

this

this的指向在函数定义的时候是确定不了的，只有在函数执行的时候才能确定this指向的是谁，this的最终指向是调用它的对象。

```
function a(){
  var user = "追梦子";
  console.log(this.user); //undefined
  console.log(this); //Window
}
a();
```

(1) 如果一个函数中有this，但是它没有被上一级的对象所调用，那么this指向的就是window。

(2) 如果一个函数中有this，这个函数有被上一级的对象所调用，那么this指向的就是上一级的对象。

(3) 如果一个函数中有this，**这个函数中包含多个对象，尽管这个函数是被最外层的对象所调用，this指向的也只是它上一级的对象。**

也就是说含有this的函数在被调用的时候始终指向的是它的上一层对象。

```
var o = {
  a:10,
  b:{
    a:12,
    fn:function(){
      console.log(this.a); //undefined
      console.log(this); //window
    }
  }
}
var j = o.b.fn;
j();
```

this永远指向的是最后调用它的对象，也就是看它执行的时候是谁调用的，将fn赋值给变量j的时候并没有执行所以最终指向的是window。