第四组：函数：（杨兵、王勇武）；条件语句（曹白雨、吴喆枭）

# 一、函数

## 一、定义：

函数是由事件驱动的或者当它被调用时执行的可重复使用的代码块。

## 二、语法：

函数声明、循环体、调用函数；

函数就是包裹在花括号中的代码块，前面使用了关键词 function：

function functionname()//函数声明

{//循环体

这里是要执行的代码

}

functionname()//调用函数

当调用该函数时，会执行函数内的代码。

**提示：**JavaScript 对大小写敏感。关键词 function 必须是小写的，并且必须以与函数名称相同的大小写来调用函数。

### 2.1函数声明：

特点：函数声明会自动提前；一次声明，多次调用；

优点：1、减少代码冗余；2、隐藏实现细节，封装。

#### 2.1.1无参函数

function box();{

alert();

}

#### 2.1.2有参函数

function box(a,b);{//形参，是没有初始值的,多个函数用逗号隔开

document.write(a\*b);

}

box(5,6);

#### 比较：\*有参函数与无参函数，扩展性增大！

#### 注意：声明会提前

#### 当函数内部变量与外部相同时，会优先调用内部变量；而内部变量如果没有先赋值，则会显示undifined

例：

var num=10;

function demo(){

document.write(num);

var num=15;

}

demo();

显示：undifined

若将变量先提前，则显示15；

var num=10;

function demo(){

var num=15;

document.write(num);

}

demo();

#### 2.1.3带有返回值的函数

通过return进行返回

Ps：return语句之后的代码不会被执行，终止函数。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*return的使用\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

return 语句从当前函数退出，并从那个函数返回一个值。

语法：

return[()[expression][]];

可选项 expression 参数是要从函数返回的值。如果省略，则该函数不返回值。

用 return 语句来终止一个函数的执行，并返回 expression 的值。如果 expression 被省略，或在函数内没有 return 语句被执行，则把值 undefined 赋给调用当前函数的表达式。

1、终止函数的继续运行.

当遇到if…… else是。若出现return，就会出现终止运行，不会继续做出判断

例：

function user()

{

if(name=="")

{

alert('用户名为空');

return;

}

if(password=="")

{

alert('密码为空');

return;

}

alert('登陆成功');

}

PS：不加return的现象是先提示用户名没输入，然后提示密码没输入；加了return之后遇到一个没输入之后就不再继续检测

当返回的是true时，将继续操作。

当返回是false时，将中断操作。

2、返回函数值

返回函数中的值，外部调用函数时，则会调用这个值。闭包中就是运用这个属性，调用函数内部的局部变量，以不影响全局变量。

## 2.2循环体

## 2.3调用函数

## 2.4构造函数（Object()）

1.构造函数的命名必须和类名完全相同。在java中普通函数可以和构造函数同名，但是必须带有返回值；

2.构造函数不能被直接调用，必须通过new[运算符](http://baike.baidu.com/view/425996.htm)在创建对象时才会自动调用；而一般的方法是在程序执行到它的时候被调用的

var box=new Object();//构造函数

box.age=18;

box.name="haha";

box.sex="女";

box.run=function(){

document.write("name:"+this.name+",age:"+this.age);

}

#### 2.4.1构造函数可以判断类别

function User(name,age,sex){

this.name=name;

this.age=age;

this.sex=sex;

}

function Goods(name,price,number){

this.name=name;

this.price=price;

this.number=number;

}

var box1=new User("hghg",78,"woman");

var box2=new Goods("apple",12,12);

document.write(box1 instanceof Object);//true

document.write(box1 instanceof Goods);//false

### 2.5 函数重载

#### \*在java中允许同名的函数，在javascript中不允许同名函数

# 二、条件语句

## 定义：

条件语句用于基于不同的条件来执行不同的动作。

## 分类:

在 JavaScript 中，我们可使用以下条件语句：

if 语句 - 只有当指定条件为 true 时，使用该语句来执行代码

if...else 语句 - 当条件为 true 时执行代码，当条件为 false 时执行其他代码

if...else if....else 语句 - 使用该语句来选择多个代码块之一来执行

switch 语句 - 使用该语句来选择多个代码块之一来执行

### 2．1 if 语句

只有当指定条件为 true 时，该语句才会执行代码

#### 2．1.1语法：

if (条件)

{

只有当条件为 true 时执行的代码

}

**注意：**请使用小写的 if。使用大写字母（IF）会生成 JavaScript 错误！

例1：

当时间小于 20:00 时，生成一个“Good day”问候：

if (time<20)

{

x="Good day";

}

### 2.2 If...else 语句

请使用 if....else 语句在条件为 true 时执行代码，在条件为 false 时执行其他代码。

#### 2.2.1语法

if (条件)

{

当条件为 true 时执行的代码

}

else

{

当条件不为 true 时执行的代码

}

例：

当时间小于 20:00 时，将得到问候 "Good day"，否则将得到问候 "Good evening"。

if (time<20)

{

x="Good day";

}

else

{

x="Good evening";

}

### 2.3 If...else if...else 语句

使用 if....else if...else 语句来选择多个代码块之一来执行。

#### 2.3.1语法:

if (条件 1)

{

当条件 1 为 true 时执行的代码

}

else if (条件 2)

{

当条件 2 为 true 时执行的代码

}

else

{

当条件 1 和 条件 2 都不为 true 时执行的代码

}

例：如果时间小于 10:00，则将发送问候 "Good morning"，否则如果时间小于 20:00，则发送问候 "Good day"，否则发送问候 "Good evening"：

if (time<10)

{

x="Good morning";

}

else if (time<20)

{

x="Good day";

}

else

{

x="Good evening";

}

举例：

var ave=95;

if (ave >= 90){

document.write("A");

}

else if (ave >= 80){

document.write("B");

}

else if (ave >= 70) {

document.write("C");

}

else if (ave >= 60) {

document.write("D");

}

else {document.write("E");

}

## 2.4 switch 语句

### 2.4.1语法

switch 语句用于基于不同的条件来执行不同的动作。

switch(n)

{

case 1:

执行代码块 1

break;

case 2:

执行代码块 2

break;

default:

n 与 case 1 和 case 2 不同时执行的代码

}

switch(变量)  
  
{  
  
    case 值1:  
  
           代码1;  
  
           break;  
  
    case 值2:  
  
           代码2;  
  
           break;  
  
    case 值3:  
  
           代码3;  
  
           break;  
  
    default:  
  
           如果以上条件都不满足，则执行该代码;  
  
}

### 2.4.2switch结构说明：

**\*switch、case、break、default都是系统关键字，都必须全小写。**

**switch后的小括号()：小括号内一般是一个变量名，这个变量可能会有不同的取值。**

**每个case的值，与变量的值进行比对，如果一致就执行该case后的代码。**

**所有的case都是“或”的关系，每时每刻只有一个case会满足条件。**

**每个case中的代码执行完毕后，必须要用break语句结束，结束之后，程序将跳到switch结束大括号之后运行。**

**如果不写break语句的话，下面所有的case语句都会执行一遍。**

### 2.4.3if多条件判断和switch的区别

if中的条件一般是一个范围(a>100)，而switch中的条件一般是一个变量的不同取值。

### switch描述：根据一个变量的不同取值，来执行不同的代码。

## 2.5三元运算符:

如名字表示的三元运算符需要三个操作数。

语法是 条件 ? 结果1 : 结果2;. 这里你把条件写在问号(?)的前面后面跟着用冒号(:)分隔的结果1和结果2。满足条件时结果1否则结果2。

<script type="text/javascript">

var b=5;

(b == 5) ? a="true" : a="false";

document.write(" --------------------------- "+a);

</script>

结果： --------------------------- true

<script type="text/javascript">

var b=true;

(b == false) ? a="true" : a="false";

document.write(" --------------------------- "+a);

</script>

结果： --------------------------- false