

Cheatography

CoreJS Interview1 Cheat Sheet

by itsMe via cheatography.com/189476/cs/39515/

JS Basics

Falsy ⇒ "", 0, null, undefined, NaN, false

var - hoisting, let, const - blocks

Temporal Dead Zone начинается в начале области видимости переменной и заканчивается ее объявлением

Primitives: Undefined, Boolean, String, null, Null

Non Primitives:

Object, Array, Function, RegExp, Date,

Map, Set,

Error, Promise, enum

Strings

text.length

text.slice(s start, end) => new string [start, end)

text.substring(start, end) new string [start, end)

text.ubs tr(start, length) new string [start... length number of characters.

text.replace (oldValue, newValue) te xt.replace All (oldValue, newValue)

text.toUpperCase() text.toLowerCase()

se()

concat (string1, string2, ..., stringN)

text.trim() trimStart() trimEnd()

padStart(targetLength, padString): adds padding to the beginning

padEnd (targetLength, padString): adds padding to the end

charAt (index) charCodeAt(index)

split(separator, limit)

Timers

Методы setInterval(func, delay, ...args) и setTimeout(func, delay, ...args) позволяют выполнять func регулярно или только один раз после задержки delay, заданной в мс.

Для отмены выполнения необходимо вызвать clearInterval/clearTimeout со значением, которое возвращают методы setInterval/setTimeout.

Timers (cont)

Планирование с нулевой задержкой setTimeout(func, 0) делает то же самое, setTimeout(func) используется для выполнения захвата (capturing), цель должны быть исполнены как можно скорее, по завершении текущего кода.

Браузер ограничивает 4-мя мс минимальную задержку между вызовами обработчиков Symbol, BigInt и более вложенными вызовами setTimeout, а также для каждого события захват обработчики начиная с 5-го вызова.

```
let timerId = setTimeout(...); clearTimeout(timerId);
```

Objects

Creating Objects

Literal/ Constructor Function/ Class

```
{x: "y"} (new) function() {} new Object()
same as {} class Banana {} Object.create
te(proto[, proper tie object])
delete book. user reading
```

Обход объекта

```
for (let key in myObject) Object.entries()
s() => [key, value]
Object.keys() => [key] Object.values()
> [values]
```

Методы

```
someObj.hasOwnProperty('key1')
obj.__proto__ === person => Object.getPrototypeOf(obj)
Object.setPrototypeOf(obj, Object.prototype)
The valueOf() method of [Object] instances converts the this value [to an object]
```

Object.create()

```
function MyClass() {
  SuperClass.call(this);
  OtherSuperClass.mixin(this);
}
MyClass.prototype =
Object.create(SuperClass.prototype,
ass.prototype);
```

позволяет создавать новый объект и устанавливать прототип этого объекта в объект, переданный в качестве параметра

Events

Event Phases: фазы жизненного цикла события: capture (захват), target (цель), bubble (всплытие), cancel (отмена). Разница между этими фазами заключается в порядке выполнения обработчиков

Всплытие: сверху вниз, в фазе bubble. В фазе цели вызывается только один обработчик, связанный с элементом цели.

Mouse Events: - click - dblclick - mousedown - mouseup - mousemove - mouseover - mouseout - contextmenu

Keyboard Events: - keydown - keyup - keypress

Form: formdata reset submit;

Input: change, input, cut, copy, paste

Array NOT Mutating

```
concat(array1, array2, ..., arrayN)
array.fill(value, start, end)
Array
array.map(elt, index)
array.push(elt, index)
array.splice(start, end)
array.reduce(acc, current)
reduceRight(callback, initialValue)
```

Array Mutations

```
push(element1, element2, ..., elements)
pop()
shift()
unshift(element1, element2, ...)
splice(start, deleteCount, items)
sort(compareFunction)
reverse()
fill(value, start, end)
```



By itsMe
cheatography.com/itsme/

Published 9th July, 2023.
Last updated 9th July, 2023.
Page 1 of 2.

Sponsored by CrosswordCheats.com
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>

Iterate Array

```
for (let i = 0; i < scores.length; i++) {}  
for (i in scores) {}  
for (score of scores) {}  
scores.forEach(score =>  
{});  
every( function (element,  
index, array) { / ... / },  
thisArg)  
array.reduce(function(total,  
el, i, arr), initial Value)  
every( function (element,  
index, array) { / ... / },  
thisArg)
```

Copy Array

```
const arrNew = [...arrOld];  
const newArray = [...oldArray];  
{ const json = JSON.stringify -  
fy( old Array);  
const newArray = JSON.parse -  
e(j son); }  
const arrNew = arrOld.slice();  
const arrNew = Array.from(a -  
rrOld);  
const arrNew = [].concat(ar -  
rOld);
```



By **itsMe**
cheatography.com/itsme/

Published 9th July, 2023.
Last updated 9th July, 2023.
Page 2 of 2.

Sponsored by **CrosswordCheats.com**
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>