/判断输入内容是否为空      
function IsNull(){      
    var str = document.getElementById('str').value.trim();      
    if(str.length==0){      
        alert('对不起，文本框不能为空或者为空格!');//请将“文本框”改成你需要验证的属性名称!      
    }      
}      
     
//判断日期类型是否为YYYY-MM-DD格式的类型      
function IsDate(){       
    var str = document.getElementById('str').value.trim();      
    if(str.length!=0){      
        var reg = /^(\d{1,4})(-|\/)(\d{1,2})\2(\d{1,2})$/;       
        var r = str.match(reg);       
        if(r==null)      
            alert('对不起，您输入的日期格式不正确!'); //请将“日期”改成你需要验证的属性名称!      
        }      
}       
     
//判断日期类型是否为YYYY-MM-DD hh:mm:ss格式的类型      
function IsDateTime(){       
    var str = document.getElementById('str').value.trim();      
    if(str.length!=0){      
        var reg = /^(\d{1,4})(-|\/)(\d{1,2})\2(\d{1,2}) (\d{1,2}):(\d{1,2}):(\d{1,2})$/;       
        var r = str.match(reg);       
        if(r==null)      
        alert('对不起，您输入的日期格式不正确!'); //请将“日期”改成你需要验证的属性名称!      
    }      
}       
     
//判断日期类型是否为hh:mm:ss格式的类型      
function IsTime()       
{       
    var str = document.getElementById('str').value.trim();      
    if(str.length!=0){      
    reg=/^((20|21|22|23|[0-1]\d)\:[0-5][0-9])(\:[0-5][0-9])?$/       
        if(!reg.test(str)){      
            alert("对不起，您输入的日期格式不正确!");//请将“日期”改成你需要验证的属性名称!      
        }      
    }      
}       
     
//判断输入的字符是否为英文字母      
function IsLetter()       
{       
        var str = document.getElementById('str').value.trim();      
        if(str.length!=0){      
        reg=/^[a-zA-Z]+$/;       
        if(!reg.test(str)){      
            alert("对不起，您输入的英文字母类型格式不正确!");//请将“英文字母类型”改成你需要验证的属性名称!      
        }      
        }      
}       
     
//判断输入的字符是否为整数      
function IsInteger()       
{         
        var str = document.getElementById('str').value.trim();      
        if(str.length!=0){      
        reg=/^[-+]?\d\*$/;       
        if(!reg.test(str)){      
            alert("对不起，您输入的整数类型格式不正确!");//请将“整数类型”要换成你要验证的那个属性名称！      
        }      
        }      
}       
     
//判断输入的字符是否为双精度      
function IsDouble(val)       
{       
        var str = document.getElementById('str').value.trim();      
        if(str.length!=0){      
        reg=/^[-\+]?\d+(\.\d+)?$/;      
        if(!reg.test(str)){      
            alert("对不起，您输入的双精度类型格式不正确!");//请将“双精度类型”要换成你要验证的那个属性名称！      
        }      
        }      
}       
     
     
//判断输入的字符是否为:a-z,A-Z,0-9      
function IsString()       
{       
        var str = document.getElementById('str').value.trim();      
        if(str.length!=0){      
        reg=/^[a-zA-Z0-9\_]+$/;       
        if(!reg.test(str)){      
            alert("对不起，您输入的字符串类型格式不正确!");//请将“字符串类型”要换成你要验证的那个属性名称！      
        }      
        }      
}       
     
//判断输入的字符是否为中文      
function IsChinese()       
{       
        var str = document.getElementById('str').value.trim();      
        if(str.length!=0){      
        reg=/^[\u0391-\uFFE5]+$/;      
        if(!reg.test(str)){      
            alert("对不起，您输入的字符串类型格式不正确!");//请将“字符串类型”要换成你要验证的那个属性名称！      
        }      
        }      
}       
     
//判断输入的EMAIL格式是否正确      
function IsEmail()       
{       
        var str = document.getElementById('str').value.trim();      
        if(str.length!=0){      
        reg=/^\w+([-+.]\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*$/;      
        if(!reg.test(str)){      
            alert("对不起，您输入的字符串类型格式不正确!");//请将“字符串类型”要换成你要验证的那个属性名称！      
        }      
        }      
}       
     
//判断输入的邮编(只能为六位)是否正确      
function IsZIP()       
{       
        var str = document.getElementById('str').value.trim();      
        if(str.length!=0){      
        reg=/^\d{6}$/;      
        if(!reg.test(str)){      
            alert("对不起，您输入的字符串类型格式不正确!");//请将“字符串类型”要换成你要验证的那个属性名称！      
        }      
        }      
}       
     
//判断输入的数字不大于某个特定的数字      
function MaxValue()       
{       
    var val = document.getElementById('str').value.trim();      
        if(str.length!=0){      
        reg=/^[-+]?\d\*$/;       
        if(!reg.test(str)){//判断是否为数字类型      
            if(val>parseInt('123')) //“123”为自己设定的最大值      
            {       
                alert('对不起，您输入的数字超出范围');//请将“数字”改成你要验证的那个属性名称！      
            }       
        }      
    }      
}       
     
     
 Phone : /^((\(\d{2,3}\))|(\d{3}\-))?(\(0\d{2,3}\)|0\d{2,3}-)?[1-9]\d{6,7}(\-\d{1,4})?$/      
 Mobile : /^((\(\d{2,3}\))|(\d{3}\-))?13\d{9}$/      
 Url : /^http:\/\/[A-Za-z0-9]+\.[A-Za-z0-9]+[\/=\?%\-&\_~`@[\]\':+!]\*([^<>\"\"])\*$/     
 IdCard : /^\d{15}(\d{2}[A-Za-z0-9])?$/     
 QQ : /^[1-9]\d{4,8}$/     
 某种特殊金额：/^((\d{1,3}(,\d{3})\*)|(\d+))(\.\d{2})?$/               //说明：除“XXX    XX,XXX    XX,XXX.00”格式外

//为上面提供各个JS验证方法提供.trim()属性     
String.prototype.trim=function(){     
        return this.replace(/(^\s\*)|(\s\*$)/g, "");      
    }

调用：  
<input type="text" name="str" >  
<input type="button" value=" 确定 " onClick="">    //onClick中写自己要调用的JS验证函数

<script language="javascript" type="text/javascript">  
var patterms = new Object();  
//验证IP  
patterms.ip = /^(\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])){3}$/;  
//验证EMAIL  
patterms.email = /^[a-zA-Z0-9\_-]+@[a-zA-Z0-9\_-]+(\.[a-zA-Z0-9\_-]+)+$/;  
//验证日期格式2009-07-13  
patterms.date = /^\d{4}-(0?[1-9]|1[0-2])-(0?[1-9]|[1-2]\d|3[0-1])$/;  
//验证时间格式16:55:39  
patterms.time = new RegExp("^([0-1]\\d|2[0-3]):[0-5]\\d:[0-5]\\d$");  
//验证函数  
function verify(str,pat)  
{  
    var thePat;  
    thePat = patterms[pat];  
    if(thePat.test(str))  
    {  
        return true;  
    }  
    else  
    {  
        return false;  
    }  
}  
//测试  
alert(verify("asidycom@163.com","email")+","+verify("192.168.1.1","ip")+  
    ","+verify("16:55:39","time")+","+verify("2009-07-13","date")+","+verify("192.168","ip"));

验证数字：^[0-9]\*$

验证n位的数字：^\d{n}$

验证至少n位数字：^\d{n,}$

验证m-n位的数字：^\d{m,n}$   
  
验证零和非零开头的数字：^(0|[1-9][0-9]\*)$   
  
验证有两位小数的正实数：^[0-9]+(.[0-9]{2})?$   
  
验证有1-3位小数的正实数：^[0-9]+(.[0-9]{1,3})?$   
  
验证非零的正整数：^\+?[1-9][0-9]\*$   
  
验证非零的负整数：^\-[1-9][0-9]\*$   
  
验证非负整数（正整数 + 0） ^\d+$   
  
验证非正整数（负整数 + 0） ^((-\d+)|(0+))$   
  
验证长度为3的字符：^.{3}$   
  
验证由26个英文字母组成的字符串：^[A-Za-z]+$   
  
验证由26个大写英文字母组成的字符串：^[A-Z]+$   
  
验证由26个小写英文字母组成的字符串：^[a-z]+$   
  
验证由数字和26个英文字母组成的字符串：^[A-Za-z0-9]+$   
  
验证由数字、26个英文字母或者下划线组成的字符串：^\w+$   
  
验证用户名或昵称经常用到: ^[\u4e00-\u9fa5A-Za-z0-9-\_]\*$  只能中英文，数字，下划线，减号  
  
验证用户密码:^[a-zA-Z]\w{5,17}$ 正确格式为：以字母开头，长度在6-18之间，只能包含字符、数字和下划线。   
  
验证是否含有 ^%&',;=?$\" 等字符：[^%&',;=?$\x22]+   
  
验证汉字：^[\u4e00-\u9fa5],{0,}$   
  
验证Email地址：^\w+[-+.]\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*$   
  
验证InternetURL：^http://([\w-]+\.)+[\w-]+(/[\w-./?%&=]\*)?$ ；^[a-zA-z]+://(w+(-w+)\*)(.(w+(-w+)\*))\*(?S\*)?$   
  
验证电话号码：^(\(\d{3,4}\)|\d{3,4}-)?\d{7,8}$：--正确格式为：XXXX-XXXXXXX，XXXX-XXXXXXXX，XXX-XXXXXXX，XXX-XXXXXXXX，XXXXXXX，XXXXXXXX。   
  
验证身份证号（15位或18位数字）：^\d{15}|\d{}18$   
  
验证一年的12个月：^(0?[1-9]|1[0-2])$ 正确格式为：“01”-“09”和“1”“12”   
  
验证一个月的31天：^((0?[1-9])|((1|2)[0-9])|30|31)$ 正确格式为：01、09和1、31。   
  
整数：^-?\d+$   
  
非负浮点数（正浮点数 + 0）：^\d+(\.\d+)?$   
  
正浮点数 ^(([0-9]+\.[0-9]\*[1-9][0-9]\*)|([0-9]\*[1-9][0-9]\*\.[0-9]+)|([0-9]\*[1-9][0-9]\*))$   
  
非正浮点数（负浮点数 + 0） ^((-\d+(\.\d+)?)|(0+(\.0+)?))$   
  
负浮点数 ^(-(([0-9]+\.[0-9]\*[1-9][0-9]\*)|([0-9]\*[1-9][0-9]\*\.[0-9]+)|([0-9]\*[1-9][0-9]\*)))$   
  
浮点数 ^(-?\d+)(\.\d+)?$

由于手机号段的不断更新，以前的正则表达式已经无法满足需求。重新编写这条表达式，号段资料来源依据：<http://www.von-line.com/hao.htm>

|  |  |
| --- | --- |
| 1  2  3 | var regex = {      mobile: /^0?(13[0-9]|15[012356789]|18[0236789]|14[57])[0-9]{8}$/  } |

**表达式分析：**  
“/”代表一个正则表达式。  
“^”代表字符串的开始位置，“$”代表字符串的结束位置。  
“?”代表匹配前面的字符一个或零个，所以这里0?的意思是手机号码可以以0开头或不以0开头。  
接下的部分验证11位的手机号码，先从13开始，因为从130-139都有所以可选区间是[0-9]，15开头的号码没有154所以[]里面没有4这个数字，当然也可以写成[0-35-9]，下面18和14开的号码同上。  
小括号括起来的代表一个子表达式，里面是4个可选分支分别用“|”来区分开来，在正则中“|”的优先级是最低的，这里每个分支匹配的都是3个字符（一个[]只能匹配一个字符，里面是可选的意思），也就是手机号码的前3位数字，那么后面还有8位数字需要匹配，可以是0-9的任意字符，所以是“[0-9]{8}”，{}中的数字代表匹配前面字符的个数。分析完毕。