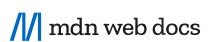


MDN Plus now available in [your](#) country! Support MDN [and](#) make it your own. [Learn more](#) 🌟



Object.keys()

The **Object.keys()** method returns an array of a given object's own enumerable property **names**, iterated in the same order that a normal loop would.

Try it

JavaScript Demo: Object.keys()

```
1 const object1 = {  
2   a: 'somestring',  
3   b: 42,  
4   c: false  
5 };  
6  
7 console.log(Object.keys(object1));  
8 // expected output: Array ["a", "b", "c"]  
9
```

Run >

Reset

> Array ["a", "b", "c"]

Syntax

`Object.keys(obj)`

Parameters

`obj`

The object of which the enumerable's own properties are to be returned.

Return value

An array of strings that represent all the enumerable properties of the given object.

Description

`Object.keys()` returns an array whose elements are strings corresponding to the enumerable properties found directly upon `object`. The ordering of the properties is the same as that given by looping over the properties of the object manually.

Examples

Using Object.keys

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

// array-like object
const obj = { 0: 'a', 1: 'b', 2: 'c' };
console.log(Object.keys(obj)); // console: ['0', '1', '2']

// array-like object with random key ordering
const anObj = { 100: 'a', 2: 'b', 7: 'c' };
console.log(Object.keys(anObj)); // console: ['2', '7', '100']

// getFoo is a property which isn't enumerable
const myObj = Object.create({}, {
  getFoo: {
    value() { return this.foo; }
  }
});
myObj.foo = 1;
console.log(Object.keys(myObj)); // console: ['foo']
```

If you want *all* properties—including non-enumerables—see [Object.getOwnPropertyNames\(\)](#).

Non-object coercion

In ES5, if the argument to this method is not an object (a primitive), then it will cause a [TypeError](#).

From ES2015 onwards, a non-object argument will be coerced to an object.

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

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

Specifications

Specification
ECMAScript Language Specification # sec-object.keys

Browser compatibility

[Report problems with this compatibility data on GitHub](#)

	Chrome 5	Edge 12	Firefox 4	Internet Explorer 9	Opera 12	Safari 5	Chrome 18 Android	Firefox 4 Android	Opera 12 Android
keys									

Full support

See also

- [Polyfill of Object.keys in core-js](#)
- [Enumerability and ownership of properties](#)
- [Object.prototype.propertyIsEnumerable\(\)](#)
- [Object.create\(\)](#)
- [Object.getOwnPropertyNames\(\)](#)
- [Object.values\(\)](#)
- [Object.entries\(\)](#)

Last modified: Jul 16, 2022, [by MDN contributors](#)