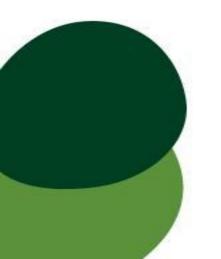


# JavaScript进阶

---JS函数对象





#### 内容提纲

- > 函数对象
- > 函数对象的属性及方法
- > 高阶函数



## 函数对象介绍

# •JS中的函数也是对象

- JS中每个函数都是作为对象来维护和运行的,即<mark>函数对象</mark>(既有属性也有方法)
- 可以将函数(函数对象)赋值给一个变量,或将函数作为参数进行传递
- 函数对象对应的类型是Function(类似于数组对象对应于Array、日期对象对应于Date)
- 如果变量是函数(函数对象)时,typeof此对象,返回function,而非object
- 内置的函数对象(Array、Function、Date等),内置的非函数对象(Math、JSON)

```
function foo(){}

console.log(foo); //function foo(){}

console.log(typeof foo); //function

console.log(foo instanceof Object); //true

console.log(foo instanceof Function); //true

console.log(foo === window.foo); //true
```



#### 内容提纲

- > 函数对象
- > 函数对象的属性及方法
- > 高阶函数



# 函数对象的属性及方法

# •函数对象的属性

- length、arguments (隐藏的局部变量)
- caller、callee (arguments的属性,常用于递归调用)
- prototype

参见实例demo06 Part1函数对象属性综述

# •函数对象的方法

- call、apply
- bind
- toString、valueOf

参见实例demo06 Part2 函数对象方法综述



#### 内容提纲

- > 函数对象
- > 函数对象的属性及方法
- > 高阶函数



#### 高阶函数

# •高阶函数是指至少满足下列条件之一的函数

- 函数作为参数被传递(最常见的形式:回调函数)
- 函数作为返回值输出(与闭包有紧密联系)

```
function add(x, y, f) {
    return f(x) + f(y);
}
add(2,3,function(x){
        return x+1;
     }
);
add(2,-3,Math.abs);
add(2,3,Math.sqrt);
```

```
var obj = \{x:23\};
var fun1 = function () {
   return function fun2() {
       return this.x;//若改为 return this;
obj.fun3 = fun1;
obj.fun4 = fun1();
console.log(obj.fun3());//输出什么,this是什么
console.log(obj.fun3()());//输出什么,this是什么
console.log(obj.fun4());//输出什么,this是什么
```







#### 作业

- •复习本章课件及练习
- •将自己的本章练习提交到个人仓库中
- •阅读《JavaScript权威指南》第8章中的8.7、8.8这 2个小节中的内容