

08-第八章 函数[自执行] [传参] [return] getComputedStyle()

一、函数自执行

函数自执行方式,即创建立即调用一次

- 1、函数后面加用小括号,然后在用小括号包起来

```
1. (function(){}())//函数后面加用小括号包起来
```

- 2、函数用小括号包起来,然后后面加小括号

```
1. (function(){}())//用小括号包起来
```

- 3、函数后面加小括号,然后在函数前面加 + - ~ ! 其中的一个符号

```
1. //+
2. +function(){}()
3.
```

```
1. //-
2. -function(){}()
```

```
1. //~
2. ~function(){}()
```

```
1. //!
2. !function(){}()
```

二、函数传参

1、对应传参

形参:即形式参数，是用来接收函数调用时传递过来的数据，命名与变量一致

实参:即真实参数，是给形参传递的具体数据，任何数据类型都可以称为实参

```
1. function fn( a,b){ //a,b为形参，且a 为 20，b为10，一一对应
2.     console.log(a); //20
3.     console.log(b); //10
4.     console.log(a+b); //30
5. }
6. fn(20,10); //20,10为实参
7.
```

2、不定参，arguments

1、不定参：实参个数不确定

```
1. function fn( a,b){ //a,b为形参，且a 为 20，b为10，一一对应
2.     console.log(a); //20
3.     console.log(b); //10
4.     console.log(a+b); //30
5. }
6. fn(20,10); //20,10为实参
7.
```

2、arguments 所有函数都存在arguments，是所有实参的集合，arguments是一个类数组，arguments.length 可以返回个数

```
1. function fn( a,b){ //a,b为形参，且a 为 20，b为10，一一对应
2.     console.log(a); //20
3.
4. }
5. fn(20,10); //20,10为实参
6.
```

三、函数中的return

1、函数中默认return的返回值为undefined

```
1. function fn( a,b){
2.     a+b;
3. }
4. var a = fn();
5. console.log( a );//undefined
```

2、自定义返回值

1)、有时候我们需要函数中返回我们需要的值，这个时候 return 很有用

```
1. function fn( a,b){
2.     a+b;
3. }
4. var a = fn();
5. console.log( a );//undefined
6.
```

2)、return 返回的数据类型 可以是任意类型

```

1.     function fn( a,b){
2.         a+b;
3.         return function(){ alert('ok') };
4.     }
5.     var a = fn();
6.     a();//此时a就是返回的函数
7.     //等同于
8.     fn()();
9.

```

3) **return** 然后的代码不再执行，整个函数结束

```

1.     function fn( a,b){
2.         a+b;
3.         return function(){};
4.         console.log('ok');//此代码不会再执行
5.     }
6.     fn();
7.

```

四、getComputedStyle()

getComputedStyle(obj,null) [cssStyle]获取计算后的样式对象,只读

- 不要获取复合样式:如 **background**
- 不要获取未设置的样式:如 **width** 谷歌是具体宽度,ie是auto

兼容 : ie8及以下 obj. **currentStyle** [cssStyle]

```

1. if(obj.currentStyle) {
2.     return obj.currentStyle[attr];
3. }else {
4.     return getComputedStyle(obj,null)[attr]; //传递null是为了
5. }

```