函数

一日三省且温故知新



旦 1、什么是函数

函数(function), 也可以被称之为方法(method),或者过程(procedure)

是一段预定义好,并可以被反复使用的代码块。其中可以包含多条可执行语句

预定义好:事先声明好,但不被执行

反复使用:允许被多个地方(元素,函数中)所应用

代码块:允许包含多条可执行的代码

函数本质上是功能完整的对象





旦 2、函数的声明

语法:

```
function 函数名(){
      可执行语句;
ex:创建一个函数名称:sayHello,主体功能为,
向控制台上输出一句 Hello World
function printHello(){
      console.log( "hello" );
      console.log( "world" );
```

WEBHIJHER



旦 3、函数的调用

执行函数中的内容 任何 JS 的合法位置处,都允许调用函数 语法:函数名称();

```
function sayHello(){
     console.log( "hello" );
     console.log( "world" );
}
sayHello();
```

WEBRITHER



旦 4、定义函数的参数和返回值

转换函数

parseInt(数据);//将指定数据转换为 整数 parseFloat(数据);//将指定的数据转换为 小数 console.log("要打印的话");

1、定义带参数函数

2、带返回值的函数



4.1、定义带参数函数

```
function 函数名(参数列表声明){
  //代码块(函数体,功能体,方法体)
参数列表:由一或多个变量名称来组成
声明函数时定义的参数,可以称之为叫作"形参"(形式参数)
ex:
   function printInfo(userName, ueserPwd){
     console.log( '用户名:'+userName + '密码:'+userPwd);
                                    传递直接量
   printlnfo('Tom', '123');
                                    做实参
```

在调用函数时所传递的参数值,被称之为"实参"(实际参数)

4.2、带返回值的函数

1、声明

```
function 函数名(0或多个参数){
    //代码块;
    return 值;
}
```



2、调用

```
var 变量 = 函数名(参数);
```

```
function add( num1, num2){
    return num1 + num2;
}
```

```
var result = add(10, 20);
console.log(result); //输出30
```

旦 5、变量的作用域

1、什么是作用域

4、声明提前

2、函数作用域

5、按值传递

3、全局作用域



5.1、什么是作用域

作用域就是变量或函数的可访问范围。它控制着变量或函数的可见性和生命周期 在JS中,变量或函数的作用域可分为:

- 1、函数作用域,只在当前函数内可访问
- 2、全局作用域,一经定义,代码的任何位置都可以方式





5.1、函数作用域中的变量

函数作用域中的变量(局部变量)只在当前函数内可访问到, 离开此范围就无法访问了。

5.2、全局作用域中的变量

全局作用域中的变量, 称之为"全局变量",在代码的任何位置处都能访问

```
var sum = 0;
                                  //全局变量
function add(){
    sum = 1 + 2;
    console.log( sum );
                                  //正确
                                         不推荐使用
console.log( sum );
                                  //正确
function add(){
                                  //全局变量
    sum = 1 + 2;
                                 //必须调用一次
add();
console.log( sum );
                                  //正确
```

WEBRITHER



5.3、声明提前

JS在正式执行之前,会将所有var声明的变量和function声明的

函数, 预读到所在作用域的顶部

但是,对变量的赋值,还保留在原来的位置处

console.log(a); //不会出错,输出undefined var a=100;

console.log(a); //100;



var a; //仅声明提前

console.log(a); //undefined a=100; //赋值仍保留在原位置

console.log(a); //100



5.4、按值传递

传参时,实际上是将实参复制了一份副本传给了函数。在函数体内对变量进行修改,实际上是不会影响到外部的实参变量的

```
var n=100; //全局变量n
function fun(n){ //参数变量也是局部变量
n-=3; //修改的是局部变量n
console.log(n); //输出的是局部变量n
}
fun(n); //按值传递, 方法内输出97;
console.log(n); //输出全局变量的值: 100
```

334

旦 小结

运算符&表达式

什么是函数

函数的声明

函数的调用

定义函数的参数和返回值

变量的作用域