



第3章 JavaScript对象编程



学习目标



- 学会使用Date对象
- 学会使用Math对象
- 学会使用数组对象
- 学会使用字符串对象
- 学会使用正则表达式对象

本章任务



- 任务一：制作动态时钟效果



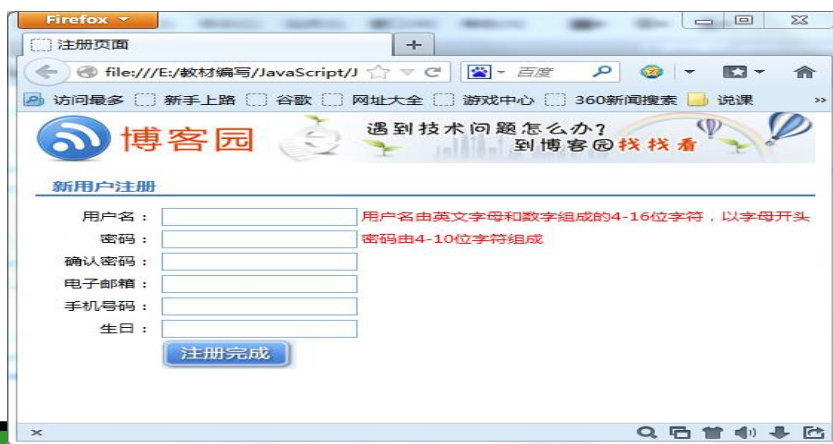
本章任务



- 任务三：制作简单的焦点图效果



- 任务四：验证注册页面信息



本章任务



- 任务五：使用正则表达式验证注册页面信息

 **博客园**  遇到技术问题怎么办?
到博客园找查看 

新用户注册

用户名: 请输入4-16位用户名

密码: 密码由4-10位字符组成

确认密码:

电子邮箱:

手机号码:

出生日期: 请按xxxx-xx-xx输入出生日期

什么是对象



属性
Height
Sex
Age
Career
.....



方法
Eat()
Speak()
Breath()
Think()
Sleep()
.....

属性
bgColor
Location
Cookie
Height
.....



方法
Open()
Close()
Focus()
Write()
.....

什么是对象



- 对象由两部分组成：
 - 一组包含各种类型数据的属性
 - 对属性中的数据进行操作的方法

JavaScript的对象



- JavaScript支持的对象包括以下几种：

- 内置对象

Date、Math、String、Array、Number等

- 浏览器对象

Window、Location、History、Navigator、Screen、
Document、Frame等

- 自定义对象

任务1:在页面动态显示系统时间



任务描述:



在“我的博客”首页动态显示客户端系统时间。

任务分析:

在页面动态显示系统时间，可以采用以下步骤：

1. 完成静态页面设计，标识要显示系统时间的位置，此例已完成该步骤。
2. 定义函数，使用日期和时间对象，获取客户端系统时间。
3. 为id为time的标签赋值。
4. 使用定时函数，每隔一秒重新调用一次函数。

时钟显示



问题

如何实现如时钟效果？



分析

由于涉及到时间的显示问题，所以要用到日期对象**Date**。

Date对象



Date 对象存储的日期为自 1970 年 1 月 1 日
00:00:00 以来的毫秒数
首先创建对象实例：

```
var 日期对象 = new Date ( )
```

```
var 日期对象 = new Date (毫秒数)
```

```
var 日期对象 = new Date (字符串)
```

```
var 日期对象 = new Date (年、月、日等参数)
```



```
var 日期对象 = new Date ( )  
无参数表示当前日期和时间  
var mydate=new Date( )  
document.write(mydate);
```

例如：

```
var mydate=new Date("July 29, 2007,10:30:00" )  
document.write(mydate);
```

例如：

```
var mydate=new Date(3580)  
document.write(mydate);
```

时间标准



- **UTC: 全球标准时间(Coordinated Universal Time)**,由世界时间标准设定的时间,原先也被称为格林威治标准时间(**GMT**)。

UTC+800: UTC是零时区,即格林威治标准时间; 800的含义是8:00即8个小时0分; UTC+800是指比格林威治标准时间提前8个小时,应当是东八区的时间,也就是北京时间。



```
var 日期对象 = new Date ( )
```

无参数表示当前日期和时间

```
var mydate=new Date (2011,3,20,6,7,8)
```

```
document.write(mydate);
```

Date 对象的方法



- Date 方法的分组:

方法分组	说 明
setxxx	这些方法用于设置时间和日期值
getxxx	这些方法用于获取时间和日期值
Toxxx	这些方法用于从 Date 对象返回字符串值
parsexxx & UTCxx	这些方法用于解析字符串

Date 对象方法的参数



- 用作 Date 方法的参数的整数：

值	整数
Seconds 和 minutes	0 至 59
Hours	0 至 23
Day	0 至 6 （星期几）
Date	1 至 31 （月份中的天数）
Months	0 至 11 （一月至十二月）

Date 对象方法



- Set 方法:

方法	说明
setDate	设置 Date 对象中月份中的天数，其值介于 1 至 31 之间。
setHours	设置 Date 对象中的小时数，其值介于 0 至 23 之间。
setMinutes	设置 Date 对象中的分钟数，其值介于 0 至 59 之间。
setSeconds	设置 Date 对象中的秒数，其值介于 0 至 59 之间。
setTime	设置 Date 对象中的时间值。
setMonth	设置 Date 对象中的月份，其值介于 1 至 12 之间。

Date 对象方法



- Get 方法:

方法	说明
getDate	返回 Date 对象中月份中的天数，其值介于 1 至 31 之间
getDay	返回 Date 对象中的星期几，其值介于 0 至 6 之间
getHours	返回 Date 对象中的小时数，其值介于 0 至 23 之间
getMinutes	返回 Date 对象中的分钟数，其值介于 0 至 59 之间
getSeconds	返回 Date 对象中的秒数，其值介于 0 至 59 之间
getMonth	返回 Date 对象中的月份，其值介于 0 至 11 之间
getFullYear	返回 Date 对象中的年份，其值为四位数
getTime	返回自某一时刻（1970 年 1 月 1 日）以来的毫秒数

Date 对象方法



- To 方法:

方法	说明
ToGMTString	使用格林尼治标准时间 (GMT) 数据格式将 Date 对象转换成字符串表示
ToLocaleString	使用当地时间格式将 Date 对象转换成字符串表示

- Parse 方法和 UTC 方法

方法	说明
Date.parse (date string)	用日期字符串表示自 1970 年 1 月 1 日以来的毫秒数
Date.UTC (year, month, day, hours, min., secs.)	Date 对象中自 1970 年 1 月 1 日以来的毫秒数

Date对象做时钟显示



示例

[查看完整代码](#)

```
<SCRIPT language="JavaScript">
function disptime( )
{ var now= new Date( );
  var hour = now.getHours( );
  if (hour>=0 && hour <=12)
    document.write("<H2>上午好!</H2>")
  if (hour>12 && hour<= 18)
    document.write("<H2>下午好!</H2>");
  if (hour>18 && hour <24)
    document.write("<H2>晚上好!</H2>");
  document.write("<H2>今天日期:"+now.getFullYear( )+
    "+(now.getMonth( )+1)+"月"+now.getDate()+"日</H2>");
  document.write("<H2>现在时间:"+now.getHours( )+
    "+now.getMinutes( )+"分</H2>");
}
</SCRIPT>
<BODY onload="disptime( )">
```

获得当前日期和时间

获得小时，即当前是几点

判断上午、下午还是晚上

月份数字
0-11,
注意+1



晚上好!

今天日期:2009年4月5日

现在时间:21点4分

练习



晚上好!

今天日期:2010年3月22日

现在时间:20点4分

0/1



100%

Date对象做时钟显示



问题

上一页PPT示例中时钟不能动态改变，怎么办？



分析

由于时间在不停地走，所以应该每隔1秒调用显示时间的方法。如何解决？

使用setTimeout()方法每隔1秒调用显示时间的方法

setTimeout()的用法



该方法在设定的时间到来时触发某指定的事件。

`var timer=window.setTimeout (“调用的函数”, ”定时的毫秒数”)`

返回值`timer`为该事件超时的引用变量名。

提供`clearTimeout()`方法清除定时操作，该方法唯一参数就是事件超时的引用变量名。

例：`var timer=setTimeout(“disptime()”, 1000) ;`
`clearTimeout(timer) ;`

设定时间间隔



- `setInterval()`：用于设定时间间隔，用于按某个指定时间间隔去周期触发某个事件。激活后就会重复执行，直到关闭为止（`clearInterval`），所以一般放在函数体外。
- `setTimeout()`：用于设置某事件的超时，即在设定的时间来到时触发某指定的事件。激活后只会执行一次，所以一般放在函数体内。

Date对象做时钟显示



[查看完整代码](#)

```
<SCRIPT language="JavaScript">
function disptime( ){
  var time = new Date( ); //获得当前时间
  var hour = time.getHours( ); //获
  var minute = time.getMinutes( );
  var second = time.getSeconds( );
  document.myform.myclock.value =hour+":"+minute+":"+second+" ";
  var myTime = setTimeout("disptime()",1000);
}
</SCRIPT>
<BODY onLoad="disptime( )">
<FORM NAME="myform">
H2>当前时间:
  <INPUT name="myclock" type="text" value="" size="10" >
  </H2></FORM >
</BODY>
```

设置文本框的内容
为当前时间

设置定时器每隔1秒
(1000毫秒),调用函数
disptime()执行

小结



编写如左图所示，具有在网页中指定位置显示动态时钟效果的页面。

练习答案

settimeout方法 - Microsoft Internet Explorer

文件(E) 编辑(E) 查看(V) 收藏(A) 工具(T) 帮助(H)

地址(D) D:\JavaScript客户端验证和页面特效制作\JavaScript_chapter2\Pr 转到

[钻石信誉]五区 阴影之刺 联盟600点在线快充 (另有全区点)

一口价: 26.50元

运费: 卖家承担运费

立即购买!

支付宝 此宝贝支持支付宝, 网上汇款免手续费。收货满意后卖家才能拿钱, 货款都安全!

本期售出: 0件

累计售出: 0件

宝贝类型: 全新 所在

宝贝数量: 9999 件 浏览量:

推荐给好友 收藏这件宝贝

当前时间: 10:14:54 AM

自动动态变化的时钟

完毕 我的电脑

任务1:制作随机选号页面



任务描述:



假定班上有**60**名同学，现制作一个提问选号器，如图所示。单击开始按钮在页面随机显示**1-60**的学号，单击停止按钮在页面显示选中学号。



任务分析



实现随机选号页面的制作可以采用以下步骤：

1. 产生1-60之间的随机整数，并在页面显示。
2. 单击开始按钮时使用定时函数隔60毫秒产生一个随机整数。
3. 单击停止按钮时清除定时函数。

Math对象



- **Math**对象能够进行比基本数学运算更为复杂的运算。
- **Math**对象不需要生成对象的实例，可以直接访问它的属性和方法。

Math 对象常见属性和方法



	名称	说明
属性	PI	π 的值，约等于 3.1415
	LN10	10 的自然对数的值，约等于 2.302
	E	Euler 的常量的值，约等于 2.718 。 Euler 的常数是自然对数的底数
<p>Math.random()：产生0—1的随机小数 产生m和n之间的随机数 Math.random()*(n-m)+m</p> <p>Math.round()：四舍五入取整，如9.34 取整为9</p>		
方法	tan (y)	返回 y 的正切，返回值以弧度为半圆
	min (x, y)	返回 x 和 y 两个数中较小的数
	max (x, y)	返回 x 和 y 两个数中较大的数
	random	返回 0-1 的随机数
	round (y)	四舍五入取整
	sqrt (y)	返回 y 的平方根

完成任务二



A software window with a white background and a blue border. In the center is a large, bold, black outline of the number '0'. To the right of the '0' are two buttons: '开始' (Start) and '停止' (Stop). At the bottom left of the window is a yellow bar with a small icon and the text '完成' (Complete). At the bottom right is a blue bar with a small icon and the text '字' (Character).

任务3:制作简单的焦点图效果



任务描述:



焦点图效果是各大网站常用的效果，效果如图所示，页面上**5**张图片**2**秒轮换显示，单击向右图片实现播放下一张图片，图片向后继续**2**秒轮换显示，单击向左图片实现播放上一张图片，图片向前继续**2**秒轮换显示。



任务分析



- 1.设计HTML页面，应用CSS美化页面。
- 2.定义数组，将轮换显示的图片地址保存到数组中。
- 3.定义两个全局变量，一个变量用于控制定时器，另一个变量用于控制数组下标。
- 4.定义函数实现图片的轮换显示。在函数中改变图片的地址，使用定时器函数，2秒更换图片地址，实现图片的轮流显示。
- 5.单击上一张或下一张按钮时将定时器清除，再重新调用图片轮换显示函数。

数组的创建与初始化



```
var MyArray= new Array ( )
```

```
var MyArray = new Array (4)
```

```
var MyArray= new Array ( arg1,arg2,arg3...,argN)
```

第一句声明一个空数组

第二句声明一个长度为4的空数组

第三句声明一个长度为N，并用参数直接初始化数组元素。该方法在实际应用中最为广泛。

数组的创建与初始化



```
var MyArray= [ ]
```

```
var MyArray = [ , , , ]
```

```
var MyArray= [arg1,arg2,arg3...,argN]
```

功能同上一页的创建与初始化方法相同。

数组常用属性和方法



- 常用属性

length : 返回数组中元素的个数

- 常用方法

方法	说明
join	将数组中的元素组合成字符串
reverse	颠倒数组元素的顺序，使第一个元素成为最后一个，而最后一个元素成为第一个
sort	对数组元素进行排序

数组的遍历



- 通过下标访问，遍历数组。

```
for (var i=0;i<MyArray.length;i++){  
    document.write(MyArray[i]);  
}
```

- for (var i in MyArray){
 document.write(MyArray[i]);
}

像遍历一个对象一样遍历数组。

数组的排序



- **reverse()**方法

将数组中元素完全颠倒，返回颠倒后的数组。

MyArray.reverse();

- **sort()**方法

基于某种顺序重新排列数组元素。

MyArray.sort() 按照ASCII码(字母顺序)排序

MyArray.sort(function) 由参数指定排序算法

数组转为字符串



- **join()**方法

将数组中所有元素转化为字符串，并将各元素由逗号隔开合并成一个字符串作为方法的结果返回。

- **join(string)**

由指定的字符串隔开

- **toString()**方法

无须显式声明此方法的调用

数组方法应用



```
<script type="text/javascript">
    var fruit=new Array(5);
    fruit[0]="Apple";
    fruit[1]="Orange";
    fruit[2]="Banana";
    fruit[3]="Peach";
    fruit[4]="Grape";
    document.write("<h3>输出数组: </h3>");
    for(var i=0;i<fruit.length;i++)
    {
        document.write("<h5>第"+i+"个元素是: "+fruit[i]+"<br/></h5>");
    }
    document.write("<h3>反转后的数组: </h3>");
    fruit.reverse();
    for(var i=0;i<fruit.length;i++)
    {
        document.write("<h5>第"+i+"个元素是: "+fruit[i]+"<br/></h5>");
    }
    document.write("<h3>排序后的数组: </h3>");
    fruit.sort();
    for(var i in fruit)
    {
        document.write("<h5>第"+i+"个元素是: "+fruit[i]+"<br/></h5>");
    }
    document.write("<h3>使用连接符连接数组: </h3>")
    var str=fruit.join("-");
    document.write(str);
</script>
```


数组的连接



- `concat()`方法

```
var myNewArray=MyArray.concat(arg1,arg2...,argN)
```

该方法将按照参数的顺序将它们添加到目标数组MyArray的后面，并将最终的结果返回新数组myNewArray

多维数组



- JavaScript中**只有一维数组**
- 实际开发中需要使用多维数组，我们可以利用JavaScript数组存储在数据类型以不同的特性，先将数据分别存储在多个数组中，然后将所有的数组存放在一个一维数组中，即**数组中存放数组**。

任务实现



- 设计静态页面
- 定义数组
- 定义向后轮流播放图片的函数
- 定义向前轮流播放图片的函数
- 定义向前向后“按钮”效果函数

任务4：验证注册页面信息



Firefox

注册页面

file:///E:/教材编写/JavaScript/J 百度

访问最多 新手上路 谷歌 网址大全 游戏中心 360新闻搜索 说课

博客园 遇到技术问题怎么办？
到博客园找查看

新用户注册

用户名： 用户名由英文字母和数字组成的4-16位字符，以字母开头

密码： 密码由4-10位字符组成

确认密码：

电子邮箱：

手机号码：

生日：

String 对象



创建字符串有两种不同方法：

- 使用 var 语句

```
var newstr = “这是我的字符串”
```

- 创建 String 对象

```
var newstr = new String (“这是我的字符串”)
```

String 对象常见属性和方法



	名称	说明
	<p>语法:</p> <p>indexOf(“查找的子字符串” , 查找的起始位置)</p> <p>返回子字符串所在的位置, 如果没找到, 返回 -1</p> <p>例如:</p> <pre>var x var y="abcdefg"; x=y.indexOf("c" , 0); //返回结果为2, 起始位置是0</pre>	
	toLowerCase()	将字符串转换成小写
	toUpperCase()	将字符串转换成大写



获取表单元素的值

```
<SCRIPT LANGUAGE = "JavaScript">
  function checkEmail( ) { var strEmail=document.myform.txtEmail.value;
  if (strEmail.length==0)
    { alert("电子邮件不能为空!");
      return false;    }
  if (strEmail.indexOf("@",0)==-1)
    { alert("电子邮件格式不正确\n必须包含@符号! ");
      return false;    }
  if (strEmail.indexOf(".",0)==-1)
    { alert("电子邮件格式不正确\n必须包含.符号! ");
      return false;    }
    return true;    }
</SCRIPT>
```

返回结果-1表示
没找到“@”字符

表单的提交事件

```
.....
<FORM name="myform" method="post" action="reg_success.htm"
onSubmit="return checkEmail( )"> .....
<INPUT name="registerButton" type="submit" id="registerButton" value=" 注
册 " >
.....
```

练习：完成简单的表单验证




用户名:

输入密码:

再次输入密码:

电子邮件地址: @

Message from webpage

 用户名为空，请输入用户名！



```
<script language="javascript" type="text/javascript">
function getFilepic()
{
    var stxt;
    stxt=form1.pic.value.substring(form1.pic.value.lastIndexOf(".")+1,form1.pic.value.length);
    stxt=stxt.toUpperCase();
    switch(stxt)
    {
        case "GIF":alert("你输入的是GIF文件"); break;
        case "JPG":alert("你输入的是JPG文件"); break;
        case "BMP":alert("你输入的是BMP文件"); break;
        case "PNG":alert("你输入的是PNG文件"); break;
        default:alert("你必须输入GIF,JPG,BMP,PNG等格式的图片!"); break;
    }
}
</script>
...
<form action="" method="post" enctype="multipart/form-data" name="form1" id="form1">
  <p align="center">选择图片:
    <input type="file" name="pic" />
  </p>
  <p align="center">
    <input type="button" value="上传" onclick="getFilepic()" />
  </p>
</form>
```

获取表单元素的值

将字符转化为大写

返回结果-1表示
没找到“@”字符

练习：实现限制上传文件类型



- 限制上传文件只能为doc或txt文件

练习：



- 使用**String**对象方法和**Array**对象方法完成如下图所示效果

请输入你的出生日期： (按xxxx-xx-xx格式输入)

Microsoft Internet Exp...

你的出生年份是：1988
你的出生月份是：10

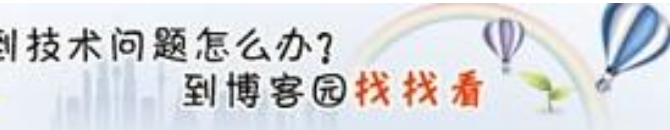
任务5：制作严谨的表单验证页面



博客园



遇到技术问题怎么办?
到博客园找找看



新用户注册

用户名: 用户名由英文字母和数字组成的4-16位字符，以字母开头

密码: 密码由4-10位字符组成

确认密码:

电子邮箱:

手机号码:

生日:

注册完成

正则表达式



- 为什么需要正则表达式
 - 简洁的代码
 - 严谨的验证文本框中的内容

正则表达式



- 在JavaScript脚本中，利用正则表达式可以很容易的实现文本字符串的检测、替换等功能。
- 正则表达式是**字符串**，它定义了一个用来搜索匹配字符串的模式。
- 在WEB开发时，常会判断一个字符串有效性，如一个字符串是否是数字、是否为合法的E-MAIL地址等

定义正则表达式



- 普通方式



语法 **var reg=/表达式/附加参数**



```
var reg=/white/;  
var reg=/white/g;
```

- 构造函数



语法 **var reg=new RegExp("表达式","附加参数")**



```
var reg=new RegExp("white");  
var reg=new RegExp("white","g");
```

RegExp对象的属性



- RegExp对象的属性

属性	描述
global	RegExp对象是否具有标志g
ignoreCase	RegExp对象是否具有标志i
multiline	RegExp对象是否具有标志m

RegExp对象



- RegExp对象的方法

方法	描述
exec	检索字符串中是正则表达式的区配，返回找到的值，并确定其位置
test	检索字符串中指定的值，返回true或false

如何使用正则表达式？



问题

从指定字符串中搜索子字符串“abc”

```
<script language="javascript" type="text/javascript" >
function test1 ()
{  var mystr=document.form1.test.value
  var myreg=new RegExp("abc");
  if(myreg.test(mystr));
    alert("目标字符串中存在子串 'abc'");
  else
    alert("目标字符串中不存在子串 'abc'");
</script>
.....
<form onsubmit="return test1();">
  <input type="text" name="test" value="输入检测的字符串" >
.....
```

创建正则表达式

调用正则表达式的test方法

使用正则表达式的大致步骤：

- 创建正则表达式
- 调用相关对象的方法，得到匹配结果

如何构造正则表达式



- 要精确构造所需的正则表达式不是容易的事，要运用好正则表达式，则须充分了解正则表达式的构造语法。
- 正则表达式的语法主要是对正则表达式各个元字符功能的描述。
- 正则表达式分为简单模式和复杂模式。

表达式的模式



- 简单模式

```
var reg=/china/;
```

```
var reg=/abc8/;
```

- 复合模式

```
var reg=/^\w+$/;
```

```
var reg=/^\w+@\w+.[a-zA-Z]{2,3}(\.[a-zA-Z]{2,3})?$/;
```

简单模式



简单模式包括普通字符、特殊字符、字符类及量词等。

字符类：包括简单类、反向类、范围类、组合类及预定义类

简单类：用方括号 “[]”表示单个字符的“或”的关系，匹配方

括号内任意一个字符。

如： `/[abc]/`可以匹配a、b、c中任意一个字符

反向类： `[^]`用来匹配不在括号内的任意字符。

如： `/[^abc]/`可匹配除a、b、c外的其他任意字符

范围类： `[0-9]`表示匹配从0-9任意的数字

简单模式



字符类：包括简单类、反向类、范围类、组合类及预定义类

组合类：即简单类、反向类和范围类的组合。

如：`/[a-z0-9\.]`可以匹配a-z、0-9和.中任意一个字符

预定义类：用某些特别的表示来表示某些组合类。

正则表达式符号



- 正则表达式常用符号

符号	描述	
/ /	代表一个模式的开始和结束	
限定符	作用	说明
*	出现0次或连续多次	/a*b/可匹配b, aab, aaaab.....
+	出现至少一次	/a+b/可匹配ab, aaab, aaaab.....
?	出现0次或者一次	/a[cd]?/可匹配a, ac, ad
{n}	连续出现n次	/a{3}/相当于aaa
{n, }	连续出现至少n次	/a{3, }/可匹配aaa, aaaa,
{n, m}	连续出现至少n次, 至多m次	/ba{1, 3}/可匹配ba, baa, baaa
.	除了换行符之外的任意字符	

子匹配



- 子匹配是正则表达式语法中的分组概念，在正则表达式中用括号“（）”把一些字符串括起来表示一个子匹配，并按括号顺序编号，同时在字符串匹配时把子匹配结果存储在缓冲区，并作为查询的结果返回。
- 如：`/(ab)c/`中`(ab)`就是一个子匹配，表达式在搜索的时候不仅记录整个表达式的匹配结果，还把子匹配“**ab**”记录在缓冲区，以供查询。

正则表达式的应用



- 用户名、密码、电子邮箱、手机号码、身份证号码、生日、邮政编码、固定电话

ctrip.com
携程旅行网

导购数目最多
赏枫·望水·品花·登高

首页 | 国内酒店 | 海外酒店 | 国内机票 | 国际机票 | 度假 | 特惠

新用户注册

用户名	<input type="text"/>	姓名	<input type="text"/>
密码	<input type="text"/>	地址	<input type="text"/>
确认密码	<input type="text"/>	邮政编码	<input type="text"/>
手机号码	<input type="text"/>	固定电话	<input type="text"/>
电子邮件	<input type="text"/>		

[提交注册信息](#)

中国人发邮件用中国网邮箱
china.com.cn

新用户申请 (带*的为必填选项)

用户名	<input type="text"/>	*	真实姓名	<input type="text"/>	*
密码	<input type="text"/>	*	性别	<input checked="" type="radio"/> 女 <input type="radio"/> 男	
确认密码	<input type="text"/>	*	省/市	北京	
验证码	<input type="text"/>		地址	<input type="text"/>	
密码提示问题	<input type="text"/>	*	邮政编码	<input type="text"/>	
问题的答案	<input type="text"/>	*	生日	<input type="text"/> 年 <input type="text"/> 月 <input type="text"/> 日	
电子邮箱	<input type="text"/>		身份证号码	<input type="text"/>	
手机号码	<input type="text"/>		联系电话	<input type="text"/>	

[提交](#) [重置](#)

中国 版权所有 中国互联网新闻中心

免费注册 | 网站导航 | 宾馆索引 | 代理合作 | 广告业务 | 联系我们

Copyright © 1999-2009, ctrip.com. all rights reserved.

验证邮政编码和手机号码



示例

验证邮政编码和手机号码

分析

1. 中国的邮政编码都是**6**位
2. 手机号码都是**11**位，并且第**1**位都是**1**
3. 邮政编码和手机号码的验证的正则表达式

```
var regCode=/^\d{6}$/;
```

```
var regMobile=/^1\d{10}$/;
```

验证电子邮件地址



电子邮件地址必须含有@和.符号，并且.符号在@符号之后。

```
function checkmail()
{
    var email=document.form1.email.value;
    var reg=/^\w+@\w+\.(com)|(cn)$/;
    if(reg.test(email))
    {
        alert("合法的电子邮件！");
        return true;
    }
    else
    {
        alert("电子邮件格式不正确");
        return false;
    }
}
```

正则表达式
的表示

任务1



- 需求说明

- 使用正则表达式验证博客园注册页面，验证用户名、密码、电子邮箱、手机号码
- 用户名只能由英文字母开头
- 密码只能由英文字母和数字组成
- 生日的年份在1905-04的形式

以

练习1



- 需求说明
 - 登录名只能是中文字符、英文字母、数字及下划线
 - 身份证号码只能由
 - 密码只能由英文字
 - 固定电话不正确，
 - Email包含@和.符

轻松申请, 上门服务

欢迎您申请 SOHO联名信用卡

登录名:

登录密码:

确认密码:

身份证号码:

固定电话:

手机号码:

电子邮件:

提交信息

总结



- 内置对象之一——Date对象
- 内置对象之二——Math对象不需要创建对象实例，可以直接使用。
- setTimeout()方法和clearTimeout()方法。
- String对象的indexOf()方法用于查找子字符串
Array对象常用的属性是length，排序方法：sort
- Date对象有setxxx()方法用于设置日期和时间，
getxxx()方法用于获得日期和时间

总结



- **Date**对象有**setxxx()**方法用于设置日期和时间，**getxxx()**方法用于获得日期和时间
- **Math**对象不需要创建对象实例，可以直接使用，**Math**对象的**random()**方法可以产生0—1的随机数。
- **Array**对象常用的属性是**length**，排序方法：**sort**
- **String**对象的**indexOf()**方法用于查找子字符串
- 使用正则表达式可以制作相对严谨的表单验证



教育改变生活

