

W3 Schools

Tutorials ▾ References ▾ Exercises ▾ Certificates ▾ Search... Sign In

HTML CSS JAVASCRIPT SQL PYTHON JAVA PHP HOW TO W3.CSS C C++ C# BOOTSTRAP REACT MYSQL JQUERY EXCEL

JS Advanced JS Functions JS Objects Object Definitions Object Constructors Object this Object Destructuring Object Prototypes Object Iterations Object Management Object Get / Set Object Protection Object Reference JS Classes JS Asynchronous JS Modules JS Meta & Proxy JS Typed Arrays JS HTML DOM JS Windows JS Web APIs JS AJAX JS JSON JS jQuery

actrise

## JavaScript Object Protection

◀ Previous Next ▶

### Object Protection Methods

```
// Prevents re-assignment
const car = {type:"Fiat", model:"500", color:"white"};

// Prevents adding object properties
Object.preventExtensions(object)

// Returns true if properties can be added to an object
Object.isExtensible(object)

// Prevents adding and deleting object properties
Object.seal(object)

// Returns true if object is sealed
Object.isSealed(object)

// Prevents any changes to an object
Object.freeze(object)

// Returns true if object is frozen
Object.isFrozen(object)
```

### Using const

The most common way to protect an object from being changed is by using the `const` keyword.

With `const` you can not re-assign the object, but you can still change the value of a property, delete a property or create a new property.

### JavaScript Object.preventExtensions()

The `Object.preventExtensions()` method prevents adding properties to an object.

#### Example

```
// Create Object
const person = {firstName:"John", lastName:"Doe"};

// Prevent Extensions
Object.preventExtensions(person);

// This will throw an error
person.nationality = "English";
```

Try it Yourself »

Since arrays are objects, arrays can be prevented from extensions too:

#### Example

```
// Create Array
const fruits = ["Banana", "Orange", "Apple", "Mango"];
Object.preventExtensions(fruits);

// This will throw an error:
fruits.push("Kiwi");
```

Try it Yourself »

### JavaScript Object.isExtensible()

You can use `Object.isExtensible()` to check if an object is extensible.

The `Object.isExtensible()` returns true if an object is extensible.

#### Examples

```
// Create Object
const person = {firstName:"John", lastName:"Doe"};

// Prevent Extensions
Object.preventExtensions(person);

// This will return false
let answer = Object.isExtensible(person);
```

Try it Yourself »

```
// Create Array
const fruits = ["Banana", "Orange", "Apple", "Mango"];
Object.preventExtensions(fruits);

// This will return false
let answer = Object.isExtensible(fruits);
```

Try it Yourself »

### JavaScript Object.seal()

The `Object.seal()` method prevents additions or deletions of new properties.

The `Object.seal()` method makes existing properties non-configurable.

The `Object.isSealed()` method can be used to check if an object is sealed.

**Note**

The `Object.seal()` method will fail silently in non-strict mode and throw a `TypeError` in strict mode.

#### Example

```
"use strict"
// Create Object
const person = {
  firstName: "John",
  lastName: "Doe",
  age: 50,
  eyeColor: "blue"
};

// Seal Object
Object.seal(person);

// This will throw an error
delete person.age;
```

Try it Yourself »

Since arrays are objects, arrays can be sealed too:

#### Example

```
// Create Array
const fruits = ["Banana", "Orange", "Apple", "Mango"];
Object.seal(fruits);

// This will throw an error:
fruits.push("Kiwi");
```

Try it Yourself »

### JavaScript Object.isSealed()

The `Object.isSealed()` method can be used to check if an object is sealed.

The `Object.isSealed()` returns true if an object is sealed.

#### Examples

```
// Create Object
const person = {firstName:"John", lastName:"Doe"};

// Seal Object
Object.seal(person);

// This will return true
let answer = Object.isSealed(person);
```

Try it Yourself »

```
// Create Array
const fruits = ["Banana", "Orange", "Apple", "Mango"];
Object.seal(fruits);

// This will return true
let answer = Object.isSealed(fruits);
```

Try it Yourself »

### JavaScript Object.freeze()

The `Object.freeze()` method prevents any changes to an object.

Frozen objects are read-only.

No modification, addition or deletion of properties are allowed.

**Note**

The `Object.freeze()` method will fail silently in non-strict mode and throw a `TypeError` in strict mode.

#### Example

```
"use strict"
// Create Object
const person = {
  firstName: "John",
  lastName: "Doe",
  age: 50,
  eyeColor: "blue"
};

// Freeze Object
Object.freeze(person);

// This will throw an error
person.age = 51;
```

Try it Yourself »

Since arrays are objects, arrays can be frozen too:

#### Example

```
const fruits = ["Banana", "Orange", "Apple", "Mango"];
Object.freeze(fruits);

// This will throw an error:
fruits.push("Kiwi");
```

Try it Yourself »

### JavaScript Object.isFrozen()

The `Object.isFrozen()` method can be used to check if an object is frozen.

The `Object.isFrozen()` returns true if an object is frozen.

#### Examples

```
// Create Object
const person = {firstName:"John", lastName:"Doe"};

// Freeze Object
Object.freeze(person);

// This will return true
let answer = Object.isFrozen(person);
```

Try it Yourself »

```
// Create Array
const fruits = ["Banana", "Orange", "Apple", "Mango"];
Object.freeze(fruits);

// This will return true
let answer = Object.isFrozen(fruits);
```

Try it Yourself »

**Learn More:**

[JavaScript Object Definitions](#)  
[JavaScript Object Constructors](#)  
[JavaScript Object this](#)  
[JavaScript Object Destructuring](#)  
[JavaScript Object Prototypes](#)  
[JavaScript Object Iterations](#)  
[JavaScript Object Management](#)  
[JavaScript Object Getters & Setters](#)  
[JavaScript Object Protection](#)  
[JavaScript Object Reference](#)

◀ Previous Sign in to track progress Next ▶

W3 Schools

PLUS SPACES GET CERTIFIED FOR TEACHERS FOR BUSINESS

CONTACT US

Top Tutorials Top References Top Examples

HTML Tutorial CSS Tutorial JavaScript Tutorial How To Tutorial SQL Tutorial Python Tutorial W3.CSS Tutorial Bootstrap Tutorial HTML Tutorial Java Tutorial C++ Tutorial jQuery Tutorial

HTML Reference CSS Reference JavaScript Reference SQL Reference Python Reference W3.CSS Reference Bootstrap Reference PHP Reference HTML Colors Java Reference AngularJS Reference JQuery Reference

HTML Examples CSS Examples JavaScript Examples SQL Examples Python Examples W3.CSS Examples Bootstrap Examples PHP Examples Java Examples XML Examples JQuery Examples

Get Certified

HTML Certificate CSS Certificate JavaScript Certificate