

JavaScript基本对象： 正则表达式对象

豆连军 @八月虎baidu
北京乐美无限科技有限公司

创建正则表达式对象

❑ 对象创建语法

下面两种对象创建方式没有任何差别：

❑ 字面量方式

```
var reExample = /pattern/;
```

❑ 构造函数方式 （Regular Expression）

```
var reExample = new RegExp("ab");//最简单的正则表达式,将匹配字母ab
```

```
var reExample = new RegExp("a","i");//第二个参数,表示匹配时不分大小写，匹配a或者A。
```

- g （全文查找）
- i （忽略大小写）
- m （多行查找）

```
var re = new RegExp("\\w+","gi");//匹配所有的a或A  
等同于：  
var re = /\w+/gi;
```

注意事项

```
var str = "\?";  
alert(str);//只会输出?  
var re = ^?;//将匹配?  
alert(re.test(str));//true  
re = new RegExp("\?");//出错,因为这相当于re = ^?/  
re = new RegExp("\\?");//正确, 将匹配?  
alert(re.test(str));//true
```

使用正则表达式对象

□ 常用方法:

□ exec()

输入一个字符串参数，检查该字符串是否包含了1个或多个正则表达式指定的模版。如果有包含，则返回一个数组；如果没有，则返回null

□ test()

用于检测一个字符串中是否存在某个匹配成功的模式

exec()方法

□ 带“g”选项，可以遍历所有匹配的模式

```
pattern = new RegExp("\\w+","g")
» » RegExp /w+/g
pattern.exec(str)

» » ["hello"]
pattern.exec(str)

» » ["world"]
pattern.exec(str)

» » null
```

```
var pattern = /\w+/g;
var str = "hello,world!";
var result;
while ((result = pattern.exec(str)) != null) {
    alert("Result : " + result + " LastIndex : " +
        pattern.lastIndex);
}
```

<http://www.iteye.com/topic/481228/>

字符串对象直接使用正则表达式

□ String

- String.split()

- String.replace()

- String.search()

- String.match()

示例

- ❑ `\d{5}/.test(55555) -> true`
- ❑ `'paipai'.replace(/i/g, 'p') -> 'papa'`
- ❑ `'#369'.replace(/(\d)/g, '$1$1') -> '#336699'`
- ❑ `'12345'.replace(/d/g, function(match){ return match*2; }) -> '246810'`
- ❑ `'abc'.replace(/w/, function(m){ return m.toUpperCase(); }) -> 'Abc'`
- ❑ `'123abc'.replace(/(\d+)\w+/g, function(match, capture1){ return capture1+'456'; }) -> '123456'`
- ❑ `'120X120'.split('X')[0] -> 120`