# [JS匿名函数理解](http://www.cnblogs.com/ClareZjy/p/6365891.html)

匿名函数的基本形式为(function(){...})();

前面的括号包含函数体，后面的括号就是给匿名函数传递参数并立即执行之

匿名函数的作用是避免全局变量的污染以及函数名的冲突

1.小括号的作用

小括号能把我们的表达式组合分块，并且每一块，也就是每一对小括号，都有一个返回值。这个返回值实际上也就是小括号中表达式的返回值。所以，当我们用一对小括号把匿名函数括起来的时候，实际上小括号对返回的，就是一个匿名函数的Function 对象。因此，小括号对加上匿名函数就如同有名字的函数般被我们取得它的引用位置了。所以如果在这个引用变量后面再加上参数列表，就会实现普通函数的调用形式。

简单来说就是小括号有返回值，也就是小括号内的函数或者表达式的返回值，所以说小括号内的function返回值等于小括号的返回值

2.匿名函数的写法

方式1，调用函数，得到返回值。强制运算符使函数调用执行

(**function(x,y){**

alert(x+y);

**return x+y;**

}(3,4));

方式2，调用函数，得到返回值。强制函数直接量执行再返回一个引用，引用再去调用执行

(**function(x,y){**

alert(x+y);

**return x+y;**

})(3,4);

这种方式也是很多库爱用的调用方式，如jQuery，Mootools。

方式3，使用void

void **function(x) {**

x = x-1;

alert(x);

}(9);

方式4，使用-/+运算符

-**function(x,y){**

alert(x+y);

**return x+y;**

}(3,4);

+**function(x,y){**

alert(x+y);

**return x+y;**

}(3,4);

--**function(x,y){**

alert(x+y);

**return x+y;**

}(3,4);

++**function(x,y){**

alert(x+y);

**return x+y;**

}(3,4);

方式5，使用波浪符(~)

~**function(x, y) {**

alert(x+y);

**return x+y;**

}(3, 4);

方式6，匿名函数执行放在中括号内

[**function(){**

console.log(**this) // 浏览器得控制台输出window**

}(**this)]**

方式7，匿名函数前加**typeof**

**typeof function(){**

console.log(**this) // 浏览器得控制台输出window**

}(**this)**

方式8，匿名函数前加**delete**

**delete function(){**

console.log(**this) // 浏览器得控制台输出window**

}(**this)**

方式9，匿名函数前加void

void **function(){**

console.log(**this) // 浏览器得控制台输出window**

}(**this)**

方式10，使用**new方式，传参**

**new function(win){**

console.log(win) // window

}(**this)**

方式11，使用**new，不传参**

**new function(){**

console.log(**this) // 这里的this就不是window了**

}

方式12，逗号运算符

**function(){**

console.log(**this) // window**

}();

方式13，按位异或运算符

^**function(){**

console.log(**this) // window**

}();

方式14，比较运算符

**function(){**

console.log(**this) // window**

}();

最后看看错误的调用方式

**function(x,y){**

alert(x+y);

**return x+y;**

}(3,4);

匿名函数的N种写法如下所示

匿名函数没有实际名字，也没有指针，怎么执行？

关于匿名函数写法，很发散~

+号是让函数声明转换为函数表达式。汇总一下

最常见的用法：

代码如下:

(**function() {**

alert('water');

})();

当然也可以带参数：

代码如下:

(**function(o) {**

alert(o);

})('water');

想用匿名函数的链式调用？很简单：

代码如下:

(**function(o) {**

console.log(o);

**return arguments.callee;**

})('water')('down');

常见的匿名函数都知道了，看看不常见的：

代码如下:

~(**function(){**

alert('water');

})();//写法有点酷~

代码如下:

void **function(){**

alert('water');

}();//据说效率最高~

代码如下:

+**function(){**

alert('water');

}();

代码如下:

-**function(){**

alert('water');

}();

代码如下:

~**function(){**

alert('water');

}();

代码如下:

!**function(){**

alert('water');

}();

代码如下:

(**function(){**

alert('water');

}());//有点强制执行的味道~