Object.keys()

O método **`Object.keys()`** retorna um array de propriedades enumeraveis de um determinado objeto, na mesma ordem em que é fornecida por um laço `for...in` (a diferença é que um laço for-in enumera propriedades que estejam na cadeia de protótipos).

Sintaxe

```
Object.keys(obj)
```

Parametros

`obj`

O objeto cujas propriedades são enumeráveis.

Descrição

`Object.keys()` retorna um array cujo os elementos são strings correspondentes para a propriedade enumerável encontrada diretamento sobre o objeto. A ordenação das propriedades é a mesma que a dada pelo loop sobre as propriedades do objeto manualmente.

Exemplos

```
var arr = ['a', 'b', 'c'];
console.log(Object.keys(arr)); // console: ['0', '1', '2']

// array com objeto
var obj = { 0: 'a', 1: 'b', 2: 'c' };
console.log(Object.keys(obj)); // console: ['0', '1', '2']

// array como objeto com ordenação aleatória por chave
var an_obj = { 100: 'a', 2: 'b', 7: 'c' };
console.log(Object.keys(an_obj)); // console: ['2', '7', '100']

// getFoo é uma propriedade que não é enumerável
var my_obj = Object.create({}, { getFoo: { value: function() { return
my_obj.foo = 1;
```

Notas

Em ES5, Se o argumento para o método this não é um objeto(um primitivo), em seguida ele irá causar um `TypeError`. Em ES2015, um argumento não-objeto será forçado a um objeto.

```
Object.keys("foo");
// TypeError: "foo" is not an object (ES5 code)

Object.keys("foo");
// ["0", "1", "2"]                    (ES2015 code)
```

Polyfill

Para adicionar suporte Object.keys compatíveis em ambientes mais antigos que não têm suporte nativo para isso, copie o seguinte trecho:

```
// De https://developer.mozilla.org/en-US/docs/Web/JavaScript/Refe
if (!Object.keys) {
  Object.keys = (function() {
    'use strict';
    var hasOwnProperty = Object.prototype.hasOwnProperty,
        hasDontEnumBug = !({ toString: null }).propertyIsEnumerable('t
        dontEnums = [
          'toString',
          'toLocaleString',
          'valueOf',
          'hasOwnProperty',
          'isPrototypeOf',
          'propertyIsEnumerable',
          'constructor'
        ],
        dontEnumsLength = dontEnums.length;

    return function(obj) {
```

```
        }
      }

      if (hasDontEnumBug) {
        for (i = 0; i < dontEnumsLength; i++) {
          if (hasOwnProperty.call(obj, dontEnums[i])) {
            result.push(dontEnums[i]);
          }
        }
      }
      return result;
    };
  }());
}
```

Por favor, note que o código acima inclui chaves não-enumeráveis no IE7 (e talvez IE8), ao passar em um objeto a partir de uma janela diferente.

Para um simples Browser Polyfill, veja Javascript - Object.keys Browser Compatibility .

Especificações

Especificação	Status	Comentário
ECMAScript 5.1 (ECMA-262) The definition of 'Object.keys' in that specification.	Padrão	Definição inicial. Implementado em JavaScript 1.8.5.
ECMAScript 2015 (6th Edition,		

- Dispositivo móvel

Caracteristica	Chrome	Firefox (Gecko)	Internet Explorer	Opera	Safari
Suporte básico	5	4.0 (2.0)	9	12	5

Caracteristica	Android	Chrome para Android	Firefox Mobile (Gecko)	IE Mobile	Opera Mobile	Safari Mobile
Suporte básico	?	?	?	?	?	?

Veja também

- Enumerability and ownership of properties
- `Object.prototype.propertyIsEnumerable()`
- `Object.create()`
- `Object.getOwnPropertyNames()`