1. 正则表达式

强大的字符串匹配工具

* 1. 字符串操作：

search 查找

substring 获取子字符串

charAt 获取某个字符

slpit 分割字符串变成数组

* 1. RegExp对象

正则默认区分大小写

js风格 new RegExp()

var str='abcde';

var re=new RegExp('a');//是否包含a

alert(re.test(str));

RegExp(‘a’,’i’);//i代表忽略大小写

简写 perl风格 /a/i

var str='abcde';

//var re=new RegExp('a');//是否包含a

var re=/a/i;

alert(re.test(str));

* 1. search

var str='abcde 22 dd 34';

var re=/\d/;

alert(str.search(re));

* 1. 函数

test()测试是否含有

search()寻找位置

match()找到匹配的字符串

* 1. match

var str='aaa e34 55 66g dd 78';

var re=/\d+/g;

alert(str.match(re));

g代表全局 不写默认匹配第一个

* 1. replace替换

var str='abcadA';

alert(str.replace(/a/gi,'T'));

返回替换后的字符串

* 1. 敏感词过滤

var re=/我们|你们|他们/g;

oTxt2.value=oTxt1.value.replace(re,'\*\*');

|表示或

* 1. 方括号

var str='1a2 av cd ';

var re=/1[abc]2/g;

alert(str.match(re));

方括号代表有其中一个

[0-9] 数字

[a-z]字母

[0-9a-z]数字或字母

^a 除了a

[^0-9a-z]除了数字字母

+代表多个

.代表任意(个)字符

\d数字

\w === [a-z0-9\_] word

\s space

\D [^0-9]

\W [^0-9a-z\_]

\S 除了空白字符

{n,m} 最少n次，最多m次

{n,} 最少n次，最多不限

{,m}最少不限，最多m次

{n} 正好n次

\* 0或任意次

? 0次或1次

* 1. 邮箱 /^\w+@[a-z0-9]+\.[a-z]{2,4}$/
  2. 中文 [\u4e00-\u9fa5]