

# this详解

---

## 一、函数调用分为四种情况

- 作为函数调用
- 作为方法调用
- 作为构造函数
- 通过它们的call()和apply()方法间接调用

### 1.作为函数的调用

在ES3以及非严格模式的ES5中，函数调用的执行上下文是全局对象，即在该情况下浏览器环境中函数调用的this为Window对象。而在严格模式下函数调用的this为undefined

以函数调用形式的函数通常不使用this，但是可以根据this来判断当前是否是严格模式。

```
// 定义并调用一个函数来确定当前脚本是否处于严格模式下执行
let strict = (function(){return !this})
```

### 2.作为方法调用

当函数挂载到一个对象上，作为对象的属性，则称该函数为对象的方法。如果通过对象来调用函数时，该对象就是本次调用的上下文，被调用函数的this也就是该对象。

### 3.作为构造函数

### 4.通过它们的call()和apply()方法间接调用

```
<script>
  var a = 1
  var obj = {
    a:2,
    m:function () {
      var a=3
      console.log(this === obj)
      console.log(this.a)
    }
  }
  obj.m()
</script>
```