[**js(=>) 箭头函数 详细解说 案例大全**](https://www.cnblogs.com/hailun/p/6279029.html)

ES6标准新增了一种新的函数：Arrow Function（箭头函数）。

为什么叫Arrow Function？因为它的定义用的就是一个箭头：

x => x \* x

上面的箭头函数相当于：

function (x) {

return x \* x;

}

箭头函数相当于匿名函数，并且简化了函数定义。箭头函数有两种格式，一种像上面的，只包含一个表达式，连{ ... }和return都省略掉了。还有一种可以包含多条语句，这时候就不能省略{ ... }和return：

[复制代码](javascript:void(0);)

x => {

if (x > 0) {

return x \* x;

}

else {

return - x \* x;

}

}

[复制代码](javascript:void(0);)

如果参数不是一个，就需要用括号()括起来：

[复制代码](javascript:void(0);)

// 两个参数:

(x, y) => x \* x + y \* y

// 无参数:

() => 3.14

// 可变参数:

(x, y, ...rest) => {

var i, sum = x + y;

for (i=0; i<rest.length; i++) {

sum += rest[i];

}

return sum;

}

[复制代码](javascript:void(0);)

如果要返回一个对象，就要注意，如果是单表达式，这么写的话会报错：

// SyntaxError:

x => { foo: x }

因为和函数体的{ ... }有语法冲突，所以要改为：

// ok:

x => ({ foo: x })