**1.1输出**

**<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">**

**<html xmlns="http://www.w3.org/1999/xhtml">**

**<head>**

**<meta http-equiv="Content-Type" content="text/html;charset=UTF-8">**

**<title></title>**

**<body>**

<input type="button" value="关闭窗口" onclick="window.close();"/>

**<script type="text/javascript">**

**<!—**

**//输出方法**

document.write("您好，现在时间为："+new Date());

**-->**

**</script>**

**</body>**

**</html>**

**（注：以后只写<Script></ Script >内部的东西;**

**注释在代码右边，极少在上面，绝不在下面）**

**2.1变量的赋值**

var username = "Tom";**//声明变量username并赋值"Tom"**

var account = 4500; **//声明变量account并赋值4500**

age = 23; **//未声明直接为变量age赋值23**

var marriaged ; **//声明变量marriaged未赋值**

**2.2变量的练习**

var a = 3,b=5; **//声明全局变量a**

function add(x,y){ **//这里参数x和y为局部变量**

var c = 6; **//局部变量c**

return x + y;}

document.write("<h2>a + b =" + add(a,b)+ "<h2>");

document.write("c的值为：" + c);**//这里会发生错误，c的值未定义**

**2.3数据类型转换**

var a;　**//声明变量a，因为没有赋值此值为undefined**

var b = null;

var c = "100";

d = parseInt(c) + b; **//c由字符转为数值**

document.write("a转换为数值型的结果是：" + parseInt(a) +"<br/>");

**2.4数据类型转换**

var a = undefined;

var b = null;

var str1 = "",str2 = "hello123"; **//声明两个字符型变量，其值分别为空和"hello123"**

var num1 = 0,num2 = NaN,num3=38; **//声明三个数值型变量，其值分别为0，NaN和38**

var obj = new Object(); **//这里创建了一个对象，名称为obj**

with(document){ **//对象的with用法**

write("undefined转换后的值为：" + new Boolean(a) +"<br/>"); **//将a转为布尔型false**

write("null转换后的值为：" + new Boolean(b) +"<br/><hr>"); **//将b转为布尔型false**

write("空字符串转换后的值为：" + new Boolean(str1) +"<br/>"); **//转为布尔型false**

write("非空字符串转换后的值为：" + new Boolean(str2) +"<br/><hr>");**//转为布尔型true**

write("0转换后的值为：" + new Boolean(num1) +"<br/>");**//转为布尔型false**

write("NaN转换后的值为：" + new Boolean(num2) +"<br/>");**//转为布尔型false**

write("非0和NaN数值转换后的值为：" + new Boolean(num3) +"<br/><hr>");**//转为布尔型true**

write("对象转换后的值为:"+new Boolean(obj) +"<br/>"); **//转为布尔型true**

}

**3.1算术运算符练习**

var sum1 = 10,sum2 = 3;

result = "字符串" + sum1 + sum2;　**//结果为字符串**

with(document){

write("<p>sum1 = " + sum1 + " , sum2 = " + sum2 + "<p/>"); **//输出数值**

write("<li>sum1 + sum2 = " + (sum1 + sum2) + "</li>");　　**//输出数值（加减乘除道理一样）**

write("<li>sum1%sum2="+(sum1%sum2)+"</li>");**//求余**

write("<li>result = " + result + "</li>");

write("<li>++sum1="+ ++sum1+"</li>");**//自加**

write("<li>sum2--="+sum2--+ "</li>");**//自减**

}

**3.2关系运算符练习**

var sum1 = 10,sum2 = 3,str1 = "asdf",str2 = "aDfE";

document.write("<li>sum1 == sum2 = " + (sum1 == sum2) + "</li>"); **//返回false**

document.write("<li>str1 != str2 = " + (str1 != str2) + "</li>"); **//返回true**

document.write("<li>sum1 >= sum2 = " + (sum1 >= sum2) + "</li>"); **//返回true**

document.write("<li>str1 < str2　 = " + (str1 < str2) + "</li>");　　　**//返回false**

document.write('<li>sum1 === "10" = ' + (sum1 === "10") + "</li>"); **//返回false**

document.write('<li>str2 !== "aDfE"' + (str2 !== "aDfE") + "</li>"); **//返回false**

**3.3逻辑运算符练习**

var a = 1,b = 2,c = 3,d = 4;

document.write("<p>a = " + a + " , b = " + b + " ,c = " + c + " ,d = " +d +"<p/>");

document.write("<li>(a&gt;b)&&(c&lt;d) = " + ((a>b)&&(c<d)) + "</li>"); **//<用&lt;代替，>用&gt;代替**

document.write("<li>(a&gt;b)||(c&lt;d)2 = " + ((a>b)||(c<d)) + "</li>"); **//返回逻辑值**

document.write("<li>!(a&gt;b)&&(c&lt;d) = " + (!(a>b)&&(c<d)) + "</li>"); **//返回逻辑值**

document.write("<li>(a&gt;b)&&(++c&lt;d) = " + ((a>b)&&(++c<d)) + "</li>"); **//返回逻辑值**

**3.4?:运算符练习**

var username= "Bob";

result = (username=="")?"用户名可以使用":"用户名已经存在，请使用其他用户名"; **//条件？表达式１（条件真时执行）：表达式２（条件假时执行）**

alert(result);

**3.5逗号(,)运算符练习**

var a = 1,b = 2,c = 3; **//，相当于连接符**

result = (a,b,c);

alert(result);

**3.6void运算符练习**

<script language="javascript" type="text/javascript">

function check(){

var username= "";

result = (username==""||username==null)?"用户名可以使用":"用户名已经存在，请使用其他用户名";

alert(result);

}

</script>

<a href="javascript:void(check());">检查用户名</a>**//viod常用在URL中调用JavaScript**

**3.7typeof运算符练习**

var num = 5;

var str = "Glad to meet you!";

var bln = true;

var obj = new Object();

with(document){

write("<p>num = "+num + ",str = " + str + ",bln = " + bln + " ,obj为对象" + "<p/>");

write("<li>num的类型：" + typeof num + "</li>");

write("<li>str的类型：" + typeof str + "</li>");

write("<li>bln的类型：" + typeof bln + "</li>");

write("<li>obj的类型：" + typeof obj + "</li>");

}//**分别检测类型**

**4.1年龄验证**

var age = prompt("请输入您的年龄：",0);

if(age<18){

alert("您还是未成年人！");

}else{

alert("您已经是成年人了!");

}

**4.2成绩等级**

var score = prompt("请输入你的成绩：",0);

if(score<60){

alert("不及格");

}else if(score<75){

alert("及格");

}else if(score<85){

alert("良好");

}else{

alert("优秀");

}

**4.3成绩等级**

var grade = prompt("请输入您的成绩等级：","不及格");

switch(grade){

case "不及格":

alert("您的分数在60以下！");

break;

case "及格":

alert("您的分数在60－75之间！");

break;

case "良好":

alert("您的分数在75－85之间！");

break;

case "优秀":

　alert("您的分数在85－100之间！");

　break;

default:

alert("对不起，您的输入有误！");

}

**4.4while循环**

var n = 1;

while(n<=3){

document.write("<h" + n + ">while循环演示！</h" + n + ">");

n++;

}

**4.5do...while循环**

var a = 5;

do{

document.write("<h1>do...while循环演示！</h1>");

a++;

}while(a>8);

**4.6for循环**

var n = 1;**//用于保存计算结果**

for(var i = 8;i>0;i--){

n = n \* i;

}

document.write("<h2>8的阶乘为：" + n + "</h2>");

**4.7break语句**

var num = prompt("请输入您要计算的阶乘数：",1);

var n = 1,m = num;

while(true){//这是一个无限循环，如果没有break语句就不能跳出循环

if(m<1){

break;//如果m<1的话就跳出循环

}

n = n \* m;

m --;

}

document.write("<h2>" + num + "的阶乘为：" + n + "</h2>");

**4.8continue语句**

for(var i = 1; i <= 20; i++){

if(i % 3 ==0){**//如果能够被3整除**

continue;**//不执行下面的代码，直接进入下一次循环**

}

document.write( i + " ");

}

**4.9平年闰年判断**

while(true){

var year = prompt("请输入年份（输入0退出程序）：",0);

year = parseInt(year);**//将year转换为数值型**

if(year==0){

break;**//如果为0退出循环**

}

if(isNaN(year)){

alert("请输入合法年份！");

continue;**//进入下一次循环**

}

if(year % 4 == 0 && !year % 100 == 0 || year % 400 == 0){

alert(year + "年为闰年！");

}else{

alert(year + "年为平年！");

}

}

**5.1函数的定义和调用**

function print(msg){

document.write(msg);

}

print("Hello,I love function!");

**5.2函数的定义和调用**

function add(x,y){

z = x + y;

return z;

}

var a = 3 ,b = 4 ;

c = add(a,b);

alert("c的值为：" + c);

**5.3函数的定义和调用**

function max(x,y){

if(x>y){

return x;

}

return y;

}

var num = 3 ,str = "abc" ;

c = max(num,str);

alert("c的值为：" + c);

**5.4函数的定义和调用**

function print(x,y){

document.write("第1个参数为：" + x);

document.write(" 第2个参数为：" + y + "<br>");

}

var arg1 = "北京" ,arg2 = "上海",arg3 = "天津" ;

print(arg1);

print(arg1,arg2,arg3);

**5.5函数的定义和调用**

var a = 3,b = new Object();

b.name = "John";

function change1(x){

x++;

}

function change2(obj){

obj.name="Tom";

}

document.write("调用change1()前a的值为：" + a + "<br/>");

change1(a);

document.write("调用change1()后a的值为：" + a + "<hr>");

document.write("调用change2()前b.name的值为：" + b.name + "<br/>");

change2(b);

document.write("调用change2()后b.name的值为：" + b.name + "<br/>");

**5.6在表达式中定义函数**

function fnc(x){

return x\*x;

}

var squre = function(x){

return x\*x;

};

document.write("调用函数fnc(3)的结果：" + fnc(3));

document.write("<br/>调用函数squre(3)的结果：" + squre(3));

**5.7创建动态函数**

<script language="javascript" type="text/javascript">

　　var myFnc = new Function("x","y","var result = x \* y;return result;");

　　alert("myFnc(3,4)的结果为：" + myFnc(3,4));

**5.8函数嵌套**

function sum(x,y){

var result = squre(x) + squre(y);

function squre(z){

return z\*z;

}

return result;

}

alert("sum(3,4)的运行结果为：" + sum(3,4));

alert("squre(3)的运行结果为：" + squre(3));

**5.9递归函数**

function myFun(n){

if(n==1){

return 1

}

return n \* myFun(n - 1);

}

alert("5的阶乘为：" + myFun(5));

**5.10函数的arguments对象**

function printArgs(){

for(var i = 0 ;i<arguments.length;i++){**//arguments相当于对象自身**

document.write("arguments[" + i +"] = "+ arguments[i] + "<br/>");

}

document.write("<hr>");

}

printArgs("hello");

printArgs("welcome",2008);

printArgs(356,52,"great",31,"nice");

**5.11callee属性（arguments对象的）**

var fnc = function(n){

if(n==1){

return 1

}

return n \* **arguments.callee**(n - 1);

}**//callee保存了当前对象的引用**

alert("10的阶乘为：" + fnc(10));

**5.12作为数据的函数**

function myFun(x,y){

return x + y;

}

**abc = myFun;//直接将函数的返回值赋给变量**

alert("abc的类型为：" + typeof abc + "\nabc(2,3)的值为：" + abc(2,3));

**5.13escape()函数和unescape()函数的使用**

var str1 = "I love 北京 2008";

str2 = escape(str1);

alert("escape(" + str1 +")的值为：" + str2);

str3 = unescape(str2);

alert("unescape(" + str2 + ")的值为：" + str3);

**5.14encodeURI()函数和decodeURI()函数的使用**

var uri = "http://www.baidu.com/baidu?word=注册会计师";

enURI = encodeURI(uri);

alert("encode('" + uri +"')的值为：" + enURI);

deURI = decodeURI(enURI);

alert("decodeURI('" + enURI + "')的值为：" + deURI);

**5.15parseInt()函数和parseFloat()函数的使用**

document.write("parseInt('3')的返回值为：" + parseInt('3') + "<br>");**//强制类型转换**

**with（**document**）{**

write("parseInt('3.13')的返回值为：" + parseInt('3.14') + "<br>");

write("parseInt('345hello')的返回值为：" + parseInt('345hello') + "<br>");

write("parseInt('super2008')的返回值为：" + parseInt('super2008') + "<br>");

write("parseInt('0xdf73')的返回值为：" + parseInt('0xdf73') + "<br>");

write("parseInt('1001',2)的返回值为：" + parseInt('1001',2) + "<br>");

write("<hr>parseFloat('6')的返回值为：" + parseFloat('6') + "<br>");

write("parseFloat('2.56')的返回值为：" + parseFloat('2.56') + "<br>");

write("parseFloat('3.8E2')的返回值为：" + parseFloat('3.8E2') + "<br>");

write("parseFloat('0.36a')的返回值为：" + parseFloat('0.36a') + "<br>");

write("parseFloat('hao888')的返回值为：" + parseFloat('hao888') + "<br>");

write("parseFloat('.10')的返回值为：" + parseFloat('.10'));

**}**

**5.16isNaN()函数和isFinite()函数的使用**

var num1 = parseInt("2008love");

var num2 = parseInt("love2008");

var x = isNaN(num1)?'parseInt("2008love")是NaN':'parseInt("2008love")不是NaN';

document.write(x + "<br>");

var y = isNaN(num2)?'parseInt("love2008")是NaN':'parseInt("love2008")不是NaN';

document.write(y + "<br>");

var z = num2==num2?'parseInt("love2008")不是NaN':'parseInt("love2008")是NaN';

document.write(z + "<br>");

var m = isFinite(num2)? 'parseInt("love2008")是有限的':'parseInt("love2008")不是有限的';

document.write(m + "<br>");

var n = isFinite(47/3)?'47/3是有限的':'47/3不是有限的';

document.write(n);

**5.17eval()函数的使用**

function printMsg(str,i){

for(var m = 1 ; m <=i ; m++){

document.write(str + " ");

}

document.write("<br>");

}

for(var i = 1 ;i <= 10 ; i++){

eval("printMsg('^\_^'," + i + ")");//**eval()函数将字符串当代码执行**

}

**5.18进制转换器**

var str1 = prompt('请输入一个二进制数：',0);

var num1 = parseInt(str1,2);

if(isNaN(num1)){

document.write('对不起，您的输入有误，请重新输入！');

}else{

document.write(str1 , '<sub>2</sub> = ' , num1 , "<sub>10</sub><p/>");

}

var str2 = prompt('请输入一个十六进制数：',0);

var num2 = parseInt(str2,16);

if(isNaN(num2)){

document.write('对不起，您的输入有误，请重新输入！');

}else{

document.write(str2 , '<sub>16</sub> = ' , num2 , "<sub>10</sub><br>");

}

**6.1创建对象实例**

var date1 = new Date();

var date2 = new Date();

date1.setYear(2007);

date1.setMonth(10);

date1.setDate(1);

document.write(date1 + "<br>");

document.write(date2 + "<br>");

var date3 = new Date(2008,12,13,20,30,100);

document.write(date3);

**6.2构造函数和this关键字**

function Person(){

this.age = 18;

this.name = "张三";

this.sex = "男";

this.speak = function(){

document.write("Hi,I'm speaking..." + "<br>");

}

this.run = run;

}

function run(){

document.write("Hi,I'm running...");

}

var person = new Person();

document.write("name:" + person.name + "<br>");

document.write("sex:" + person.sex + "<br>");

document.write("age:" + person.age + "<br>");

person.speak();

person.run();

**6.3创建无类型对象**

var book = {

title:"JavaScript & Ajax 网页设计技术手册",

author:"Smartdong",//结尾使用逗号

price:"Unknown",

introduce:function(){

document.write("Hello,My name is " + this.title +

".<br>" + this.author + " is my author and my price is " +

this.price + ".<br>Enjoy yourself!")

}

}

book.introduce();

**6.4使用[]访问对象的属性和方法**

function Car(color,mileage,brand){

this.color = color;

this.mileage = mileage;

this.brand = brand;

this.start = function(){

document.write("start!start!" + "<br>");

}

this.stop = stop;

this.whistle = whistle;

}

this.stop = function(){

document.write("stop!stop!" + "<br>");

}

function whistle(){

document.write("di,di,di...");

}

var car1 = new Car("yellow",100,"BMW");

var car2 = new Car("black",200,"大奔");

document.write(car1["color"] + "<br>");

document.write(car1["mileage"] + "<br>");

document.write(car1["brand"] + "<br>");

car1["start"]();

document.write(car2["color"] + "<br>");

document.write(car2["mileage"] + "<br>");

document.write(car2["brand"] + "<br>");

car2["stop"]();

car2["whistle"]();

**6.5添加对象的属性和方法**

function Person(){

this.age = 18;

this.name = "张三";

this.sex = "男";

this.speak = function(){

document.write("Hi,I'm speaking..." + "<br>");

}

this.run = run;

}

function run(){

document.write("Hi,I'm running...");

}

var person = new Person();

person.height = 190;**//添加height属性**

person.jump = function(){**//添加jump方法**

document.write("Be careful!I'm jumping...!");

}

document.write(person.height + "<br>");

person.jump();**//调用新添加的jump方法**

var person2 = new Person();

person2.jump();**//person2中并没有添加jump方法，所以会出现错误提示**

**6.6重定义属性和方法**

function Person(){

this.age = 18;

this.name = "张三";

this.sex = "男";

this.speak = function(){

document.write("Hi,I'm speaking..." + "<br>");

}

this.run = run;

}

function run(){

document.write("Hi,I'm running...");

}

var person = new Person();

document.write("重定义前name的值为:" + person.name + "<br>");

person.name = "李四";

person.jump = function(){

document.write("sorry,my leg is so hurt!I can't jump");

}

document.write("重定义后name的值为:" + person.name + "<br>");

person.jump();

**6.7删除属性和方法**

function Person(){

this.age = 18;

this.name = "张三";

this.sex = "男";

this.speak = function(){

document.write("Hi,I'm speaking..." + "<br>");

}

this.run = run;

}

function run(){

document.write("Hi,I'm running...");

}

var person = new Person();

delete person.age;**//直接删除**

delete person.speak;

person.name = "undefined";**//设置为未定义**

document.write("name: " + person.name + "<br>");

document.write("age: " + person.age);

person.speak();

**6.8对象的废除**

var date = new Date();

date.setYear(2008);

date.setMonth(12);

date = null;**//设置为空**

**6.9with语句的使用**

var date = new Date();

with(date){

setFullYear(2008);

setMonth(11);

setDate(13);

setHours(3);

setMinutes(23);

setSeconds(35);

}

alert(date.toLocaleString());

**6.10for...in语句的使用**

function Plane(width,length,height){

this.width = width;

this.length = length;

this.height = height;

this.fly = function(){

document.write("I'm flying...");

}

}

var plane = new Plane(8,20,3);

function listProperties(obj){**//定义函数用于输出对象的所有属性**

var str = "";

for(var ppt in obj){

str += ppt + ":"+ obj[ppt] + "<br>";**//使用[]访问对象的属性**

}

document.write(str + "<hr>");

}

listProperties(plane);

delete plane.fly;

plane.width = undefined;

listProperties(plane);

**6.11Function对象的length属性**

function test(str1,str2,str3){

with(document){

write("arguments.length:" + arguments.length);

write(" arguments.callee.length:" + arguments.callee.length);

write("<br>");

}

}

test("123","456","789");

test("123","456");

test("123","456","789","000");

**6.12Function对象的call()方法和apply()方法**

function Person(name){

this.name = name;

this.sayHello = function(city){

document.write("Hello,my name is <b>" + this.name +

"</b>.I live in <b>" + city + "</b>.<br>");

}

}

var person1 = new Person("Tom");

var person2 = new Person("John");

person1.sayHello("BeiJing");

person2.sayHello("ShiJiazhuang");

person1.sayHello.**call(person2,"BeiJing")**;

person2.sayHello.call(person1,"ShiJiazhuang");

person1.sayHello.**apply(person2,["BeiJing"])**;

person2.sayHello.apply(person1,["ShiJiazhuang"]);

**6.13constructor属性（用来引用初始化该对象的构造函数）**

function Truck(width,height){

this.width = width;

this.height = height;

}

var truck = new Truck(2,2);

document.write("Truck的构造函数：" + truck.constructor + "<br>");

if(typeof truck == "object" && truck.constructor == Truck){**//判断truck是否为Truck的实例**

document.write("truck是Truck的一个实例");

}

**6.14利用Object()构造函数定义对象**

var person = new Object();

person.name = "李四";

person.age = 22;

person.sex = "女";

person.introduce = function(){

document.write("Hi,my name is " + this.name +

".I'm " + this.age + "years old.<br>");

}

**person.introduce();**

document.write("I'm a " + person.sex + ".");

**7.1实例属性和实例方法**

function Dog(){

//**实例属性**

this.name = "小白";

this.age = 3;

//**实例方法**

this.run = function(){

document.write("注意，我正在奔跑...<br>");

}

this.bark = bark;

}

function bark(){

document.write("汪汪汪...");

}

var dog1 = new Dog();

var dog2 = new Dog();

var dog3 = new Dog();

document.write("dog1.name: " + dog1.name + "<br>");

dog2.name = "小黑";

document.write("dog2.name: " + dog2.name + "<br>");

document.write("dog1.name: " + dog1.name + "<br>");

dog2.run();

dog3.bark();

**7.2类属性和类方法**

function Circle(radius){

this.radius = radius;

}

Circle.PI = 3.14;//**类属性**

Circle.calcPerimete = function(radius){**//类方法**

return 2\*Circle.PI\*radius;

}

Circle.calcArea = function(radius){ **//类方法**

return Circle.PI\*radius\*radius;

}

var radius1 = 4;

var radius2 = 9;

with(document){

write("半径为" + radius1 + "，周长为:" + Circle.calcPerimete(radius1) + "<br>");

write("半径为" + radius1 + "，面积为:" + Circle.calcArea(radius1) + "<br>");

write("半径为" + radius2 + "，周长为:" + Circle.calcPerimete(radius2) + "<br>");

write("半径为" + radius2 + "，面积为:" + Circle.calcArea(radius2) + "<br>");

}

7.3**私有属性和方法**

function Person(){

var age = 0;**//定义私有变量age**

function check(age){**//定义私有方法，用于检测age的合法性**

if(age<0 || age>150){

return false;

}

return true;

}

this.setAge = function(theAge){**//方法setAge,用于设置age的值**

if(check(theAge)){

age = theAge;

}else{

alert("你输入的年龄有误！");

}

}

this.getAge = function(){**//方法getAge，用于获取age的值**

return age;

}

}

var person = new Person();

person.setAge(25);

document.write("年龄：" + person.getAge());

person.setAge(151);

**7.3利用原型对象实现继承**

function Person(name,sex){

this.name = name;

this.sex = sex;

}

Person.prototype.sayHello = function(){

document.write("大家好，我叫" + this.name + "。<br>");

}

function Student(){

}

Student.prototype = new Person("张三","男");

var student = new Student();

Student.prototype.grade = "三";

Student.prototype.introduce = function(){

document.write("我是" + this.grade + "年级的学生。");

}

student.sayHello();

student.introduce();

**7.4通过构造函数调用实现继承**

function ClassA(name,**age**){

this.tempMethod = ClassB;

this.tempMethod(name);

delete this.tempMethod;**//删除**

this.age = **age**;

this.sayHello = function(){

document.write("你好，我的名字叫" + this.name + "。我" + this.age +"岁了。<br>");

}

}

function ClassB(name){

this.name = name;

this.sayHello = function(){

document.write("你好，我的名字叫" + this.name + "。<br>");

}

}

var classB = new ClassB("张三");

classB.sayHello();

var classA = new ClassA("李四",23);

classA.sayHello();

**7.5使用call()方法和apply()方法实现继承**

function Animal(color,age){

this.color = color;

this.age = age;

this.cry = function(){

document.write("Hi,I'm " + this.color + ".I'm " + this.age + " years old.<br>");

}

}

function Dog(color,age){

**Animal.call(this,color,age);//对象+参数**

}

function Cat(color,age){

**Animal.apply(this,[color,age]); //对象+参数**

}

var animal = new Animal("yellow",5);

animal.cry();

var dog = new Dog("black",3);

dog.cry();

var cat = new Cat("white",4);

cat.cry();

**8.1创建数组**

var array1 = new Array();

var array2 = new Array(10);

var array3 = new Array("str",345,true,new Object());

**var array4 = [2008,56,"意义",false];//推荐使用该方法**

with(document){

write("array1:" + array1 + "<br>");

write("array2:" + array2 + "<br>");

write("array3:" + array3 + "<br>");

write("array4:" + array4);

}

**8.2访问数组元素**

var array1 = new Array(4);**//创建一个长度为4的数组**

array1[0] = "你好";**//设置第一个数组元素的值为字符串“你好”**

array1[1] = 12345;**//设置第二个数组元素的值**

array1[2] = true;**//设置第三个数组元素的值**

var array2 = ["abc",789,false,new Date()];**//创建数组array2**

for(var i = 0 ;i < 4 ;i++){**//使用for循环输出array1的所有数组元素**

document.write("array1[" + i + "]: " + array1[i] + "<br>");

}

for(var i in array2){**//使用for...in语句输出array2的所有数组元素**

document.write("array2[" + i + "]: " + array2[i] + "<br>");

}

**8.3多维数组**

var array1 = new Array();**//创建空数组对象**

for(var i = 0 ; i < 3; i++){**//使用for循环向数组中添加5个元素**

array1[i] = new Array(123,'abc');**//每个元素又是一个数组对象**

}

var array2 = [['aaa','bbb','ccc'],[111,222,333]];**//通过使用嵌套的[]创建二维数组**

for(var i = 0 ; i < 3;i++){

for(var m = 0;m < 2; m++){

document.write("array1[" + i + "][" + m + "]:" + array1[i][m] + "<br>");

}

}

document.write("<hr>");

for(var i = 0 ; i < 2;i++){

for(var m = 0;m < 3; m++){

document.write("array2[" + i + "][" + m + "]:" + array2[i][m] + "<br>");

}

}

**8.4数组的length属性**

var array = new Array(123,456,'aaa','bbb','ccc');

**document.write("数组的长度：" + array.length + "<hr>");**

function listElements(array){**//遍历数组元素**

for(var i = 0; i < array.length; i ++){

document.write("array[" + i + "]：" + array[i] + "<br>");

if(i == array.length - 1){

document.write("<hr>");

}

}

}

array.length = 8;

listElements(array);

array.length = 3;

listElements(array);

</script>

**8.5合并数组**

var array1 = new Array(1,2,3,4);

var array2 = ['a','b','c','d'];

function list(array){

for(var i = 0; i < array.length; i ++){

document.write("array[" + i + "]:" + array[i] + "　");

if(i == array.length - 1){

document.write("<hr>");

}

}

}

list(array1);

list(array2);

**var array3 = array1.concat(array2);// 合并数组**

list(array3);

var array4 = array2.concat(array1);

list(array4);

**8.6push()方法和pop()方法（栈）**

var array = new Array('a');

var l = array.push('b','c','d');**//添加元素（可以多个）**

document.write("数组的长度：" + l + "<br>");

function list(array){

for(var i = 0; i < array.length; i ++){

document.write("array[" + i + "]:" + array[i] + "　");

if(i == array.length - 1){

document.write("<br>");

}

}

}

list(array);

var element = array.pop();**//删除元素并返回元素**

document.write("删除元素：" + element + "<br>");

document.write("数组的长度：" + array.length);

**8.7shift()方法和unshift()方法（队）**

var array = new Array();

for(var i = 1 ;i < 5 ;i ++){

array.unshift(i);

}

function list(array){

for(var i = 0; i < array.length; i ++){

document.write("array[" + i + "]:" + array[i] + "　");

if(i == array.length - 1){

document.write("<br>");

}

}

}

list(array);

array.unshift('a','b'); **//添加元素（可以多个）**

list(array);

var x = array.shift();**//删除元素并返回元素**

document.write("删除的元素：" + x + "<br>");

list(array);

**8.8splice(start，count，p1…pn)方法**

var array = new Array("x");

array.splice(0,0,1,2,3,4,5)**;//在第1个元素前插入三个元素'a'、'b'和'c'**

function list(array){**//定义函数list用于输出数组所有元素**

for(var i = 0; i < array.length; i ++){

document.write("array[" + i + "]:" + array[i] + "　");

if(i == array.length - 1){

document.write("<hr>");

}

}

}

document.write("插入数据后数组中元素：<br>" );

list(array);**//输出数组元素**

array.splice(2,3,"a","b","c");**//分别使用"a"、"b"和"c"替换3、4和5**

document.write("替换数据后数组中元素：<br>" );

list(array);

var m = array.splice(3,2);**//删除数组元素"b"和"c"**

document.write("删除数组元素：" + m + "<br>");**//输出删除的数组元素**

document.write("删除数据后数组中元素：<br>" );

list(array);

**8.9toString()方法和join()方法**

var array = [1,2,3,'a','b',true,false];

with(document){

write(array.toString() + "<br>");**//显式调用toString()方法**

write(array + "<br>");**//toString()方法会自动被调用**

write(array.join('⊙') + "<br>");**//使用⊙作为分隔符**

write(array.join(' ') + "<br>");**//使用空格作为分隔符**

write(array.join(','));**//使用逗号作为分隔符结果和toString()方法一样**

}

**8.10reverse()方法的使用**

var array = ["a","b","c","d","e","f"];

with(document){

write("输出反转前数组元素：" + array + "<br>");**//输出反转前数组元素的值**

write("输出反转后数组元素：" + array.reverse());/**/输出反转后数组元素的值**

}

**8.11slice(start，end)方法的使用**

var array = ["a","b","c","d","e","f"];

with(document){

write("数组元素：" + array + "<br>");**//输出截取前数组元素**

write("调用方法array.slice(2,3)：" + array.slice(2,3)+ "<br>");**//截取部分数组元素**

write("调用方法array.slice(5)：" + array.slice(3)+ "<br>");**//一直截取到数组末尾**

write("调用方法array.slice(-3,-4)：" + array.slice(-5,-3));**//等价于array.slice(1,3)**

}

var array = [222,11,3,4444];

with(document){

write("默认排序：" + array.sort() + "<br>");**//默认排序**

write("按规则排序：" + array.sort(function(a,b){return a - b}));**//按规则排序**

}

**8.12length属性**

var name = prompt("请输入你的姓名：","");

var password = prompt("请输入你的密码：","");

if(name.length <2){

document.write("姓名不得少于2个字符，而你的姓名只有" + name.length + "个字符。<br>");

}else{

document.write("恭喜你，你的姓名合法。<br>");

}

if(password.length < 6){

document.write("密码不得少于6位，而你的密码只有" + password.length + "位。<br>");

}else{

document.write("恭喜你，你的密码合法。<br>");

}

**8.13concat方法（string.concat（））**

var name = "张三";

var age = 23;

var sex = "男";

var str = "大家好，".concat("我叫" ，name，"，我今年" ，age，"岁，我是"，sex，"生。");

document.write(str);

**8.14toUpperCase方法和toLowerCase方法**

var str1 = "eBFzx57Dh";**//定义字符串变量str1**

var str2 = '12PQcemNHg';**//定义字符串变量str2**

with(document){**//使用with语句**

write("str1:" + str1 + "<br>");

write("str1.toLowerCase():" + str1.toLowerCase() + "<br>");**//将str1转换为小写**

write("str1.toUpperCase():" + str1.toUpperCase() + "<hr>");**//将str1转换为大写**

write("str2:" + str2 + "<br>");

write("str2.toUpperCase():" + str2.toUpperCase() + "<br>");**//将str2转换为大写**

write("str2.toUpperCase().toLowerCase():" + str2.toUpperCase().toLowerCase() + "<br>");/**/先将str2转换为大写再转换为小写**

}

**8.15charAt方法和charCodeAt方法（指定位置返回字符）**

var str = "夫君子之行，静以修身，俭以养德。非淡泊无以明志，"

+ "非宁静无以致远。夫学须静也，才须学也，非学无以广才，非志无以成学。"

+ "淫慢则不能励精，险躁则不能冶性。年与时驰，意与日去，遂成枯落，"

+ "多不接世，悲守穷庐，将复何及！";

with(document){

write("字符串的长度：" + str.length + "<br>");

write("str.charAt(7)：" + str.charAt(7) + "<br>");

write("str.charCodeAt(7)：" + str.charCodeAt(7) + "<br>");

write("str.charAt(25)：" + str.charAt(25) + "<br>");

write("str.charAt(25)：" + str.charCodeAt(25) + "<br>");

write("str.charAt()：" + str.charAt() + "<br>");**//空返回首字符**

write("str.charAt(103)：" + str.charAt(103) + "<br>");**//超出长度返回空**

}

**8.16indexOf（“str”，start）方法和lastIndexOf（“str”，start）方法（指定字符返回位置）**

var str = "abcdefghijklmnedcba";

with(document){

write("字符串的长度：" + str.length + "<br>");

write("str.indexOf('e')：" + str.indexOf('e') + "<br>");

write("str.indexOf('e',-5)：" + str.indexOf('e',-5) + "<br>");**//起始位置省略或为负，从头开始**

write("str.indexOf('e',7)：" + str.indexOf('e',7) + "<br>");**//“7”表示起始位置**

write("str.indexOf('bcd')：" + str.indexOf('bcd') + "<br>");

write("str.indexOf('o')：" + str.indexOf('o') + "<br>");

write("str.lastIndexOf('e')：" + str.lastIndexOf('e') + "<br>");

write("str.lastIndexOf('e',10)：" + str.lastIndexOf('e',10) + "<br>");

}

**8.17split（分隔符，分割项数）方法**

var str = "北京，天津，上海，石家庄，唐山";

function list(array){

for(var i = 0; i < array.length; i ++){

document.write("array[" + i + "]:" + array[i] + "　");

if(i == array.length - 1){

document.write("<hr>");

}

}

}

array1 = str.split("，");

list(array1);

array2 = str.split("，",3);**//舍去后面两项**

list(array2);

array3 = str.split("");

list(array3);

array4 = str.split();

list(array4);

**8.18slice（start，end）方法、subString（start，end）方法和subStr（start，length）方法**

var str = "abcdefgh";

with(document){

write("str.slice(3)：" + str.slice(3) + "<br>");

write("str.slice(3,5)：" + str.slice(3,5) + "<br>");

write("str.slice(-5,-1)：" + str.slice(-5,-1) + "<br>");

write("str.substring(3)：" + str.substring(3) + "<br>");

write("str.substring(3,5)：" + str.substring(3,5) + "<br>");

write("str.substring(-5,-1)：" + str.substring(-5,-1) + "<br>");

write("str.substring(5,3)：" + str.substring(5,3) + "<br>");

write("str.substr(3)：" + str.substr(3) + "<br>");

write("str.substr(3,5)：" + str.substr(3,5) + "<br>");

write("str.substr(3,0)：" + str.substr(3,0) + "<br>");

}

**8.19localeCompare方法**

var string1 = "股票";

var string2 = "基金";

var string3 = "期货";

function compare(str1,str2){

var r = **str1.localeCompare(str2);**

switch(r){**//比较返回值只有1，-1，0**

case -1:

r = "小于";

break;

case 0:

r = "等于";

break;

case 1:

r = "大于";

break;

}

document.write(str1 + " <b>" + r + "</b> "+ str2 + "<br>");

}

compare(string1,string2);

compare(string2,string1);

compare(string2,'基金');

compare(string3,string1);

**8.20fromCharCode方法**

var str = "我爱北京天安门";**//定义字符串str**

var array = new Array(str.length);**//创建一个和str长度相同的数组array**

for(var i = 0;i < str.length; i ++){

array[i] = str.charCodeAt(i);**//将每个字符的Unicode编码依次存储到array数组中**

document.write(str.charAt(i) + ":" + str.charCodeAt(i) + " ");**//输出每个字符对应的Unicode编码**

}

document.write("<hr>");**//输出下画线**

for(var i = array.length - 3;i < array.length; i ++){

document.write(String.fromCharCode(array[i]));**//根据array数组中后三个元素表示的字符的Unicode编码生成新的字符串**

}

**8.21修饰字符串方法**

with(document){

write("测试字符串.anchor(chapter1)".anchor("chapter1"),'<p/>'); write("测试字符串.big()".big(),'<p/>');

write("测试字符串.blink()".blink(),'<p/>');

write("测试字符串.bold()".bold(),'<p/>');

write("测试字符串.fixed()".fixed(),'<p/>');

write("测试字符串.fontcolor(blue)".fontcolor("blue"),'<p/>');

write("测试字符串.fontcolor(DDDCCC)".fontcolor("gainsboro"),'<p/>');

write("测试字符串.fontsize(20)".fontsize(20),'<p/>');

write("测试字符串.italics()".italics(),'<p/>');

write("测试字符串.link(http://www.sina.com)". link("http://www.sina.com"),'<p/>');

write("测试字符串.small()".small(),'<p/>');

write("测试字符串.strike()".strike(),'<p/>');

write("测试字符串.sub()".sub(),'<p/>');

write("测试字符串.sup()".sup(),'<p/>');

}

等价于：

<a name="chapter1">测试字符.anchor(chapter1)</a>

<p><big>测试字符串.big()</big></p>

<p><blink>测试字符串.blink()</blink></p>

<p><b>测试字符串.bold()</b></p>

<p><tt>测试字符串.fixed()</tt></p>

<p><font color="blue">测试字符串. fontcolor(blue) </font></p>

<p><font color="gainsboro">测试字符串. Fontcolor (DDDCCC)</font></p>

<p><font size="20">测试字符串.fontsize(20) </font> </p>

<p><i>测试字符串.italics()</i></p>

<p><a href="#">测试字符串.link(<http://www.sina.com>) </a></p>

<p><small>测试字符串.small()</small></p>

<p><strike>测试字符串.strike()</strike></p>

<p><sub>测试字符串.sub()</sub></p>

<p><sup>测试字符串.sup()</sup></p>

**9.1Math对象的属性**

with(document){

write("Math.E = " + Math.E + "<br>");

write("Math.LN2 = " + Math.LN2 + "<br>");

write("Math.LN10 = " + Math.LN10 + "<br>");

write("Math.LOG2E = " + Math.LOG2E + "<br>");

write("Math.LOG10E = " + Math.LOG10E + "<br>");

write("Math.PI = " + Math.PI + "<br>");

write("Math.SQRT1\_2 = " + Math.SQRT1\_2 + "<br>");

write("Math.SQRT2 = " + Math.SQRT2 + "<br>");

}

**9.2abs方法**

var a = -3,b = 0,c = 5;**//定义三个数值型变量**

**//定义函数check用于查检x是负数还是非负数**

function check(x){

var result = "";

if(x==Math.abs(x)){

result = "为非负数";

}else{

result = "为负数";

}

document.write(x + result + "<br>");

} **//输出a，b，c的绝对值**

with(document){

write(a + "的绝对值为：" + Math.abs(a) + "<br>");

write(b + "的绝对值为：" + Math.abs(b) + "<br>");

write(c + "的绝对值为：" + Math.abs(c) + "<br>");

} **//查检a，b，c是否为负数**

check(a);

check(b);

check(c);

**9.3计算三角函数的方法**

var a = -0.8,b = 0,c = 0.5,d = 1,e = 5;

with(document){

with(Math){

write("sin(" + a +")：" + sin(a) + "<br>");

write("sin(" + b +")：" + sin(b) + "<br>");

write("asin(" + c +")：" + asin(c) + "<br>");

write("asin(sin(" + d +"))：" + asin(sin(d)) + "<br>");

write("cos(" + b +")：" + cos(b) + "<br>");

write("cos(" + c +")：" + cos(c) + "<br>");

write("acos(" + e +")：" + acos(e) + "<br>");

write("acos(cos(" + b +"))：" + acos(cos(b)) + "<br>");

write("tan(" + a +")：" + tan(a) + "<br>");

write("tan(" + e +")：" + tan(e) + "<br>");

write("atan(" + c +")：" + atan(c) + "<br>");

write("atan2(" + a + "," + c +")：" + atan2(a,c) + "<br>");

write("atan2(" + d + "," + e +")：" + atan2(d,e) + "<br>");

write("atan(" + d + "/" + e +")：" + atan(d/e) + "<br>");

}

}

**9.4数值舍入运算的方法**

var a = 3.5,b = -3.5,c = 3,d = 3.49,e = -3.49;

with(document){

**//使用ceil函数进行舍入**

write("Math.ceil(" + a +")：" + Math.ceil(a) + "<br>");

write("Math.ceil(" + b +")：" + Math.ceil(b) + "<br>");

write("Math.ceil(" + c +")：" + Math.ceil(c) + "<br>");

**//使用floor函数进行舍入**

write("Math.floor(" + a +")：" + Math.floor(a) + "<br>");

write("Math.floor(" + b +")：" + Math.floor(b) + "<br>");

write("Math.floor(" + c +")：" + Math.floor(c) + "<br>");

**//使用round函数进行四舍五入**

write("Math.round(" + a +")：" + Math.round(a) + "<br>");

write("Math.round(" + b +")：" + Math.round(b) + "<br>");

write("Math.round(" + c +")：" + Math.round(c) + "<br>");

write("Math.round(" + d +")：" + Math.round(d) + "<br>");

write("Math.round(" + e +")：" + Math.round(e) + "<br>");

}

**9.5random方法**

var m = parseInt(prompt("请输入随机数的下限：",0));

var n = parseInt(prompt("请输入随机数的上限：",0));

document.write(m + "到" + n + "之间的十个随机数为：<br>" );

for(var i = 0; i < 10; i++){

var result = Math.round(Math.random()\*(n - m)) + m;

document.write(result + "，");

}

**9.6max方法和min方法(组合使用限定取值范围)**

var a = -5,b = 0 ,c = 15;

document.write(a + "，" + b + "，" + c + "中最大的数为：" + Math.max(a,b,c) + "<br>");

document.write(a + "，" + b + "，" + c + "中最小的数为：" + Math.min(a,b,c));

**9.7pow方法和exp方法**

with(document){

**//使用pow方法进行幂运算**

write("Math.pow(2,3)：" + Math.pow(2,3) + "<br>");

write("Math.pow(0.5,-2)：" + Math.pow(0.5,-2) + "<br>");

write("Math.pow(5,0.3)：" + Math.pow(5,0.3) + "<br>");

**//使用exp方法进行幂运算**

write("Math.exp(5)：" + Math.exp(5) + "<br>");

write("Math.exp(0)：" + Math.exp(0) + "<br>");

write("Math.exp(-3)：" + Math.exp(-3) + "<br>");

}

**9.8log方法和sqrt方法**

with(document){

**//使用log方法计算自然对数**

write("Math.log(0)：" + Math.log(0) + "<br>");

write("Math.log(1)：" + Math.log(1) + "<br>");

write("Math.log(100)：" + Math.log(100) + "<br>");

write("Math.log(Math.exp(50))：" + Math.log(Math.exp(50)) + "<br>");

**//使用sqrt方法计算平方根**

write("Math.sqrt(-3)：" + Math.sqrt(-3) + "<br>");

write("Math.sqrt(0)：" + Math.sqrt(0) + "<br>");

write("Math.sqrt(256)：" + Math.sqrt(256) + "<br>");

}

**9.9Date对象获取日期时间信息的方法**

var date1 = new Date("January 1,1998 00:25:18");

var date2 = new Date();**//获取表示当前时间的Date对象**

var date3 = new Date(2007,0,31);

var date4 = new Date(523004958372);

with(document){ **//从date1中获取日期时间信息**

write("date1：" + date1.toLocaleString() + "<br>");

write("date1.getFullYear()：" + date1.getFullYear() + "<br>"); **//这里要注意UTC时间和本地时间的差别**

write("date1.getUTCFullYear()：" + date1.getUTCFullYear() + "<br>");

write("date1.getYear()：" + date1.getYear() + "<hr>"); **//从date2中获取日期时间信息**

write("date2：" + date2.toLocaleString() + "<br>");

write("date2.getMonth()：" + date2.getMonth() + "<br>");

write("date2.getDate()：" + date2.getDate() + "<br>");

write("date2.getUTCDate()：" + date2.getUTCDate() + "<hr>"); **//从date3中获取日期时间信息**

write("date3：" + date3.toLocaleString() + "<br>");

write("date3.getDay()：" + date3.getDay() + "<br>");

write("date3.getUTCDay()：" + date3.getUTCDay() + "<br>");

write("date3.getHours()：" + date3.getHours() + "<hr>");

**//从date4中获取日期时间信息**

write("date4：" + date4.toLocaleString() + "<br>");

write("date4.getMinutes()：" + date4.getMinutes() + "<br>");

write("date4.getTime()：" + date4.getTime() + "<br>");

write("date4.getTimezoneOffset()：" + date4.getTimezoneOffset() + "<hr>");

}

**9.10Date对象设置日期时间值的方法**

var date = new Date();**//创建表示当前时间的Date对象**

document.write(date.toLocaleString() + "<br>");**//将date对象转换为字符串的形式**

**//设置date所表示时间的各部分的值**

with(date){

date.setFullYear(2008);

date.setMonth(11);

date.setDate(13);

date.setHours(12);

date.setMinutes(34);

date.setSeconds(56);

}

document.write(date.toLocaleString() + "<br>");

date.setTime(4238498293894);

document.write(date.toLocaleString());

**9.11Date对象将日期时间转换为字符串的方法**

var date = new Date();**//创建表示当前时间的Date对象**

with(document){

write("date.toDateString()：" + date.toDateString() + "<br>");

write("date.toUTCString()：" + date.toUTCString() + "<br>");

write("date.toGMTString()：" + date.toGMTString() + "<br>");

write("date.toLocaleString()：" + date.toLocaleString() + "<br>");**//常用**

write("date.toLocaleDateString()：" + date.toLocaleDateString() + "<br>");

write("date.toTimeString()：" + date.toTimeString() + "<br>");

write("date.toLocaleTimeString()：" + date.toLocaleTimeString() + "<br>");

write("date.toString()：" + date.toString() + "<br>");

}

**9.12Date对象parse方法、UTC方法和valueOf方法**

var strDate = "December 25,2005";

var ms = Date.parse(strDate);

document.write(strDate + " 距离GMT时间有" + ms + "毫秒<br>");**//与1970-1-1午夜时间相差毫秒数**

document.write("Date.UTC(2007,11,23,34,23,46)：" + Date.UTC(2007,11,23,34,23,46) + "<br>");

var date = new Date();

document.write("date.valueOf()：" + date.valueOf() + "<br>");

document.write("date.getTime()：" + date.getTime());

#### 9.13随时间变化的提示

var today = new Date();

var day;

var date;

if(today.getDay()==0){

day = "星期日";

}

if(today.getDay()==1){

day = "星期一";

}

if(today.getDay()==2){

day = "星期二";

}

if(today.getDay()==3){

day = "星期三";

}

if(today.getDay()==4){

day = "星期四";

}

if(today.getDay()==5){

day = "星期五";

}

if(today.getDay()==6){

day = "星期六";

}

date1 = "<font size=4 color=red>" + (today.getYear()) + "年" + (today.getMonth() + 1 ) + "月" + today.getDate() + "日 " + "</font>";

date2 = "<font size=4 color=red>" + day + "</font>";

document.write("<center>" + date1 + date2 + "</center>");

document.write("<center><font color=#0066ff size=+1> ");

day = new Date();

hr = day.getHours();

if (hr == 1){

document.write("已经凌晨一点了，快休息吧！");

}

if (hr == 3){

document.write("夜深人静，这个时候你会想起谁？？？");

}

if (hr == 4){

document.write("四点过了，还不没睡呐？？？");

}

if (hr ==6){

document.write("该起床了！");

}

if (hr ==7){

document.write("吃早饭喽！")

}

if ((hr ==8) || (hr == 9) || (hr ==10) || hr ==11){

document.write("上午好，工作愉快！");

}

if (hr ==12){

document.write("十二点过了，还不下班？");

}

if ((hr==13) || (hr==14)){

document.write("你不睡午觉？");

}

if ((hr==15) || (hr==16) || (hr==17)){

document.write("下午好,打起精神了，不要打盹！");

}

if ((hr==18) || (hr==19)){

document.write("吃晚饭了吗？");

}

if ((hr==20) || (hr==21) || (hr==22)){

document.write("又在玩电脑，晚上的宝贵时间可不要浪费啊！");

}

if (hr==23){

document.write("真有精神啊！");

}

if (hr==0){

document.write("凌晨了，还不睡？");

}

document.write("</font></center>");

**10.1Number对象的使用**

**//定义一个数值型变量并初始化一个值**

var num = 9876.5432123456789;

with(document){

**//输出Number对象提供的静态常量的值**

write("Number.MAX\_VALUE = " + Number.MAX\_VALUE + "<br>");

write("Number.MIN\_VALUE = " + Number.MIN\_VALUE + "<br>");

write("Number.NEGATIVE\_INFINITY = " + Number.NEGATIVE\_INFINITY + "<br>");

write("Number.POSITIVE\_INFINITY = " + Number.POSITIVE\_INFINITY + "<br>");

**//调用toString方法**

write("num.toString() = " + num.toString() + "<br>");

write("num.toString(8) = " + num.toString(8) + "<br>");

**//调用toLocaleString方法**

write("num.toLocaleString() = " + num.toLocaleString() + "<br>");

**//调用toFixed方法**

write("num.toFixed(3) = " + num.toFixed(3) + "<br>")**;//指定小数位为三**

**write("num.toFixed() = " + num.toFixed() + "<br>");**

**//调用toExponential方法**

write("num.toExponential(2) = " + num.toExponential(2) + "<br>");

write("num.toExponential() = " + num.toExponential() + "<br>");

**//调用toPrecision方法**

write("num.toPrecision(10) = " + num.toPrecision(10) + "<br>");

write("num.toPrecision(3) = " + num.toPrecision(3) + "<br>");

write("num.toPrecision() = " + num.toPrecision() + "<br>");

**//调用valueOf方法**

write("num.valueOf() = " + num.valueOf() + "<br>");

}

**10.2try...catch...finally的使用**

try{

var date = new Date();

date.test();

document.write("try块执行结束<br>");

}catch(error){

with(document){

write("出现了异常<br>");

write("异常类型：" + error.name + "<br>");

write("异常消息："+ error.message);

}

}finally{

document.write("异常处理完毕！");

}

**10.3throw语句的使用**

function testEval(exp){

try{

result = eval(exp);

}catch(error){

throw new Error("对不起，您输出的表达式不合法！");

}

document.write("表达式的值为：" + result + "<br>");

}

var isOk = false;

while(!isOk){

var x = prompt("请输入一个表达式：","");

try{

testEval(x);

isOk = true;

}catch(error){

document.write(error.message + "<br>");

}

}

**11.1RegExp对象的使用**

var str = prompt("请输入要匹配的字符串：","");

if(str.length>0){

document.write(matchStr(str));

}

function matchStr(str){

var s;

var re = new RegExp("d(b+)(d)","ig");

var arr = re.exec(str);

s = "共找到" + arr.length + "个匹配的结果！<br/>";

s += "$1的值为：" + RegExp.$1 + "<br/>";

s += "$2的值为：" + RegExp.$2 + "<br/>";

s += "$3的值为：" + RegExp.$3 + "<br/>";

s += "input的值为：" + RegExp.input + "<br/>";

s += "lastMatch的值为：" + RegExp.lastMatch + "<br/>";

s += "leftContext的值为：" + RegExp.leftContext + "<br/>";

s += "rightContext的值为：" + RegExp.rightContext + "<br/>";

s += "lastParen的值为：" + RegExp.lastParen + "<br/>";

return(s);

}

**11.2验证字符串格式**

var email = prompt("请输入符合“字符#字符%字符”格式的字符串：","");

if(email!=null && email.length>0){

checkEmail(email);

}

function checkEmail(email){

var re = /^\S+#\S+\%\S+$/;

if(re.test(email)){

alert("恭喜，你输入的字符串符合标准！");

}else{

alert("对不起，你输入的字符串不符合标准！");

}

}

11.3search()方法和replace()方法

<body>

<script type="text/javascript">

<!--

var s = "compile()方法用于编译一个正则表达式对象。\n"

+ "exec()方法用于运行正则表达式匹配。\n"

+ "test()方法用于测试正则达式匹配";

document.write(s + "<br/>");

function searchText(){

var re = /\(\)/;

var pos = s.search(re);

if(pos!=-1){

alert("找到位于索引" + pos + "处。");

}else{

alert("对不起，没有找到！");

}

}

function replaceText(){

var re = /[A-Za-z]+/g;

var s2 = s.replace(re,"XXXXX");

alert("替换后的字符串为：\n" + s2);

}

-->

</script>

<p><a href="javascript:searchText()">查找</a> <a href="javascript:replaceText()">替换</a> </p>

</body>

**11.4String对象的match()方法**

**<body>**

<script type="text/javascript">

<!--

var s = "The rain in Spain falls mainly in the plain";

document.write(s + "<br>");

function matchText1(){

var r, re;

re = /ain/i;

r = s.match(re);

alert("匹配的字符为：" + r);

}

function matchText2(){

var r, re;

re = /ain/ig;

r = s.match(re);

alert("匹配的字符为：" + r);

}

-->

</script>

<p><a href="javascript:matchText1()">非全局匹配</a> <a href="javascript:matchText2()">全局匹配</a> </p>

</body>

**11.5验证电子邮件地址**

var email = prompt("请输入E-mail地址：","");

if(email!=null && email.length>0){

checkEmail(email);

}

function checkEmail(email){

var re = /^\S+#\S+\%\S+$/;

if(re.test(email)){

alert("恭喜，你输入的E-mail合法！");

}else{

alert("对不起，你输入的E-mail有误！");

}

}

**11.6验证MasterCard卡号**

var card = prompt("请输入MasterCard：","");

if(card!=null && card.length>0){

checkCard(card);

}

function checkCard(card){

var re = /^5[1-5]\d{2}[\s\-]?\d{4}[\s\-]?\d{4}[\s\-]?\d{4}$/;

if(re.test(card)){

alert("恭喜，你输入的MasterCard卡号合法！");

}else{

alert("对不起，你输入的MasterCard卡号有误！");

}

}

**11.7验证身份证号码**

<script type="text/javascript">

//建立存在地址码及地址名称的数组

var aCity={11:"北京",12:"天津",13:"河北",14:"山西",15:"内蒙古"

　　　　　　,21:"辽宁",22:"吉林",23:"黑龙江 ",31:"上海",32:"江苏"

　　　　　　,33:"浙江",34:"安徽",35:"福建",36:"江西",37:"山东",41:"河南"

　　　　　　,42:"湖北 ",43:"湖南",44:"广东",45:"广西",46:"海南",50:"重庆"

　　　　　　,51:"四川",52:"贵州",53:"云南",54:"西藏 ",61:"陕西",62:"甘肃"

　　　　　　,63:"青海",64:"宁夏",65:"新疆",71:"台湾",81:"香港",82:"澳门",91:"国外 "

　　　　　　}

function checkCard(cardId){

var iSum = 0;

var re = /^\d{17}(\d|X)$/i;

if(!re.test(cardId)){

return false;

}

cardId = cardId.replace(/X$/i,"a");

**//验证地址码是否合法**

if(aCity[parseInt(cardId.substr(0,2))] == null){

alert('地址非法！');

return false;

}

**//验证出生日期码是否合法**

sBirthday = cardId.substr(6,4)+"-"+Number(cardId.substr(10,2))+"-"+Number(cardId.substr(12,2));

var date = new Date(sBirthday.replace(/-/g,"/"));

if(sBirthday!=(date.getFullYear()+"-"+ (date.getMonth()+1) + "-" + date.getDate())){

alert('出生日期非法！');

return false;

}

**//验证检验码是否合法**

for(var i = 17;i>=0;i--){

iSum += (Math.pow(2,i) % 11) \* parseInt(cardId.charAt(17 - i),11);

}

if(iSum%11!=1){

alert("校验码非法！");

return false;

}

alert('地址：'+aCity[parseInt(cardId.substr(0,2))] + "\n"

+ "出生日期：" +sBirthday + "\n"

+ "性别：" + (cardId.substr(16,1)%2?"男":"女"));

}

</script>

</head>

<body>

<form id="form1" name="form1" method="post" action="">

<label>请输入身份证号码：<br />

<br />

<input name="card" type="text" id="card" maxlength="19" />

</label>

<p>

<label>

<input type="button" name="Submit" value="确定" onclick='checkCard(this.form.card.value)'/>

</label>

<label>

<input type="reset" name="Submit2" value="重置" />

</label>

</p>

</form>

</body>

**11.8验证IP地址**

<script type="text/javascript">

function checkIP(){

var ip = document.getElementById('ip').value;

var re = /^(\d{1,2}|1\d\d|2[0-4]\d|25[0-5])\.(\d{1,2}|1\d\d|2[0-4]\d|25[0-5])\.(\d{1,2}|1\d\d|2[0-4]\d|25[0-5])\.(\d{1,2}|1\d\d|2[0-4]\d|25[0-5])$/;

if(re.test(ip)){

alert("恭喜，你输入的IP地址有效！");

}else{

alert("对不起，你输入的IP地址无效！");

}

}

</script>

</head>

<body>

<form id="form1" name="form1" method="post" action="">

<label>请输入IP地址：<br />

<br />

<input name="ip" type="text" id="ip" maxlength="19" />

</label>

<p>

<label>

<input type="button" name="Submit" value="验证" onclick='checkIP()'/>

</label>

<label>

<input type="reset" name="Submit2" value="重置" />

</label>

</p>

</form>

</body>

#### 12.1窗口的打开与关闭

<script language="javascript" type="text/javascript">

var subWin;

function openWin(){

subWin = window.open("13-1-2.html","new","status,menubar");

}

function closeSubWin(){

if(subWin && !subWin.closed){

subWin.close();}

}

</script>

<body>

<h2>父窗口</h2>

<form action="" method="post" enctype="multipart/form-data" name="form1" id="form1">

<label>

<input type="button" name="Submit" value="打开窗口" onclick="openWin()"/>

</label>

<label>

<input type="button" name="Submit2" value="关闭子窗口" onclick="closeSubWin()"/>

</label>

<label>

<input type="button" name="Submit3" value="关闭当前窗口" onclick="window.close()"/>

</label>

</form>

</body>

<script language="javascript" type="text/javascript">

function closeWin(){

if(window.opener && !opener.closed){

opener.close();

}

}

</script>

<body>

<h2>子窗口</h2>

<form action="" method="post" enctype="multipart/form-data" name="form1" id="form1">

<label>

<input type="button" name="Submit3" value="关闭当前窗口" onclick="window.close()"/>

</label>

<label>

<input name="Submit" type="button" value="关闭父窗口" onclick="closeWin()"/>

</label>

</form>

</body>

#### 12.2改变窗口位置与大小

function zoomBy(){

byWidth = document.forms[0].byWidth.value;

byHeight = document.forms[0].byHeight.value;

window.resizeBy(byWidth,byHeight);

}

function zoomTo(){

toWidth = document.forms[0].toWidth.value;

toHeight = document.forms[0].toHeight.value;

window.resizeTo(toWidth,toHeight);

}

function move(){

x = document.forms[1].x.value;

y = document.forms[1].y.value;

window.moveTo(x,y);

}

</script>

<body>

<h2>改变窗口大小</h2>

<form>

<p>窗口缩放 宽：

<label>

<input id="byWidth" name="textfield5" type="text" size="5" />

</label>

高：

<label>

<input id="byHeight" name="textfield6" type="text" size="5" />

</label>

<label>

<input type="button" name="Submit4" value="确定" onclick="zoomBy()"/>

</label>

</p>

<p>窗口大小 宽：

<label>

<input id="toWidth" name="textfield" type="text" size="5" />

</label>

高：

<label>

<input id="toHeight" name="textfield2" type="text" size="5" />

</label>

<label>

<input type="button" name="Submit" value="确定" onclick="zoomTo()"/>

</label>

</p>

</form>

<hr />

<h2>改变窗口位置</h2>

<form id="form1" name="form1" method="post" action="">

<label>

<input type="button" name="Submit2" value="左移" onclick="window.moveBy(-10)"/>

</label>

<label>

<input type="button" name="Submit5" value="右移" onclick="window.moveBy(10)"/>

</label>

<label>

<input type="button" name="Submit6" value="上移" onclick="window.moveBy(0,-10)"/>

</label>

<label>

<input type="button" name="Submit7" value="下移" onclick="window.moveBy(0,10)"/>

</label>

<p>窗口位置　X：

<label>

<input id="x" name="textfield3" type="text" size="5" />

</label>

Y：

<label>

<input id="y" name="textfield4" type="text" size="5" />

</label>

<label>

<input type="button" name="Submit3" value="确定" onclick="move()"/>

</label>

</p>

</form>

<hr />

<h2>屏幕信息</h2>

<script language="javascript" type="text/javascript">

with(document){

write("屏幕分辨率：" + screen.width + " X " + screen.height + "<br>");

write("屏幕颜色深度：" + screen.colorDepth);

}

</script>

**12.3系统对话框**

var name = prompt("请输入您的姓名：","");

if(confirm("你确认已经成年了吗？")){

alert("欢迎" + name + "光临！");

}else{

alert("未成年人请先离开！");

}

**12.4状态栏文字**

<body>

<h2>设置状态栏上的文字</h2>

<a href='http://www.cicpa.org.cn/'>中国注册会计师协会</a>

<script language="javascript" type="text/javascript">

<!--

var str = prompt('请输入要在状态栏上显示的文字：','');

if(str!=null){

window.defaultStatus = str;

}

// -->

</script>

</body>

**12.5定时器的使用**

<script language="javascript" type="text/javascript">

var timer;

function start(){

date = new Date().toLocaleString();

window.status=date;

}

</script>

<body>

<form id="form1" name="form1" method="post" action="">

<label>

<input type="button" name="Submit" value="显示时钟" onclick="timer=setInterval('start()',1000)"/>

</label>

<label>

<input type="submit" name="Submit2" value="停止显示" onclick="clearInterval(timer)"/>

</label>

</form>

</body>

**12.6状态栏滚动文字**

<script language="javascript" type="text/javascript">

<!--

**// 定义要滚动的字符串**

var scrtxt = "欢迎来到JavaScript的精彩世界...";

**// 获取滚动显示的字符串的长度**

var length = scrtxt.length;

**// 定义状态栏中用于显示字符串的空间的宽度**

var width = 100;

**// 定义起始置为-102**

var pos = -(width +2 );

function scroll() {

pos++;

var scroller = "";

**//如果位置参数已经等于字符串长度则回到初始位置**

if (pos == length) {

pos = -(width + 2);

}

**//如果字符串还没有显示完整**

if (pos < 0) {

**//用循环在显示的部分字符前加若干空格**

for (var i = 1; i <= Math.abs(pos); i++) {

scroller = scroller + " ";

}

scroller = scroller + scrtxt.substring(0, width - i + 1);

}else{

scroller = scroller + scrtxt.substring(pos, width + pos);

}

**//将当前要显示的部分字符发送到状态栏**

window.status = scroller;

**//0.2秒执行一次,以使显示出来的字符串内容得到更新并向左移动**

setTimeout("scroll()", 200);

}

**//-->**

</script>

</head>

<body onload='scroll()'>

<h2>状态栏滚动文字特效</h2>

</body>

**12.7检测浏览器和操作系统**

var isIE = (navigator.appName=="Microsoft Internet Explorer");

var isNetscape = (navigator.appName=="Netscape");

var isWin = (navigator.userAgent.indexOf("Win") != -1);

var isMac = (navigator.userAgent.indexOf("Mac") != -1);

var isUnix = (navigator.userAgent.indexOf("X11") != -1);

var isJava = navigator.javaEnabled()?"支持Java":"不支持Java";

document.write("您的浏览器为：" + (isIE?"Microsoft Internet Explorer":(isNetscape?"Netscape":"非IE和Netscape")) + "<br/>");

var system = "";

if(isWin){

system = "Windows";

}else if(isMac){

system = "Mac";

}else{

system = "Unix";

}

document.write("您的操作系统为：" + system + "<br/>");

document.write("您的浏览器" + isJava);

**12.8History对象和Location对象的使用**

<script language="javascript" type="text/javascript">

document.write("历史记录个数：" + history.length + "<br/>");

</script>

<form>

<input type="button" value="后退" onclick="history.back()"/>

<input type="button" value = "前进" onclick="history.forward()"/>

<input id="url" type="text" value="" />

<input type="button" value="转到" onclick="window.location=document.forms[0].url.value"/>

</form>

**12.9自动关闭的弹出窗口**

<script language="javascript" type="text/javascript">

<!--

var popWin;

**//弹出窗口函数**

function popWin(){

popWin = window.open('ad.html','win','width=250,height=200');

setTimeout('closeWin()',8000);

}

**//关闭弹出窗口函数**

function closeWin(){

if(popWin && !popWin.closed){

popWin.close();

}

}

//-->

</script>

</head>

<body onload='popWin()'>**//页面加载完打开窗口**

</body>

**12.10自动缩放大小的窗口**

<script language="javascript" type="text/javascript">

<!--

var popWin;

var count = 0;

var timer = null;

var countMain = 0;

**//弹出窗口函数**

function openWin(){

popWin = window.open('popWin.html','win','width=1,height=1');

resize();

}

function resize(){

popWin.resizeBy(10,8);

count++;

var timerID = setTimeout('resize()',20);

if(count==20){

clearTimeout(timerID);

}

}

function init(){

window.resizeTo(1,1);

timer = setInterval('resizeMain()',20);

}

function resizeMain(){

window.resizeBy(10,8);

countMain++;

if(countMain==30){

clearTimeout(timer);

}

}

//-->

</script>

</head>

<body onload='init()'>

<h2>自动缩放大小的窗口</h2>

<a href='javascript:openWin()'>打开新窗口</a>

</body>

<script language="javascript" type="text/javascript">

<!--

function closeWin(){

alert('谢谢光临，欢迎再来！');

}

//-->

</script>

</head>

<body onunload='closeWin()'>

<h2>关闭窗口时会出现提示</h2>

<a href='javascript:window.close()'>关闭窗口</a>

</body>

**14.1Form对象的属性**

<script language="javascript" type="text/javascript">

**//定义全局变量form用于引用表单对象**

var form;

**//定义函数init()用于初始化表单的属性**

function init(){

form = document.forms[0];

form.name="myform";

form.method="post";

form.action="data.jsp";

}

**//定义函数getFrompro()用于在新窗口中显示表单的属性**

function getFormPro(){

win = window.open("","","width=200,height=150");

if(win!=null&&!win.closed){

with(win.document){

**//输出表单的属性**

write("<li>name：" + form.name + "</li>");

write("<li>method：" + form.method + "</li>");

write("<li>action：" + form.action + "</li>");

close();

}

}

}

</script>

<body onload="init()">

<form >

<label>用户名：

<input type="text" name="textfield" />

</label>

<br />

<br />

<label>密码：

<input type="text" name="textfield3" />

</label>

<p>

<label>

<input type="button" name="Submit" value="表单属性" onclick="getFormPro()"/>

</label>

</p>

</form>

</body>

**14.2Form对象的方法**

<script language="javascript" type="text/javascript">

**//定义全局变量form用于引用表单对象**

var form;

**//定义函数init()用于初始化表单的属性**

function init(){

form = document.forms[0];

form.method="post";

**//设置form的onsubmit事件处理程序**

**//被赋于的应该是函数名hanleSubmit，不要带括号**

form.onsubmit = handleSubmit;

}

**//定义函数handleSubmit()用于处理提交前的工作**

function handleSubmit(){

**//弹出确认对话框让用户确认信息**

**if(confirm("你确认要提交吗？")){**

**//如果用户单击“确定”按钮则提交表单**

form.submit();

}else{

**//如果用户单击“取消”按钮则不提交表单**

return false;

}

}

</script>

<body onload="init()">

<form >

<label>姓名：

<input type="text" name="textfield" />

</label>

<br />

<br />

<label>性别：

<input type="text" name="textfield3" />

</label>

<br />

<br />

国籍：

<label>

<input type="text" name="textfield2" />

</label>

<br />

<p>

<label>

<input type="submit" name="Submit" value="确定" />

</label>

<label>

<input type="reset" name="Submit2" value="重置" />

</label>

</p>

</form>

</body>

**14.3输入框与按钮**

<script language="javascript" type="text/javascript">

**//定义全局变量form用于引用表单对象**

var form;

**//定义函数init()用于初始化表单的属性**

function init(){

form = document.forms[0];

form.method="post";

**//设置form的onsubmit事件处理程序**

**//被赋于的应该是函数名hanleSubmit，不要带括号**

form.onsubmit = handleSubmit;

}

**//定义函数handleSubmit()用于处理提交前的验证工作**

function handleSubmit(){

**//获取三个文本框中的值分别保存在变量name、sex和comment中**

var name = form.name.value;

var sex = form.elements[1].value;

var comment = form.elements.comment.value;

**//如果姓名为空则弹出警告对话框**

if(name == ""){

alert("请输入姓名!");

**//返回false不提交表单**

return false;

}else if(sex == ""){

alert("请输入性别!");

return false;

}else if(comment == ""){

alert("请输入自我评价!");

return false;

}

**//提交表单**

form.submit();

}

</script>

<body onload="init()">

<form >

<label>姓名：

<input type="text" name="name" />

</label>

<br />

<br />

<label>性别：

<input type="text" name="sex" />

</label>

<br />

<br />

自我评价：

<label><br />

<textarea name="comment" rows="6"></textarea>

</label>

<br />

<p>

<label>

<input type="submit" name="Submit" value="确定" />

</label>

<label>

<input type="reset" name="Submit2" value="重置" />

</label>

</p>

</form>

</body>

**14.4显示、隐藏密码**

<style>

body,input{

font:menu

}

</style>

<script language="JavaScript">

<!--

**//函数ps()用于将密码框改变为文本框**

function ps(){

if (this.forms.password.type="password"){

box.innerHTML="密码：<input type='text' name='password' size='20' value="+this.forms.password.value+">";

click.innerHTML="<a href='javascript:txt()'> 隐藏密码</a>"

}

}

**//函数txt()用于将文本框改变为密码框**

function txt(){

if (this.forms.password.type="text"){

box.innerHTML="密码：<input type='password' name='password' size='20' value="+this.forms.password.value+">";

click.innerHTML="<a href='javascript:ps()'> 显示密码</a>"

}

}

//-->

</script>

</head>

<body>

<h2>显示、隐藏密码</h2>

<form method="POST" action="" name="forms">

<span id="box">密码：

<input type="password" name="password" size="20" value="www.sjzue.edu.cn">

</span>

<span id="click">

<a href="javascript:ps()">显示密码</a>

</span>

</form>

</body>

**14.5复选框和单选按钮**

<script language="javascript" type="text/javascript">

var form;

function init(){

form = document.forms[0];

form.method="post";

form.onsubmit = handleSubmit;

}

function handleSubmit(){

var name = form.name.value;

var sex = form.sex[0].checked?"帅哥":"美女";

**//定义变量count选择的运动数**

var count = 0;

var array = new Array();

var len = form.sport.length;

for(var i = 0;i < len ;i++){

if(form.sport[i].checked){

**//变量count自加1**

count++;

array.push(form.sport[i].value);

}

}

win = window.open("","","width=300,height=200");

if(win!=null&& !win.closed){

with(win.document){

open("text/plain");

writeln("你好，" + name + sex + "。");

writeln("你共喜欢" + count + "项运动，包括：");

writeln(array + "。");

close();

}

}

}

</script>

<body onload="init()">

<form >

<label>姓名：

<input name="name" type="text" size="10" />

</label>

<br />

<br />

<label>性别：</label>

<label>

<input name="sex" type="radio" value="男" checked="checked" />

男</label>

<label>

<input type="radio" name="sex" value="女" />

女</label>

<label> </label>

<br />

<br />

喜欢的运动：<br />

<br />

<label>

<input type="checkbox" name="sport" value="游泳" />

游泳</label>

<label>

<input type="checkbox" name="sport" value="足球" />

足球</label>

<br />

<br />

<label>

<input type="checkbox" name="sport" value="篮球" />

篮球</label>

<label>

<input type="checkbox" name="sport" value="爬山" />

爬山</label>

<br />

<p>

<label>

<input type="submit" name="Submit" value="确定" />

</label>

<label>

<input type="reset" name="Submit2" value="重置" />

</label>

</p>

</form>

</body>

**14.6选择框和选项**

<script language="javascript" type="text/javascript">

**//定义函数addOptions()用于将选项添加到右侧选择框中**

function addOptions(){

**//获取文档中的两个选择框对象**

sel1 = document.form1.select1;

sel2 = document.form1.select2;

**//左侧选择框中被选中选项的索引**

var num = sel1.selectedIndex;

if (num > 1) {

**//使用Option构造函数创建新Option对象**

var option = new Option(sel1.options[num].text);

**//获取当前右侧选择框中的选项数量**

var len = sel2.options.length;

**//当新创建的选项添加到右侧选择框中**

sel2.options[len] = option;

**//删除左侧选择框中已经添加到右侧选择框中的选项**

sel1.options[num] = null;

}

sel1.selectedIndex = 100;

}

**//定义函数delOptions()用于实现删除右侧选择框中选项的功能**

function delOptions() {

**//定义变量opt引用右侧选择框中的options[]数组**

var opt = form1.select2.options;

**//使用for循环逐个判断选项是否被选中**

for(var i = 2;i < opt.length;i++){

if(opt[i].selected){

**//如果选项被选中，则将其则设置为null将其从选择框中删除**

opt[i] = null;

**//因为有选项被删除，options[]数组的长度就减少了1，所以i的值自减1**

i--;

}

}

}

</script>

<body>

<form name="form1">

<select name="select1">

<option>可以选择的水果 </option>

<option>--------------- </option>

<option value="香蕉">香蕉</option>

<option value="葡萄">葡萄</option>

<option value="苹果">苹果</option>

<option value="桔子">桔子</option>

</select>

<input name="button" type="button" onclick="addOptions()" value="选择" />

<select name="select2" size="10" multiple="multiple">

<option>已选择的水果</option>

<option>---------------</option>

</select>

<input name="button2" type="button" onclick="delOptions()" value="删除" />

</form>

</body>

**14.7表单验证**

<script language="javascript" type="text/javascript">

function checkForm(){

//定义变量form引用表单对象

var form = document.forms[0];

//分别定义变量引用表单元素对象

var name = form.name;

var pwd = form.pwd;

var pwd2 = form.pwd2;

var email = form.email;

var qq = form.qq;

var language = form.language;

**//验证姓名的长度**

if (name.value.length<4){

alert("姓名最小长度为4!");

name.select();

return false;

}

**//验证姓名的组成是否合法**

if(/[^A-Za-z0-9\_]/.exec(name.value)!=null){

alert("姓名只能由数字、字母、下划线组成!");

name.select();

return false;

}

**//验证密码的合法性**

if(/[\s]/.exec(pwd.value)!=null) {

alert("密码不能含有空白字符！");

pwd.focus();

return false;

}

**//验证密码的长度**

if(pwd.value.length <6) {

alert("密码最小长度为6!");

pwd.focus();

return false;

}

**//验证确认密码是否填写**

if(pwd2.value.length == 0) {

alert("请输入您的确认密码!");

pwd2.focus();

return false;

}

**//验证确认密码是否与原密码匹配**

if(pwd.value!=pwd2.value ) {

alert("您的密码与确认密码不匹配请重新输入!");

pwd2.focus();

return false;

}

**//验证电子邮件地址的合法性**

if(email.value.length !=0){

if(/^\S+@\S+\.\S+$/.exec(email.value)==null){

alert("E-Mail的格式不正确，请重新输入!");

email.select();

return false;

}

}

**//验证QQ的合法性**

if (isNaN(qq.value)||qq.value.length<1){

alert("QQ号码的格式不正确!");

qq.select();

return false;

}

**//验证编程语言是否已经选择，要求最少选择一种**

**//定义标记变量check**

var check = false;

**//使用for循环依次检查复选框是否已经选择**

for(var i=0;i<language.length;i++){

if(language[i].checked){

check = true

break;

}

}

**//如果一个都没有被选择，返回false**

if(!check){

alert("请至少选择一种编程语言！");

return false;

}

**//通过前面所有的验证，返回true提交表单**

return true;

}

</script>

<body>

<form id="form1" name="form1" method="post" action="" onsubmit="return checkForm()">

<label>姓名：

<input type="text" name="name" />

</label>

<p>

<label>密码：

<input type="password" name="pwd" />

</label>

</p>

<p>

<label>确认密码：

<input type="password" name="pwd2" />

</label>

</p>

<p>E-mail:

<label>

<input type="text" name="email" />

</label>

</p>

<p>

<label>QQ：

<input type="text" name="qq" />

</label>

</p>

<p>使用的编程语言（至少选择一种）：</p>

<label>

<input type="checkbox" name="language" value="checkbox" onsubmit="return checkForm()"/>

Java

</label>

<label>

<input type="checkbox" name="language" value="checkbox" />

C

</label>

<label>

<input type="checkbox" name="language" value="checkbox" />

C++

</label>

<p>

<label>

<input type="checkbox" name="language" value="checkbox" />

Delphi

<input type="checkbox" name="language" value="checkbox" />

</label>

Ruby

<label>

<input type="checkbox" name="language" value="checkbox" />

</label>

C#</p>

<p>

<label>

<input type="submit" name="Submit" value="提交" />

</label>

<label>

<input type="reset" name="Submit2" value="重置" />

</label>

</p>

</form>

</body>

**14.8上传图片前预览图片效果**

<script language="javascript" type="text/javascript">

<!--

function preview()

{

var file = document.form1.UpFile.value;

var fileext = file.substring(file.lastIndexOf("."),file.length);

fileext = fileext.toLowerCase();

if ((fileext!='.jpg')&&(fileext!='.gif')&&(fileext!='.jpeg')&&(fileext!='.png')&&(fileext!='.bmp'))

{

alert("系统不支持的图片格式！");

document.form1.UpFile.focus();

}

else

{

document.getElementById("preview").innerHTML="<img src='" + file + "' style='border:6px double #ccc';padding:5px;>";

}

}

//-->

</script>

<body>

<h2>请选择要上传的图片，下方预览效果</h2>

<form name="form1" method="POST" enctype="multipart/form-data">

<input type="file" name="UpFile" size="30" onchange="preview()">

<div id="preview">

</div>

</form>

</body>

**15.1添加和删除选项**

<script language="javascript" type="text/javascript">

**//定义全局变量form用于引用表单对象**

**var form;**

**//定义函数init()用于初始化表单的属性**

function init(){

form = document.forms[0];

form.name="myform";

form.method="post";

}

**//定义函数addOption()用于将文本框中的值添加到选择框中**

function addOption(value){

**//调用HTMLSelectElement对象add()方法添加选项**

form.select.add(new Option(value));

}

**//定义函数delOption()用于将选项从选择框中删除**

function delOption(index){

if(index>0){

**//调用HTMLSelectElement对象的remove()方法将选项删除**

form.select.remove(index);

}

}

</script>

<body onload="init()">

<form >

<label>城市：

<input type="text" name="city" />

</label>

<label>

<input type="button" name="Submit" value="确定" onclick="addOption(this.form.city.value)"/>

</label>

<p>

<label></label>

<label>

<select name="select" size="8">

<option>城市列表</option>

</select>

</label>

<label>

<input type="button" name="Submit2" value="删除" onclick="delOption(this.form.select.selectedIndex)"/>

</label>

</p>

</form>

</body>

**15.2处理表格**

<script language="javascript" type="text/javascript">

**//定义函数insertR()，用于实现在表格的指定位置插入一行的功能**

function insertR(indexStr){

**//获取表格对象**

table = document.getElementById("myTable");

index = parseInt(indexStr) - 1;

**//判断输入的值是否合法**

if(isNaN(index)||index<0||index>table.rows.length-1){

alert("请正确输入要插入的行!");

return;

}

**//调用insertRow()方法插入一行**

table.insertRow(index);

}

**//定义函数deleteR()，用于实现将表格指定位置的行删除**

function deleteR(row){

table = document.getElementById("myTable");

index = parseInt(row) - 1;

**//判断输入的值是否合法**

if(isNaN(index)||index<0||index>table.rows.length-1){

alert("请正确输入要删除的行!");

return;

}

**//调用deleteRow()方法删除指定的行**

table.deleteRow(index);

}

**//定义函数insertC()，用于实现在第一行的最后插入一个单元格的功能**

function insertC(){

table = document.getElementById("myTable");

**//获取HTMLTableElement对象的行的命令**

rowList = table.rows;

**//调用item()方法获取第一行**

**firstRow = rowList.item(0);**

**//调用调用insertCell()插入单元格**

**firstRow.insertCell(firstRow.cells.length);**

}

**//定义函数deleteC()，用于实现删除最后一行的第一个单元格的功能**

function deleteC(){

table = document.getElementById("myTable");

rowList = table.rows;

**//获取表格中的最后一行**

lastRow = rowList.item(rowList.length-1);

**//调用deleteCell()方法，删除指定的单元格**

lastRow.deleteCell(0);

}

</script>

</head>

<body>

<div name="myDiv"></div>

<form id="form1" name="form1" method="post" action="">

<h2>插入删除行</h2>

<label>

在第

<input name="insertNum" type="text" size="5" />

</label>

行前插入一行

<label>

<input type="button" name="Submit" value="插入" onclick="insertR(this.form.insertNum.value)"/>

</label>

<p>删除第

<label>

<input name="deleteNum" type="text" size="5" />

</label>

行

<label>

<input type="button" name="Submit2" value="删除" onclick="deleteR(this.form.deleteNum.value)"/>

</label>

</p>

<h2>插入删除单元格</h2>

<p>

<label></label>

在第一行的末尾插入单元格

<label>

<input type="button" name="Submit3" value="插入" onclick="insertC()"/>

</label>

</p>

<p>删除最后一行第一个单元格

<label>

<input type="button" name="Submit4" value="删除" onclick="deleteC()"/>

</label>

</p>

</form>

<table id="myTable" border="1" >

<tr>

<td width="20%" >第一行</td>

<td width="20%" >&nbsp;</td>

<td width="20%" >&nbsp;</td>

<td width="20%" >&nbsp;</td>

<td width="20%">&nbsp;</td>

</tr>

<tr>

<td>第二行</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>第三行</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>第四行</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

<tr>

<td>第五行</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

<td>&nbsp;</td>

</tr>

</table>

</body>

**15.3查找文档中的元素与设置元素属性**

<script language="javascript" type="text/javascript">

**//定义函数changeColor()，用于设置表格的背景颜色**

function changeColor(){

**//使用getElementById()方法获取id为myTable的表格**

var table = document.getElementById("myTable");

**//使用getAttribute()方法获取表格的背景颜色的值**

var color = table.getAttribute("bgColor");

**//使用setAttribute()方法设置表格的背景颜色**

table.setAttribute("bgColor","blue");

alert("颜色设置成功!原来的颜色为" + color + "。");

}

**//定义函数getLinks()，用于获取文档中的链接**

function getLinks(){

**//使用getElementsByTagName()方法获取链接的集合**

var links = document.getElementsByTagName("a");

**//使用length属性获取链接的个数**

var address = new Array(links.length);

for(var i=0;i<links.length;i++){

**//使用item()方法获取指定的元素**

address[i] = links.item(i).getAttribute("href");

}

win = window.open("","","width=200,height=150");

if(win!=null && !win.closed){

for(var i=0;i<address.length;i++){

win.document.write("<li>"+ address[i] +"</li>");

}

win.document.close();

}

}

</script>

</head>

<body>

<form id="form1" name="form1" method="post" action="">

<table id="myTable" width="200" border="1" bgcolor="#FF0000">

<tr>

<th scope="col">&nbsp;</th>

</tr>

</table>

<br/>

<label>

<input type="button" name="Submit" value="更改颜色" onclick="changeColor()"/>

</label>

<ul>

<li><a href="http://www.csdn.net">CSDN</a></li>

<li><a href="http://www.javaeye.com">JavaEye</a></li>

<li><a href="http://www.163.com">网易</a></li>

<li><a href="http://www.sohu.com">搜狐</a></li>

<li><a href="http://www.sina.com">新浪</a></li>

</ul>

<input name="button" type="button" value="网址一览" onclick="getLinks()"/>

</form>

</body>

**15.4遍历文档元素**

<script language="javascript" type="text/javascript">

**//定义函数getInfo()，用于获取小节的详细信息**

function getInfo(num){

**//使用getElementById()方法获取<ul>元素以及<div>元素**

var list = document.getElementById("list");

var result = document.getElementById("result");

**//使用childNodes数组导航到相应的小节**

var node = list.childNodes[num];

**//如果小节中有详细的内容则subNode表示小节中的列表**

var subNode = node.childNodes[1];

**//判断子小节中是否有内容**

if(subNode!=null && subNode.hasChildNodes()){

**//根据子小节的数量创建数组**

var array = new Array(subNode.childNodes.length);

for(var i=0;i<subNode.childNodes.length;i++){

**//使用nodeValue属性将子小节的具体内容保存到数组array中**

array[i] = subNode.childNodes[i].lastChild.nodeValue;

}

**//显示子小节的内容**

result.childNodes[0].nodeValue = array.toString();

}else{

result.childNodes[0].nodeValue = "无详细内容!";

}

}

</script>

</head>

<body>

<ul id="list">

<li>16.1 DOM概述</li>

<li> 16.2 树与结点

<ul>

<li> 16.2.1 HTML文档与DOM树</li>

<li> 16.2.2 节点 </li>

</ul>

</li>

<li>16.3 使用DOM HTML API

<ul>

<li>16.3.1 DOM HTML API简述 </li>

<li>16.3.2 HTMLCollection </li>

<li>16.3.3 属性的命名 </li>

<li>16.3.4 添加和删除选择框中选项 </li>

<li>16.3.5 处理表格 </li>

</ul>

</li>

<li>16.4 使用DOM Core API

<ul>

<li>16.4.1 Node和NodeList </li>

<li>16.4.2 查找文档中的元素与设置元素属性 </li>

<li>16.4.3 遍历文档元素 </li>

<li>16.4.4 创建文档元素 </li>

<li>16.4.5 动态修改文档的内容 </li>

</ul>

</li>

</ul>

<form id="form1" name="form1" method="post" action="">

<label>

<input type="button" name="Submit" value="第一小节" onclick="getInfo(0)"/>

</label>

<label>

<input type="button" name="Submit2" value="第二小节" onclick="getInfo(1)"/>

</label>

<label>

<input type="button" name="Submit3" value="第三小节" onclick="getInfo(2)"/>

</label>

<label>

<input type="button" name="Submit4" value="第四小节" onclick="getInfo(3)"/>

</label>

</form>

<div id="result">此处显示结果</div>

</body>

**15.5我的收藏夹**

<script language="javascript" type="text/javascript">

**//定义函数add用于将链接添加到列表中**

function add(){

**//使用Document对象的getElementById()方法分别获取文本框元素以及div元素**

var title = document.getElementById("title").value;

var url = document.getElementById("url").value;

var list = document.getElementById("list");

**//使用Document对象的createElement()方法创建<a>元素**

var link = document.createElement("a");

**//使用setAttribute()方法设置href属性**

link.setAttribute("href",url);

**//使用innerHTML属性设置<a></a>之间的文本**

link.innerHTML = title;

**//创建输入元素对象**

var button = document.createElement("input");

**//为输入元素对象设置type属性，这个是必须的，因为多个元素都使用<input>标签**

button.setAttribute("type","button");

**//设置value属性，也可以使用setAttribute()方法**

button.value="删除";

**//设置onclick事件的事件处理程序为deLink()函数**

button.onclick = delLink;

**//创建div元素**

var div = document.createElement("div");

**//使用appendChild()方法将按钮添加到div元素中**

div.appendChild(button);

**//使用insertBefore()方法将链接添加到按钮的前面**

div.insertBefore(link,button);

**//将div元素添加到列表中**

**list.appendChild(div);**

}

**//定义函数delLink()用于实现将链接从列表中删除**

function delLink(event){

**//定义变量target代表要被的按钮**

var target;

**//使用Event对象获取目标按钮**

if(event==null){

target = window.event.srcElement;

}else{

target = event.target;

}

**//使用parentNode属性获取容纳链接的div元素**

var div = target.parentNode;

**//使用removeChild()方法删除整个div元素，包括其中的链接和按钮**

div.parentNode.removeChild(div);

alert("删除成功!");

}

</script>

</head>

<body>

<h2>我的收藏夹</h2>

<form id="form1" name="form1" method="post" action="">

<label>名称：

<input type="text" name="title" id="title"/>

</label>

<label>网址：

<input type="text" name="url" id="url"/>

</label>

<label>

<!--设置onclick属性的值为“add()”-->

<input type="button" name="Submit" value="添加" onclick="add()"/>

</label>

</form>

<hr />

<div id="list">

</div>

</body>

**16.1绑定事件处理器**

<body>

<form id="form1" name="form1" method="post" action="">

<label>姓名：

<input type="text" id="username" name="username" />

</label>

<p>

<label>

<input type="button" id="button1" name="button1" value="确定1" onclick="sayHello()"/>

<input type="button" id="button2" name="button2" value="确定2" />

</label>

<label>

<input type="button" id="button3" name="Submit3" value="确定3" />

</label>

</p>

</form>

<script language="javascript" type="text/javascript">

**//获取button2元素**

var button2 = document.getElementById("button2");

**//设置button2的onclick属性为sayHello**

button2.onclick = sayHello;

**//定义函数sayHello()**

function sayHello(){

var username = document.getElementById("username").value;

alert("你好，" + username + "。");

}

var button3 = document.getElementById("button3");

**//使用匿名函数设置button3的onclick属性**

button3.onclick = function(){

**//匿名函数实现和sayHello()函数相同的功能**

var username = document.getElementById("username").value;

alert("你好，" + username + "。");

}

</script>

</body>

16.2事件处理函数的作用域和this关键字

<body>

<form id="form1" name="form1" method="post" action="">

<label>城市：

<input type="text" id="city" name="city" />

</label>

<p>

<label>

<!--在onclick属性中使用this关键字，引用当前的按钮对象-->

<input type="button" id="button1" name="button1" value="确定1" onclick="alert(this.form.city.value)"/>

<input type="button" id="button2" name="button2" value="确定2" />

</label>

</p>

</form>

<script language="javascript" type="text/javascript">

**//获取button2元素**

var button2 = document.getElementById("button2");

**//设置button2的onclick属性为getCity**

button2.onclick = getCity;

**//定义函数getCity()**

function getCity(){

**//this关键字引用button2元素**

var city = this.form.city.value;

alert(city);

}

</script>

</body>

**16.3改变事件的默认行为**

<script language="javascript" type="text/javascript">

**//函数validateForm()用于表单的验证**

function validateForm(){

var form = document.getElementById("form1");

var name = form.username;

var pwd = form.pwd;

//验证用户名的长度

if (name.value.length<4){

alert("用户名最小长度为4!");

name.select();

**//返回false**

return false;

}

**//验证用户名的组成是否合法**

if(/[^A-Za-z0-9\_]/.exec(name.value)!=null){

alert("用户名只能由数字、字母、下划线组成!");

name.select();

**//返回false**

return false;

}

**//验证密码的合法性**

if(/[\s]/.exec(pwd.value)!=null) {

alert("密码不能含有空白字符！");

pwd.focus();

**//返回false**

return false;

}

**//验证密码的长度**

if(pwd.value.length <6) {

alert("密码最小长度为6!");

pwd.focus();

**//返回false**

return false;

}

}

</script>

</head>

<body>

<form id="form1" name="form1" method="post" action="" onsubmit="return validateForm()">

<label>用户名：

<input type="text" name="username" />

<br />

<br />

密码：

<input type="password" name="pwd" />

</label>

<p>

<label>

<input type="submit" name="Submit" value="提交" />

</label>

<label>

<input type="reset" name="Submit2" value="重置" />

</label>

</p>

</form>

</body>

16.4Netscape的Event对象

<body>

<form id="form1" name="form1" method="post" action="">

<label></label>

<h2>Netscape的Event对象</h2>

<p>

<label>

<!--手动传入event对象-->

<input type="button" name="Submit" value="按钮1" onclick="show(event)"/>

</label>

<label>

<input id="button2" type="button" name="Submit2" value="按钮2" />

</label>

</p>

</form>

<script language="javascript" type="text/javascript">

var button2 = document.getElementById("button2");

**//设置button2的onclick属性**

button2.onclick= show;

**//show函数有一个event作为参数**

function show(event){

**//使用Event对象的target属性获取事件源**

alert(event.target.value +"被点击！");

}

</script>

</body>

**16.5使用&lt;script&gt;标签的属性绑定事件处理器**

<!--为myButton按钮绑定onclick事件处理器-->

<script language="javascript" type="text/javascript" for="myButton" event="onclick">

alert("你单击了按钮！");

</script>

<!--为myBolg超链接绑定onmouseover事件处理器-->

<script language="javascript" type="text/javascript" for="myBlog" event="onmouseover">

if(confirm("你确认要转到我的博客吗？")){

window.location = document.getElementById("myBlog").href;

}

</script>

</head>

<body>

<form id="form1" name="form1" method="post" action="">

<h2>使用&lt;script&gt;标签的属性绑定事件处理器</h2>

<p>

<label>

<input id="myButton" type="button" name="Submit" value="单击我" />

</label>

</p>

<p><a id="myBlog" href="http://smartdong.javaeye.com">我的博客</a></p>

</form>

</body>

**16.6可拖动窗口**

<style type="text/css">

body{

font-family:Verdana;

font-size:11px;

color:#333333;

}

#win{

position:absolute;

left:50px;

top:50px;

width:200px;

height:150px;

border:1px solid #000000;

}

#title{

width:100%;

height:18px;

color:#ffffff;

cursor: move;

background-color: #0066FF;

}

</style>

<script language="javascript" type="text/javascript">

**//定义变量move表示窗口是否可以移动**

var move=false;

**//变量left和变量top分别表示窗口的X坐标和Y坐标**

var left = 50;

var top = 50;

**//变量deltaLeft表示鼠标的X坐标与窗口X坐标之差**

var deltaLeft = 0;

**//变量deltaTop表示鼠标的Y坐标与窗口Y坐标之差**

var deltaTop = 0;

**//定义变量title表示窗口的标题栏**

var title ;

**//定义函数init()初始化title变量**

**function init(){**

title = document.getElementById("title");

title.attachEvent("onmousedown",startDrag);

title.attachEvent("onmousemove",drag);

title.attachEvent("onmouseup",stopDrag);

}

**//定义函数startDrag()初始化拖动相关参数**

function startDrag(){

**//判断是否是鼠标左键按下并且点击在div元素上**

if(event.button == 1 && event.srcElement.tagName.toLowerCase()=="div"){

deltaLeft = event.clientX - left;

deltaTop = event.clientY - top;

**//调用setCapture()方法**

title.setCapture();

title.style.background="#0099FF";

**//将变量move设置为true**

**move=true;**

}

}

**//定义函数drag()实现拖动的功能**

function drag(){

if(move){

**//获取容纳标题栏窗口的整个div窗口**

var win = title.parentNode;

**//通过设置style属性改变窗口的坐标**

win.style.left=event.clientX-deltaLeft;

win.style.top=event.clientY-deltaTop;

}

}

**//定义函数stopDrag()，当鼠标松开时停止拖动**

function stopDrag(){

**//将标题栏背景颜色改为原来的颜色**

title.style.background="#0066FF";

**//获取窗口当前的坐标**

left = event.clientX-deltaLeft;

**top = event.clientY-deltaTop;**

**//调用releaseCapture()方法**

title.releaseCapture();

**//将move设置为false;**

move=false;

}

</script>

</head>

<body onload="init()">

<div id="win">

<div id="title" onMousedown="startDrag(this)" onMouseup="stopDrag(this)" onMousemove="drag(this)" >拖动窗口</div>

用鼠标拖动窗口标题栏移动窗口

</div>

</body>

**16.7冒泡事件传递机制**

<body onclick="show('BODY元素')">

<form id="form1" name="form1" method="post" action="">

<h2>分别单击下面两个按钮</h2>

<div id="myDiv" onclick="show('DIV元素')">

<p id="myP" onclick="show('P元素')">

<label>

<input id="button1" type="button" name="Submit" value="按钮1" onclick="show('按钮1')"/>

</label>

<label>

<input id="button2" type="button" name="Submit2" value="按钮2" onclick="show2('按钮2')"/>

</label>

</p>

</div>

</form>

<div id="result"></div>

<script language="javascript" type="text/javascript">

function show(name){

result.innerHTML += "<b>" + name + "</b>被单击了!<br/>";

}

function show2(name){

result.innerHTML += "<b>" + name + "</b>被单击了!<br/>";

**//设置cancelBubble属性为true，中止事件的传递**

event.cancelBubble = true;

}

</script>

</body>

**16.8DOM 2事件处理**

</head>

<body id="myBody">

<form id="form1" name="form1" method="post" action="">

<h2>单击下面的按钮</h2>

<div id="myDiv">

<p id="myP">

<label>

<input id="button" type="button" name="Submit" value="确定" />

</label>

</p>

</div>

</form>

<div id="result"></div>

<script language="javascript" type="text/javascript">

**//获取input元素并为其注册捕获阶段和冒泡阶段的事件处理器**

var button = document.getElementById("button");

button.addEventListener("click",show,true);

button.addEventListener("click",show,false);

**//获取p元素并为其注册捕获阶段和冒泡阶段的事件处理器**

var p = document.getElementById("myP");

p.addEventListener("click",show,true);

p.addEventListener("click",show,false);

**//获取div元素并为其注册捕获阶段和冒泡阶段的事件处理器**

var div = document.getElementById("myDiv");

div.addEventListener("click",show,true);

div.addEventListener("click",show,false);

**//获取body元素并为其注册捕获阶段和冒泡阶段的事件处理器**

var body = document.getElementById("myBody");

body.addEventListener("click",show,true);

body.addEventListener("click",show,false);

**//获取用于显示结果的div元素**

var result = document.getElementById("result");

**//定义事件处理函数show**

**function show(event){**

var phase;

if(event.eventPhase == Event.CAPTURING\_PHASE){

phase = "事件捕获阶段：";

}else if(event.eventPhase == Event.AT\_TARGET){

phase = "到达目标阶段：";

}else{

phase = "事件冒泡阶段：";

}

result.innerHTML += phase + "<b>" + event.currentTarget.tagName.toUpperCase() + "</b>元素被单击了!<br/>";

}

</script>

</body>

**16.9跟随鼠标的文字**

<STYLE>

.spanstyle {

COLOR: #0066ff;

FONT-FAMILY: 隶书;

FONT-SIZE: 14pt;

FONT-WEIGHT: normal;

POSITION: absolute;

TOP: -50px;

VISIBILITY: visible

}

</STYLE>

<script language="javascript" type="text/javascript">

**//定义跟随鼠标的文字**

var message="欢 迎 来 到 javascript 世 界 !";

**//定义变量x,y用于记录鼠标的坐标**

var x,y;

**//定义变量step表示字符的间距**

var step = 12;

**//定义标记变量flag并初始化为0**

var flag = 0;

message = message.split("");

var xpos = new Array();

for (i = 0;i <= message.length - 1;i++) {

xpos[i] = -50;

}

var ypos = new Array();

for (i = 0;i <= message.length - 1;i++) {

ypos[i] = -50;

}

**//定义初始化函数init()**

function init() {

makeSnake();

**//设置定时器**

window.setTimeout("init();", 2);

}

**//定义函数handerMM()用于在鼠标移动时获取鼠标的坐标**

function handlerMM(e) {

x = (document.layers) ? e.pageX : document.body.scrollLeft + event.clientX+10;

y = (document.layers) ? e.pageY : document.body.scrollTop + event.clientY;

flag=1;

}

function makeSnake() {

**//如果是在IE浏览器中**

if (flag == 1 && document.all) {

for (i = message.length-1; i>=1; i--) {

xpos[i] = xpos[i-1]+step;

ypos[i] = ypos[i-1];

}

xpos[0] = x + step;

ypos[0] = y;

for (i = 0; i <= message.length-1; i++) {

var thisspan = eval("span"+(i)+".style");

thisspan.posLeft = xpos[i];

thisspan.posTop = ypos[i];

thisspan.color = Math.random() \* 255 \* 255 \* 255 + Math.random() \* 255 \* 255 + Math.random() \* 255;

}

**//如果是在Netscape浏览器中**

}else if (flag == 1 && document.layers) {

for (i = message.length - 1; i>=1; i--) {

xpos[i] = xpos[i-1] + step;

ypos[i] = ypos[i-1];

}

xpos[0] = x + step;

ypos[0] = y;

for (i=0; i<message.length-1; i++) {

var thisspan = eval("document.span"+i);

thisspan.left = xpos[i];

thisspan.top = ypos[i];

thisspan.color=Math.random() \* 255 \* 255 \* 255 + Math.random() \* 255 \* 255 + Math.random() \* 255;

}

}

}

</script>

</head>

<body onload="init()">

<script type="text/javascript">

**//将要显示的字符串放入到一个span元素中并设置其CSS样式**

for (i=0;i<=message.length-1;i++) {

document.write("<span id='span"+i+"' class='spanstyle'>");

document.write(message[i]);

document.write("</span>");

}

if (document.layers) {

document.captureEvents(Event.MOUSEMOVE);

}

document.onmousemove = handlerMM;

</script>

</body>

**17.1常用文本属性**

<style type="text/css">

<!--

.rifht {

text-align: right;

}

.center {

text-align: center;

}

.line1 {

text-decoration: underline;

}

.line2 {

text-decoration: line-through;

}

#space {

letter-spacing: 8px;

}

-->

</style>

</head>

<body>

<p class="left">设置文本的对齐方式为右对齐</p>

<p class="center">设置文本的对齐方式为居中对齐</p>

<p class="line1">具有下划线的文本</p>

<p class="line2">具有删除线的文本</p>

<p id="space">设置字符间的距离</p>

</body>

**17.2常用定位属性**

<style type="text/css">

<!--

.layer1 {

position: absolute;

z-index: 0;

height: 200px;

width: 300px;

background-color: #0000FF;

}

.layer2 {

position: absolute;

z-index: 1;

height: 200px;

width: 250px;

left: 100px;

top: 100px;

background-color: #FFFF00;

}

-->

</style>

</head>

<body>

<div class="layer1">此处显示 class "layer1" 的内容</div>

<div class="layer2">此处显示 class "layer2" 的内容</div>

</body>

**17.3对象的显示与隐藏**

<style type="text/css">

<!--

.myDiv {

height: 50px;

background-color: #00FFFF;

}

-->

</style>

<script type="text/javascript">

**//定义标记变量分别标记图像和div元素的隐藏和显示情况**

var hidden1 = false;

var hidden2 = false;

function change1(){

button1 = document.getElementById("button1");

img = document.getElementById("image");

if(!hidden1){

img.style.display="none";

hidden1 = true;

button1.value="显示"

}else{

img.style.display="";

hidden1 = false;

button1.value="隐藏"

}

}

function change2(){

button2 = document.getElementById("button2");

div = document.getElementById("myDiv");

if(!hidden2){

div.style.visibility="hidden";

hidden2 = true;

button2.value="显示"

}else{

div.style.visibility="visible";

hidden2 = false;

button2.value="隐藏"

}

}

</script>

</head>

<body>

<p>使用display属性控制对象的显示与隐藏<br/>

<img id="image" src="back.jpg" width="224" height="224"></img></p>

<form id="form1" name="form1" method="post" action="">

<label>

<input type="button" id="button1" name="Submit" value="隐藏" onclick="change1()"/>

</label>

<label></label>

</form>

<hr/>

<div id="myDiv" class="myDiv">使用visibility属性控制对象的显示与隐藏</div>

<form id="form2" name="form2" method="post" action="">

<label>

<input type="button"id="button2" name="Submit2" value="隐藏" onclick="change2()"/>

</label>

</form>

</body>

**17.4常用边框属性**

<style type="text/css">

<!--

#style1 {

border: 1px dashed #FF0000;

}

#style2 {

border-top-width: thin;

border-right-width: thin;

border-bottom-width: thin;

border-left-width: thin;

border-top-style: double;

border-right-style: dotted;

border-bottom-style: double;

border-left-style: dotted;

border-top-color: #0000FF;

border-right-color: #0000FF;

border-bottom-color: #0000FF;

border-left-color: #0000FF;

}

#style3 {

border-top-width: 3px;

border-right-width: 3px;

border-bottom-width: 3px;

border-left-width: 3px;

border-top-style: inset;

border-right-style: inset;

border-bottom-style: inset;

border-left-style: inset;

}

#style4 {

border-top-width: 3px;

border-right-width: 5px;

border-bottom-width: 3px;

border-left-width: 5px;

border-top-style: outset;

border-right-style: outset;

border-bottom-style: outset;

border-left-style: outset;

border-top-color: #FFFF00;

border-right-color: #FFFF00;

border-bottom-color: #FFFF00;

border-left-color: #FFFF00;

}

-->

</style>

</head>

<body>

<div id="style1">边框为虚线，宽度为中，颜色为红色的边框</div>

<p/>

<div id="style2">上下边框为双线，左右边框为点划线，颜色为蓝色的边框</div>

<p/>

<div id="style3">边框为凹陷，宽度为3 px的边框 </div>

<p/>

<div id="style4">边框为凸出，上下边框宽度为3 px，左右边框宽度为5px，颜色为黄色的边框</div>

</body>

**17.5闪烁的文字**

<script type='text/javascript'>

<!--

var flag = 0;

function startTimer(){

var text = document.getElementById('text');

if(flag == 0){

text.style.color = '#FF0000';

text.style.textDecoration = 'line-through';

text.style.backgroundColor = '#0000FF';

flag = 1;

}else{

text.style.color = '';

text.style.textDecoration = '';

text.style.backgroundColor = '';

flag = 0;

}

setTimeout('startTimer()',500);

}

//-->

</script>

</head>

<body onload="startTimer()">

<h2>闪烁的文字</h2>

<span id='text'>欢迎来到CSS世界！</span>

</body>

**17.10JavaScript下拉菜单**

<style type="text/css">

<!--

\*{margin:0;padding:0;border:0;}

body {

font-family: arial, 宋体, serif;

font-size:12px;

}

#nav {

line-height: 24px; list-style-type: none; background:#666;

}

#nav a {

display: block; width: 80px; text-align:center;

}

#nav a:link {

color:#666; text-decoration:none;

}

#nav a:visited {

color:#666;text-decoration:none;

}

#nav a:hover {

color:#FFF;text-decoration:none;font-weight:bold;

}

#nav li {

float: left; width: 80px; background:#CCC;

}

#nav li a:hover{

background:#999;

}

#nav li ul {

line-height: 27px; list-style-type: none;text-align:left;

left: -999em; width: 100px; position: absolute;

}

#nav li ul li{

float: left; width: 100px;

background: #F6F6F6;

}

#nav li ul a{

display: block; width: 100px;text-align:left;padding-left:24px;

}

#nav li ul a:link {

color:#666; text-decoration:none;

}

#nav li ul a:visited {

color:#666;text-decoration:none;

}

#nav li ul a:hover {

color:#F3F3F3;text-decoration:none;font-weight:normal;

background:#C00;

}

#nav li:hover ul {

left: auto;

}

#nav li.sfhover ul {

left: auto;

}

-->

</style>

<script type=text/javascript>

<!--

function setMenu() {

var sfEls = document.getElementById("nav").getElementsByTagName("li");

for (var i=0; i<sfEls.length; i++) {

sfEls[i].onmouseover = function() {

this.className += (this.className.length>0? " ": "") + "sfhover";

}

sfEls[i].onmouseout = function() {

this.className = this.className.replace(new RegExp("( ?|^)sfhover\\b"),"");

}

}**//for循环结束**

}

window.onload=setMenu;

//--></script>

</head>

<body>

<ul id="nav">

<li><a href="#">首页</a>

</li>

<li><a href="#">会计审计</a>

<ul>

<li><a href="#">会计审计</a></li>

<li><a href="#">会计审计</a></li>

<li><a href="#">会计审计</a></li>

<li><a href="#">会计审计</a></li>

</ul>

</li>

<li><a href="#">财务管理</a>

<ul>

<li><a href="#">财务管理</a></li>

<li><a href="#">财务管理</a></li>

<li><a href="#">财务管理</a></li>

<li><a href="#">财务管理</a></li>

</ul>

</li>

<li><a href="#">电算会计</a>

<ul>

<li><a href="#">电算会计</a></li>

<li><a href="#">电算会计</a></li>

<li><a href="#">电算会计</a></li>

<li><a href="#">电算会计</a></li>

</ul>

</li>

</ul>

</body>