



链滴

# 浅谈 JavaScript 中的 call()与 apply()

作者: [cdq](#)

原文链接: <https://ld246.com/article/1634980736877>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

- call()方法

- 语法:

```
function.call(thisArg, arg1, arg2, ...)
```

用于Function对象, thisArg: 在function函数运行时, 指定的this值, 将函数的上下文由初始的改为thisArg指定的对象。

arg1, arg2: 指定的参数列表。

- 作用:

```
obj1.call(obj2,arg1,arg2,...)
```

我的理解就是使得obj1的this指向obj2, 也即是obj1可以访问obj2中的函数和属性。

```
function test(){
  alert(this);
  alert(this.name);
  alert(this.age)
};
```

```
var lisi= {name: "李四", age: 20};
// 更改内部的this指针引用对象为obj
test.call(lisi);
```

像上面的代码, test.call(lisi)表示的就是将test的this指向lisi中, 即在test中可以访问lisi中的值, 这就改变了test的上下文。call()还可以用于类的继承, 可以调用父构造函数。可以不指定call()的第一个参数, 但是这样做在严格模式和非严格模式中会有差别, 在非严格模式中, 不指定call()的第一个参数, 会得this的值会被绑定为全局对象; 而在严格模式下this的值是undefined。

使用call()还可以实现继承:

```
function class1(){
  this.setText = function(txt){
    alert(txt)
  }
}
```

```
function class2() {
  class1.call(this)
}
```

```
var dd = new class2()
dd.setText("dd")
```

这样 class2 就继承 class1 了, class1.call(this) 的意思就是将 this 使用class1 代替, 那么在 class2 也就能使用 class1 中的函数了, 甚至可以实现多重继承。

- apply() 方法:

- 语法:

```
func.apply(thisArg, [argsArray])
```

它与 call() 的区别在于，它传入的第二个参数必须是数组，其他的用法和 call() 差不多。

- 

```
var nums = [1,2,3,4]
console.log(Math.max.apply(null,nums))
```

可以使用 apply() 来进行查找数组中的最大值。

- 

```
var ddd = ['a','b']
var aaa = ['c','d']
ddd.push.apply(ddd,aaa)
```

了解的还比较浅，如有问题，望指出，感激不尽。