

# JavaScript函数的使用详解

作者：有故事的人 来源：范文网 www.wtabcd.cn/fanwen/

本文原地址：<https://www.wtabcd.cn/fanwen/zuowen/c67b2e38446d8b9e1d76ff783370605f.html>

范文网，为你加油喝彩！

目录

1. 声明函数  
2. 调用函数  
3. 函数的参数  
4. 函数的返回值  
5. arguments的使用  
6. 函数可以调用另外一个函数  
7. 函数的两种声明方式总结  
1. 声明函数  
2. 调用函数  
3. 函数的参数  
4. 函数的返回值  
5. arguments的使用  
6. 函数可以调用另外一个函数  
7. 函数的两种声明方式总结

## 1. 声明函数

```
//声明函数function 函数名() { //函数体代码}
```

function是声明函数的关键字，必须小写由于函数一般是为了实现某个功能才定义的，所以通常我们将函数名命名为动词，比如getsum

## 2. 调用函数

```
//调用函数函数名(); //通过调用函数名来执行函数体代码
```

声明函数本身并不会执行的代码，只有调用函数时才会执行函数体代码。

## 3. 函数的参数

在声明函数时，可以在函数名称后面的小括号中添加一些参数，这些参数被称为形参，而在调用该函数时，同样也需要传递相应的参数，这些参数被称为实参。

### 参数个数

实参数等于形参数输出正确结果  
实参数多于形参数只取到形参的个数  
实参数少于形参数多的形参定义为undefined，结果为nan  
形参的

参数的作用(在函数调用时某些值不能固定时，我们可以设置参数为undefined，这样在调用函数时传递实参的值进形参数，输出正确结果)  
sum(100,400,500,700); //实参数多于形参数只取到形参的个数  
sum(200); //实参数少于形参数多的形参定义为undefined，结果为nan

在javascript中，形参的默认值是undefined。

## 4. 函数的返回值

经常用return返回函数值

经常用return返回函数值

return终止函数

return终止函数

```
function add(num1,num2) { //函数体 return num1+num2; //注意：return后的代码不执行
```

```
function add(a,b){return a+b;}var resnum = add(12,6); //调用函数n后的代码不换行，并通过resnum接收  
函数返回值alert(resnum); //不会被执行';}var resnum = add(21,6); //调用函数，传入两个实参，并通过resnum接收  
函数返回值alert(resnum); //27
```

如果函数有return，则返回return后面的值；如果没有return，则返回 undefined。

如果函数有return，则返回return后面的值；如果没有return，则返回 undefined。

## break , continue , return 的区别

### break , continue , return的区别

`break`、`continue`、`return`的区别  
`break`：结束当前的循环体(如`for`、`while`)  
`continue`：跳出本次循环，继续执行下次循环(如`for`、`while`)  
`return`：不仅可以退出循环，还能够返回`return`语句中的值  
同时还可以结束当前的函数体内的代码

## 退出循环，还能能够返回值 5. arguments的使用

## 5. arguments的使用

当我们不确定有多少个参数传递的时候，可以用arguments来获取，在javascript中，arguments实际上它是当前函数的一个内置对象。所有函数都内置了一个arguments对象。arguments对象中存储了传递的所有实参。

## 6 函数可以调用另外一个函数

func函数可以调用另外一介函数

```
function fn1() { console.log('fn1'); } function fn2() { console.log('fn2'); } fn1(); fn2();
```

function fn1() { console.log(111); fn2(); console.log('fn1'); } function fn2() { console.log(222); console.log('fn2'); } fn1();

## 7. 函数的两种声明方式

## 7. 函数的两种声明方式

//1. 利用函数关键字自定义函数(命名函数)function fn() {}fn(); //2. 函数表达式(匿名函数)//var 变量名 = function

```
ation函数的两种声明方式 console.log('我是函数表达式'); console.log(aru); }fun('fg' );
```

//(1) fun是变量名不是函数名//(2) 函数表达式声明方式跟声明变量差不多，只不过变量里面存的是值而函数

//1. 利用函数关键字自定义函数(里面像的是函数)(3) 函数表达式或块式函数进行简述参数 /var 变量名 = fun

```
function() {} var fun = function(aru) { console.log('我是函数表达式'); console.log(aru); } fun('fa')
```

//(1) fun是变量名 不是函数名//(2) 函数表达式声明方式和普通变量差不多 只不过变量里面存的是值而函数

表达式里面存的是函数 // (3) 函数表达式也可以进行传递参数

长进到至高的定山教派,山教长进到已可以进行长进步数

**总结** 本篇文章就到这里了，希望能够给你带来帮助，也希望您能够多多关注www.887551.com的更多内容！

本更多作文请访问 [https://www.wtabcd.cn/fanwen/list/92\\_0.html](https://www.wtabcd.cn/fanwen/list/92_0.html) 的更多内容！

## 1. 声明函数

```
//声明函数function 函数名() { //函数体代码}
```

文章生成doc功能，由范文网开发

function是声明函数的关键字，必须小写由于函数一般是为了实现某个功能才定义的，所以通常我们将函数名命名为动词，比如getsum

## 2. 调用函数

```
//调用函数函数名(); //通过调用函数名来执行函数体代码
```

声明函数本身并不会执行的代码，只有调用函数时才会执行函数体代码。

## 3. 函数的参数

在声明函数时，可以在函数名称后面的小括号中添加一些参数，这些参数被称为形参，而在调用该函数时，同样也需要传递相应的参数，这些参数被称为实参。