

千锋HTML5学院

第二阶段javascript课程课件



1

函数的概念，及作用

2

函数的定义

3

如何执行函数(调用)

4

函数的参数(实参，形参)

5

作用域

6

DOM的简单操作

7

递归

函数的概念和作用

□ 函数的概念

函数就是把完成特定功能的一段代码抽象出来，使之成为程序中的一个独立实体，起个名字（函数名）。可以在同一个程序或其他程序中多次重复使用（通过函数名调用）（函数是当它被调用时执行的可重复使用的代码块。）

□ 函数的作用

- 使程序变得更简短而清晰
- 有利于程序维护
- 可以提高程序开发的效率
- 提高了代码的重用性(复用性)

不使用函数，**JS**代码必须执行，且执行时间，执行哪部分代码都是不可控的。
使用函数后，浏览器不会直接执行它，只有在调用这个函数时，它才会被浏览器所执行。

函数的分类

□ 内置函数（系统函数，官方函数）

是官方提供好的函数，直接使用，

如： `alert()`

`isNaN()`：用于验证参数是不是NaN，是不是非法的数字的意思。`isNaN(“Hello”)` `isNaN(123)`

`document.write()` 等；

□ 自定义函数（用户自定义的函数）

用户根据实际需求，需要自己封装一个函数。

数学函数

- 设A, B是非空的数集, 如果按照某种确定的**对应关系f**, 使对于集合A中的任意一个数x, 在集合B中都有唯一确定的数y和它对应, 那么就称**f**为从集合A到集合B的一个函数。

记作 $y = f(x)$, $x \in A$

$$y=f(x)=5x+1 \quad y=f(x)=\tan x \quad y=f(x)=\sin x$$

$$y=f(x)=x^2 \quad z=f(x,y)=x^2+x+y^2+y$$

$$k=f(x,y,x)=x^3+x^2+x+y^3+y^2+x+z^3+z^2+z$$

函数的简单定义

- Javascript中函数的定义

```
function 函数名() {  
    代码块  
}
```

如:

```
function printOut(){  
    document.write("Hello World!");  
}
```

关键词 **function** ， JavaScript 大小写敏感

函数实现了一个特定的功能, 函数可以封装任意多条语句, 而且可以在任何地方、任何时候调用执行

函数的简单调用

- Javascript中函数的调用
调用方式: 函数名()

如以下函数的调用方式:

`printOut()`

```
function printOut(){  
    document.write("Hello World!");  
}
```

函数的标准定义

□function关键字定义一个函数

```
function 函数名(参数1名字, 参数2名字, ……) //函数可以没有形参  
{  
    语句  
    [return 返回值;] //函数可以没有返回值(undefined)  
}
```

□示例:

```
function sum(one, two) {  
    var s = one+two;  
    return s;  
}
```

函数名

参数

返回值

参数是已知条件，
返回值是函数的执行结果

函数

示例：

- 1, 写一个函数计算1到n的和, 并返回结果打印出来;
(n为函数参数)
- 2, 写一个函数计算n的阶乘, 并返回结果打印出来
- 3, 写一个函数计算两个数的最小公倍数; 并返回结果打印出来

形参，实参

□形参和实参：

函数的参数跟变量是一样使用。

形参就是在函数定义时，函数名后面的参数，不能用**var**修饰。

实参就是调用时，函数名后面的参数

□参数可以传递多个，与函数的定义无关

如：

```
function test(paramX,paramY){  
}  
test("a","b","c");
```



形参，实参

- javascript函数不介意传递进来多少个参数，也不在乎传进来参数是什么数据类型，在调用这个函数时也未必一定要传递两个参数，原因是 **ECMAScript** 中的参数在内部是用一个数组来表示的。函数接收到的始终都是这个数组，而不关心数组中包含哪些参数（如果有参数的话）。

使用**arguments**对象可以判断参数的个数，**arguments**是个数组。

`arguments`

`arguments.length` 参数的个数

`arguments[i]` 访问第*i*个参数

练习：向一个函数传入不定数量的数值求和。

形参，实参

- 注意：

在一般传值调用的机制中只能把实参传送给形参，而不能把形参的值反向地传送给实参。因此在函数调用过程中，形参值发生改变，而实参中的值不会变化。

```
function addNum(n){  
    n += 5  
}  
var a=10;  
addNum(a);  
alert(a);
```

函数的返回值

- 如果我们要是外部获取到函数内运行的结果,可以通过**return**语句跟后面的要返回的值来实现返回值。

return 返回结果;

- 如:

$y = f(x)$ 函数 $f(x)$ 返回了一个值赋给 y

$sum = sum(x,y)$ 执行了一个求和

有参函数的调用

□ 通过在程序中使用函数名称，可以执行函数中包含的语句，这称为调用函数，同时传入实参。

函数名(参数1值，参数2值，……)；

□ 示例：

```
alert(“亲”)； //这句话是调用内置函数  
sum(2,8)； //计算2+8的结果。
```

作用域

- ```
var a = 10;
function m1(){
 var a = 5;
 alert(a);
}
m1();
alert(a); //从下往上寻找
```

## 作用域

□作用域：就是起作用的范围。或者说有效范围。

### □局部变量

局部变量就是定义在函数内部的变量，这个变量只能在函数内部使用，即作用域范围只是函数内部，另外，形参也是局部变量。

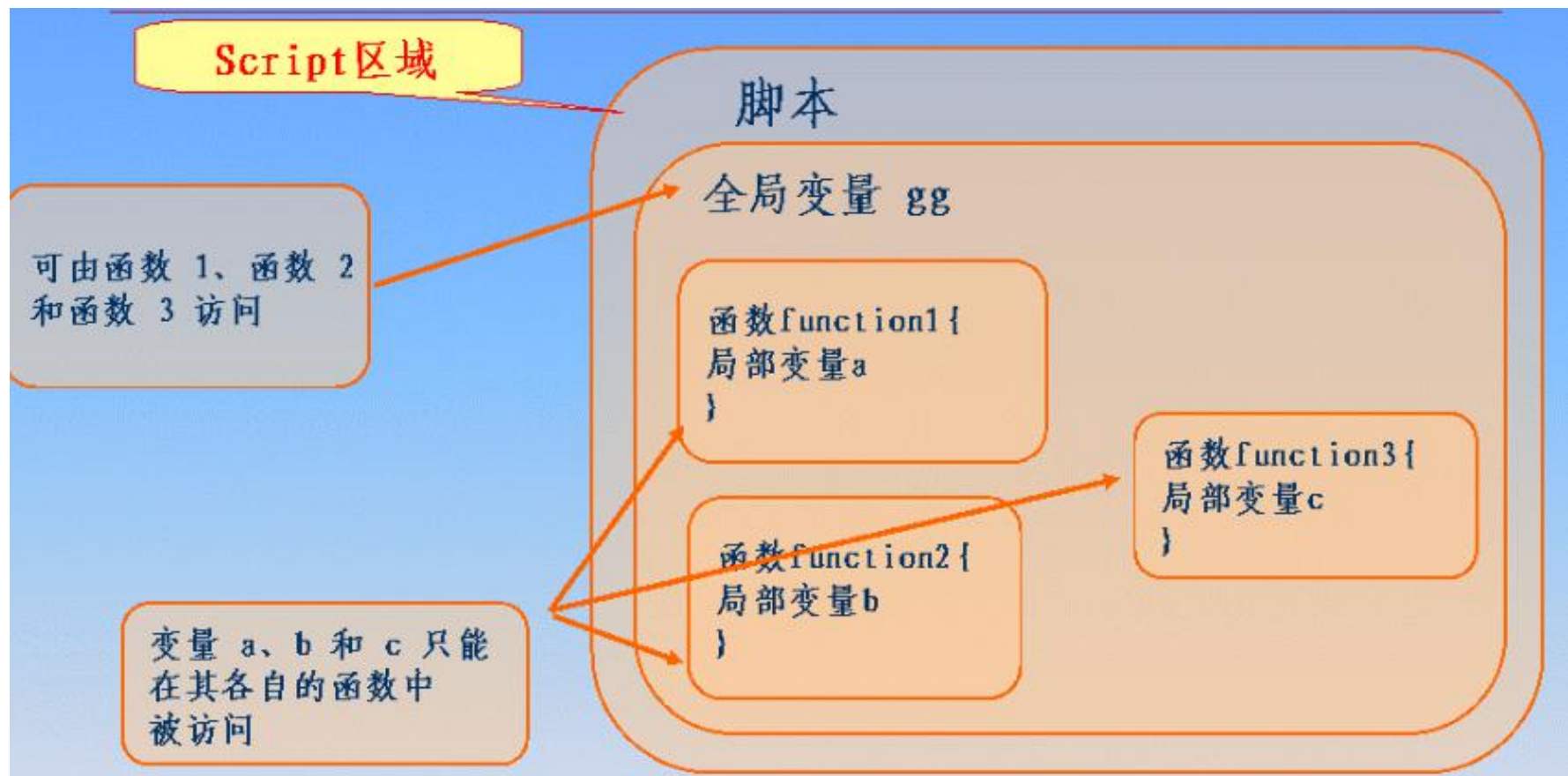
### □全局变量

全局变量就是定义在函数外部的变量，这个变量在任何函数中都有效，即作用域范围是当前文件的任何地方。

注意：定义变量时省略**var**是不安全的，不过是合法的，不写**var**的时候会被解释是全局变量



## 作用域



## 练习

- 写一个函数判断一个年份是不是闰年
- 写一个函数判断一个数是不是素数(又称质数,除了1和本身以外不再有其他数整除)
- 年月日分别为自定义函数的参数, 判断某一个日期是否为正确的日期;  
如: 2016年12月33日不是正确的日期  
2016年11月13日是正确的日期

把函数写在单独的JS的文件中,  
可以体现函数的优势

## 函数的嵌套调用

### □概念：

函数内部可以再包含其他函数

函数之间允许相互调用，也允许向外调用，不可以调用同级函数的嵌套函数

## DOM简单操作

### □ 访问元素节点的属性

- `document.getElementById("")` 获取特定元素的节点
- `value`: 文本框的`value`值
- `onclick`: 点击事件

`document.write()` 在文档页面加载完后使用会覆盖页面内容

- 在输入框中输入数字, 点击按钮调用函数来判断奇偶性?

## 事件驱动

### □事件驱动：

所有的函数，没有调用不会执行，那么函数调用的源头在何处，就是事件， 现在学习了解下事件  
onclick：点击事件

示例： 点击搜索按钮调用函数清空文本框的内容。

## 函数的递归调用

### □概念：

函数还可以自己调用自己，称为递归调用

### □重要性：

递归的分量，递归属于函数中比较难理解的知识，在应用开发中，虽然使用不是很频繁，但是很体现你的功底，而且，从事IT行业开发，最好要会递归，如果说现在可以不要求灵活运用的话，以后到公司中一定要会，如果面试中有人问你递归，说明，他对你要求挺高。

## 函数的递归调用（方法）

方法：

1. 首先去找临界值，即无需计算，获得的值。
2. 找这一次和上一次的关系（一般从后往前找）
3. 假设当前函数已经可以使用，调用自身计算上一次的运行结果，再写出这次的运行结果。

## 函数的递归调用

□ 示例 (阶乘) :

5的阶乘是多少?

$$5 \text{ 的阶乘} = 5 * 4 * 3 * 2 * 1$$

$$4 \text{ 的阶乘} = 4 * 3 * 2 * 1$$

$$3 \text{ 的阶乘} = 3 * 2 * 1$$

$$2 \text{ 的阶乘} = 2 * 1$$

$$1 \text{ 的阶乘} = 1$$

$$0 \text{ 的阶乘} = 1$$

1, 临界条件:  $1! = 1$   
2,  $n$  的阶乘 =  $n * (n-1)$  的阶乘;



## 函数的递归调用

用递归实现:

- 1, 计算 $1+2+3+\dots+100$ 的和
- 2, 输入一个 $n$ , 打印 $n$ 个hello world

## 函数的递归调用

### □强调：

【注】函数的递归调用只能用于静态的数据运算，如果是动态数据，风险过高，容易崩溃。

【功用】凡是循环能做到的，递归都能做到。

### \*拓展训练

兔子繁殖问题，设有一对新生兔子，从第四个月开始他们每个月月初都生一对兔子  
新生的兔子从第四个月月初开始又每个月生一对兔子  
按此规律，并假定兔子没有死亡, $n(n \leq 20)$ 个月月末共有多少对兔子？

## 作业

必做：

- 1, 写个函数计算所有传入参数相乘的结果(参数数量不定)
- 2, 写个函数实现加减乘除运算
- 3, 写个函数实现n的阶乘
- 4, 找出0-100之间7的倍数, 和包含7的数字
- 5、不用for循环, 计算0-100以内的数字和

可选：

- 1、制作一个简易计算器（计算两个数的加减乘除, 带可手动输入数字和运算符）
- 2、有一堆桃子不知数目, 猴子第一天吃掉一半, 觉得不过瘾, 又多吃了一只, 第二天照此办法, 吃掉剩下桃子的一半另加一只, 天天如此, 到第 $n$  ( $n \leq 10$ ) 天早上, 猴子发现只剩一只桃子了, 问这堆桃子原来有多少只?  
(设还剩 $m$ 天吃完的桃子数,  $m/2-1$ ( $m-1$ 天的桃子数),  $\text{numP}(n) = (\text{numP}(n-1) + 1) * 2$ )

