# JavaScript Function Apply

❮ Previous Next ❯

## Method Reuse

With the **apply()** method, you can write a method that can be used on different objects.

## The JavaScript apply() Method

The **apply()** method is similar to the **call()** method (previous chapter).

In this example the **fullName** method of **person** is **applied** on **person1**:

### Example

```
var person = {
    fullName: function() {
        return this.firstName + " " + this.lastName;
    }
}
var person1 = {
    firstName: "Mary",
    lastName: "Doe",
}
person.fullName.apply(person1);  // Will return "Mary Doe"
```

Try it Yourself »

## The Difference Between call() and apply()

The difference is:

The call() method takes arguments **separately**.

The apply() method takes arguments as an **array**.

The apply() method is very handy if you want to use an array instead of an argument list.

## The apply() Method with Arguments

The **apply()** method accepts arguments in an array:

### Example

```
var person = {
    fullName: function(city, country) {
        return this.firstName + " " + this.lastName + "," + city + "," +
country;
    }
}
var person1 = {
    firstName:"John",
    lastName: "Doe",
}
person.fullName.apply(person1, ["Oslo", "Norway"]);
```

Try it Yourself »

Compared with the **call()** method:

### Example

```
var person = {
    fullName: function(city, country) {
        return this.firstName + " " + this.lastName + "," + city + "," +
country;
```

```
    }
}
var person1 = {
    firstName:"John",
    lastName: "Doe",
}
person.fullName.call(person1, "Oslo", "Norway");
```

Try it Yourself »

## Simulate a Max Method on Arrays

You can find the largest number (in a list of numbers) using the Math.max() method:

### Example

```
Math.max(1,2,3);  // Will return 3
```

Try it Yourself »

Since JavaScript **arrays** do not have a max() method, you can apply the Math.max() method instead.

### Example

```
Math.max.apply(null, [1,2,3]); // Will also return 3
```

Try it Yourself »

The first argument (null) does not matter. It is not used in this example.

These examples will give the same result:

## Example

```
Math.max.apply(Math, [1,2,3]); // Will also return 3
```

Try it Yourself »

## Example

```
Math.max.apply(" ", [1,2,3]); // Will also return 3
```

Try it Yourself »

## Example

```
Math.max.apply(0, [1,2,3]); // Will also return 3
```

Try it Yourself »

## JavaScript Strict Mode

In JavaScript strict mode, if the first argument of the apply() method is not an object, it becomes the owner (object) of the invoked function. In "non-strict" mode, it becomes the global object.

❮ Previous Next ❯