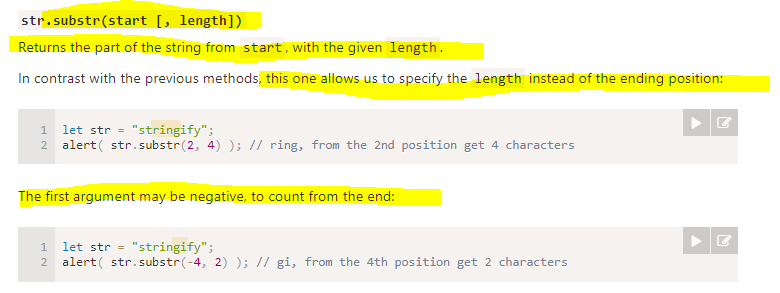


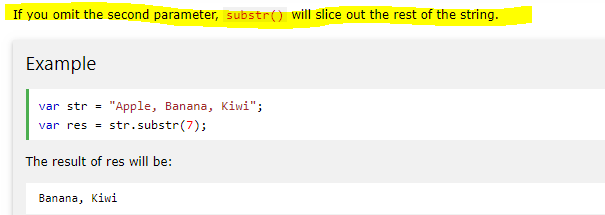


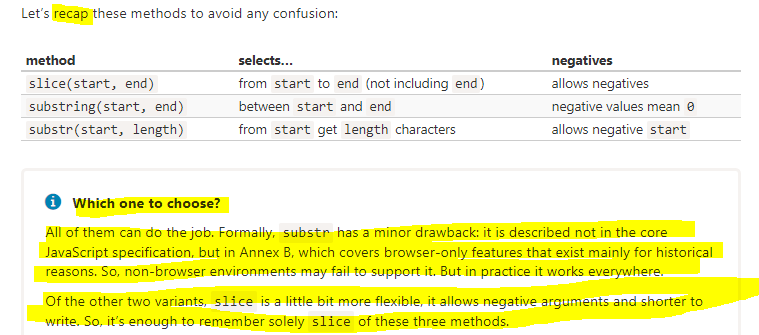




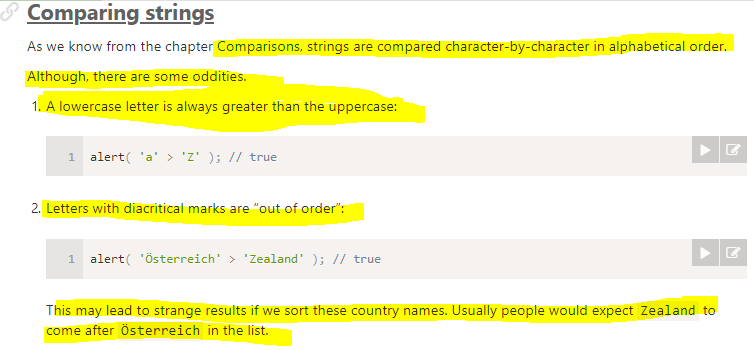


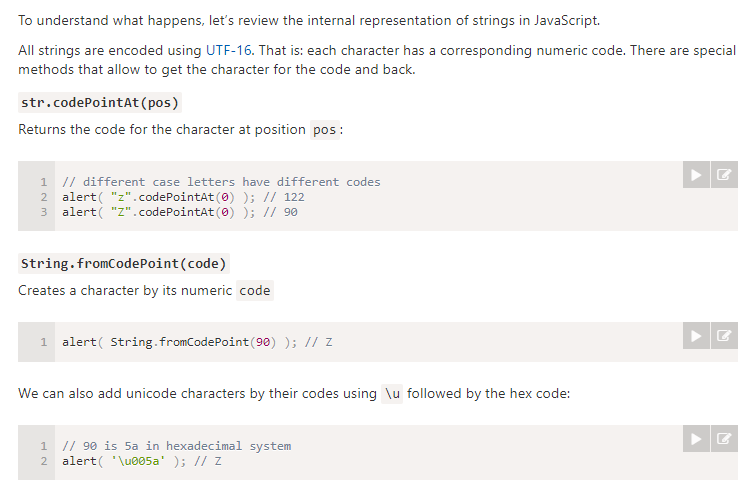


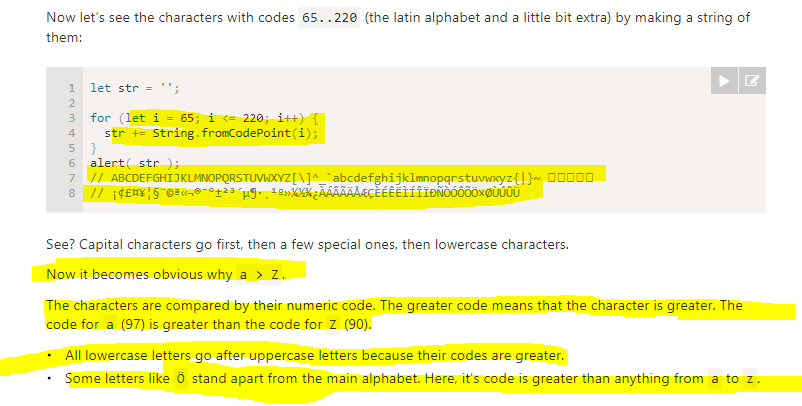


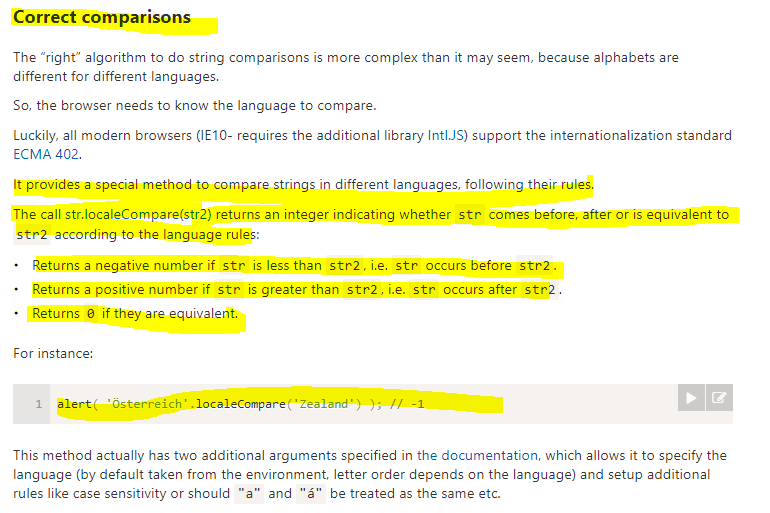


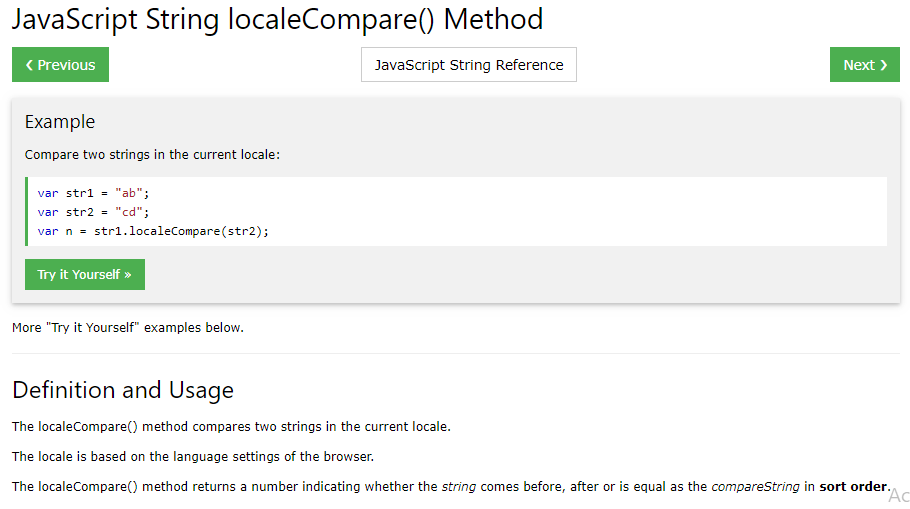
## **Comparing Strings**



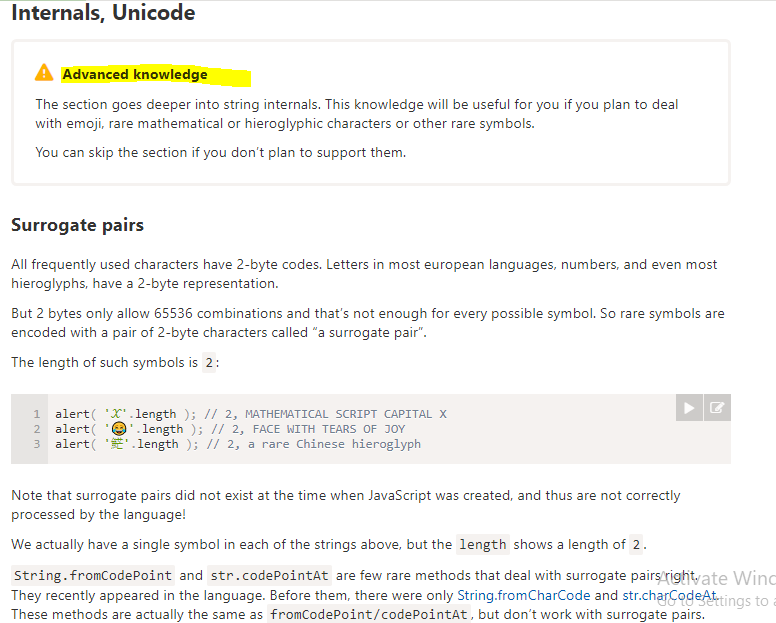


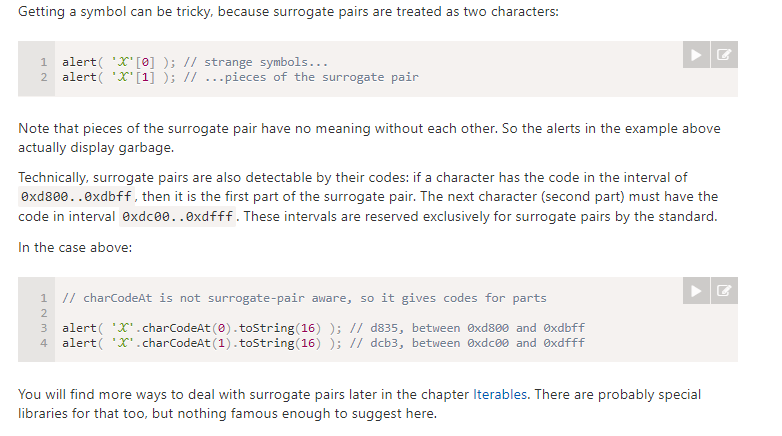


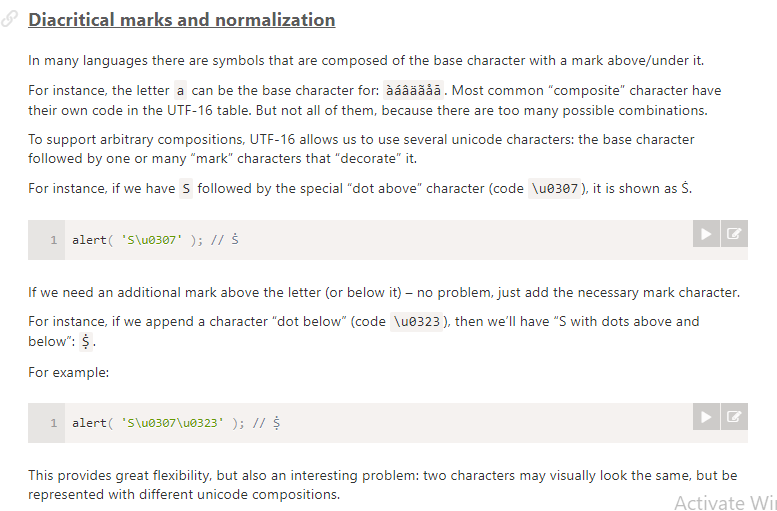


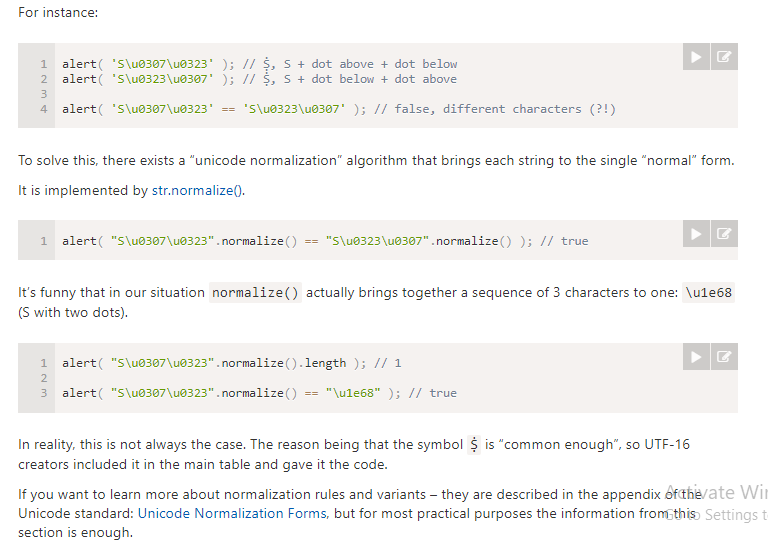


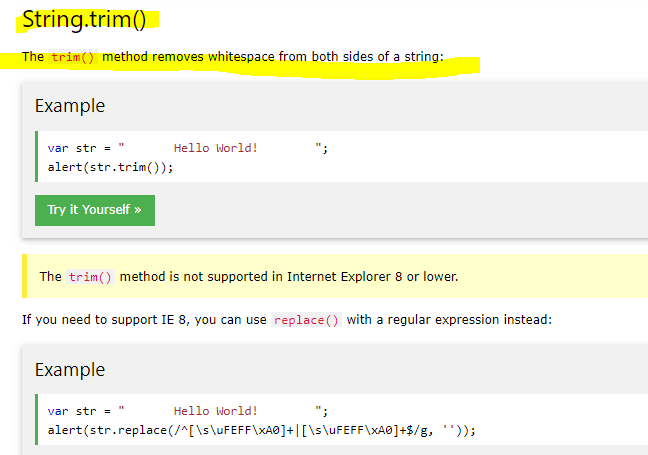
## **Internals , Unicode**

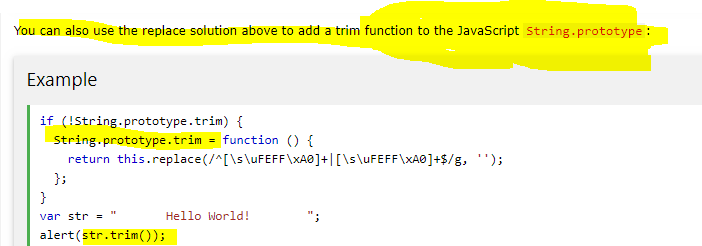


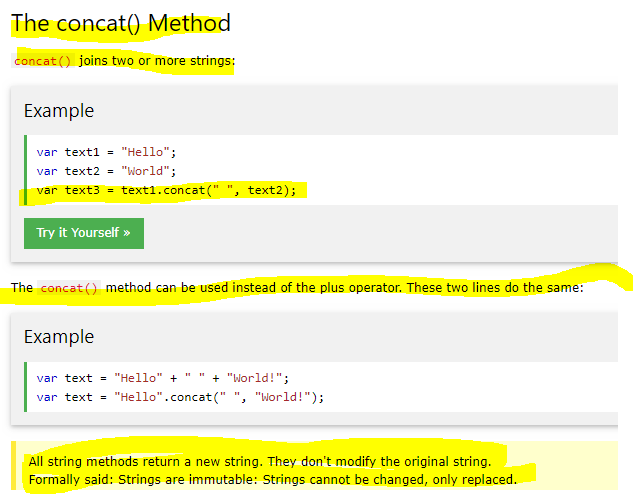


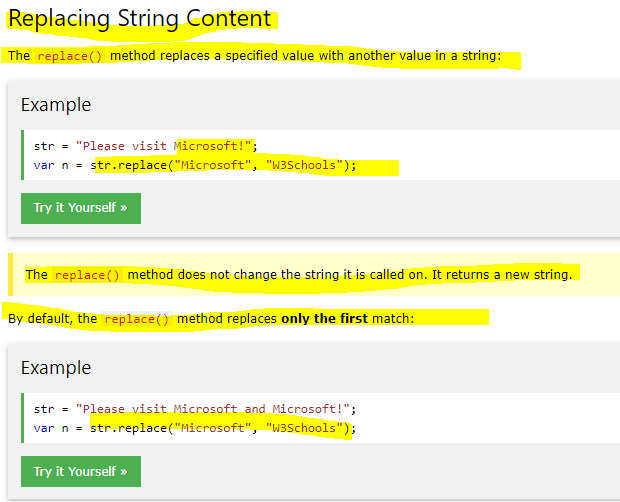


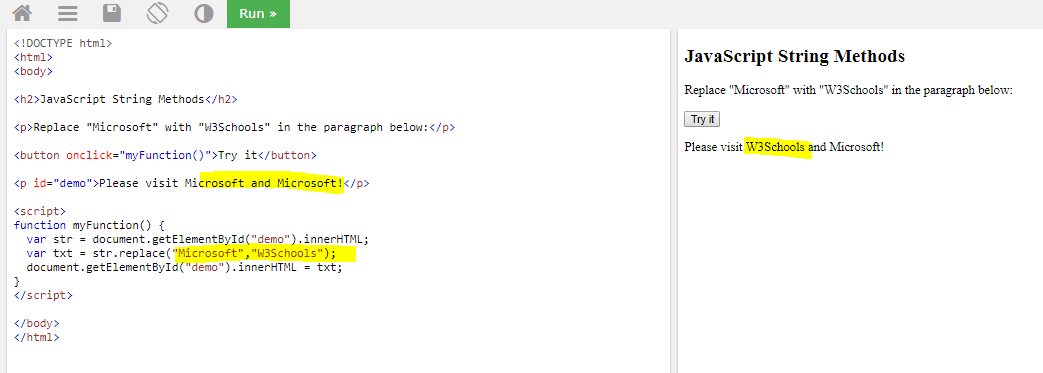


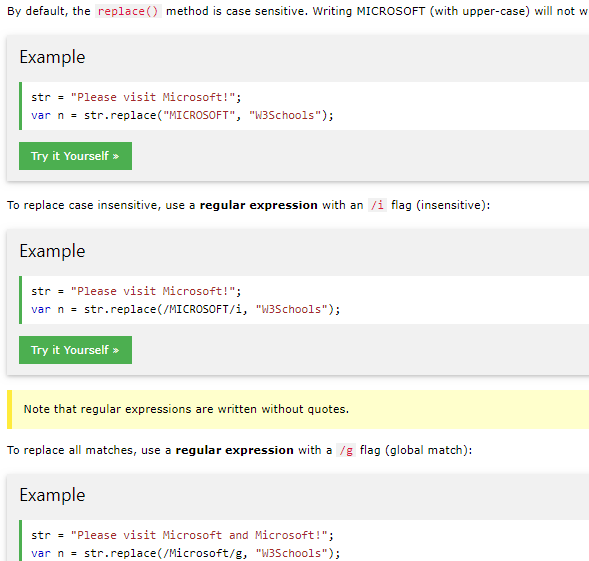


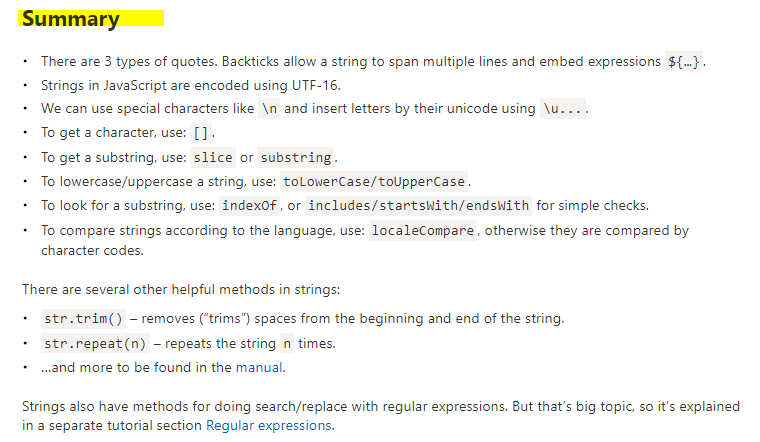


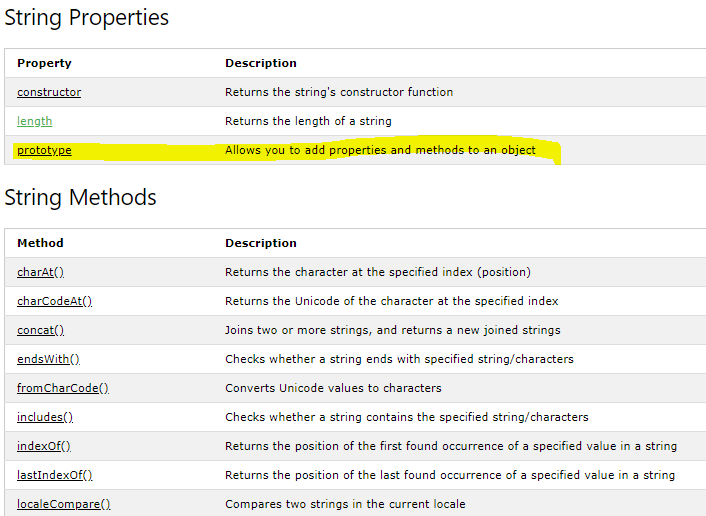


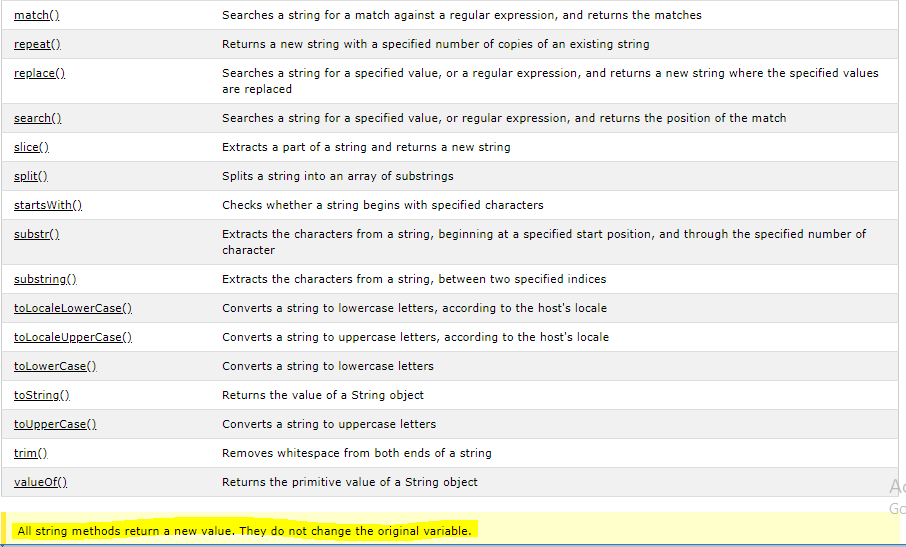












## **Coding Problem**

