

js字符串截取函数slice()、substring()、substr()

在js中字符截取函数有常用的三个slice()、substring()、substr()了，下面我来给大家介绍

slice()、substring()、substr()函数在字符截取时的一些用法与区别吧。取字符串的三个函

数:slice(start,[end]),substring(start,[end])和substr(start,[length]) 相关属性: slice() 第一个参数代表开始位置,第二个参数代表结束位置的下一个位置,截取出来的字符串的长度为第二个参数与第一个参数之间的差;若参数值为负数,则将该值加上字符串长度后转为正值;若第一个参数等于大于第二个参数,则返回空字符串...

在js中字符截取函数有常用的三个slice()、substring()、substr()了，下面我来给大家介绍

slice()、substring()、substr()函数在字符截取时的一些用法与区别吧。

取字符串的三个函数:slice(start,[end]),[substring](#)(start,[end])和substr(start,[length])

相关属性:

slice()

第一个参数代表开始位置,第二个参数代表结束位置的下一个位置,截取出来的字符串的长度为第二个参数与第一个参数之间的差;若参数值为负数,则将该值加上字符串长度后转为正值;若第一个参数等于大于第二个参数,则返回空字符串。

substring()

第一个参数代表开始位置,第二个参数代表结束位置的下一个位置;若参数值为负数,则将该值转为0;两个参数中,取较小值作为开始位置,截取出来的字符串的长度为较大值与较小值之间的差。

substr()

第一个参数代表开始位置,第二个参数代表截取的长度

PS: 字符串都从0开始计起

例子:

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

```
var stmp = "rcinn.cn";
```

```
//使用一个参数
```

```
alert(stmp.slice(3));//从第4个字符开始,截取到最后个字符;返回"nn.cn"
```

```
alert(stmp.substring(3));//从第4个字符开始,截取到最后个字符;返回"nn.cn"
```

```
//使用两个参数
```

```
alert(stmp.slice(1,5));//从第2个字符开始, 到第5个字符; 返回"cinn"
```

```
alert(stmp.substring(1,5));//从第2个字符开始, 到第5个字符; 返回"cinn"
```

```
//如果只用一个参数并且为0的话, 那么返回整个参数
```

```
alert(stmp.slice(0));//返回整个字符串
```

```
alert(stmp.substring(0));//返回整个字符串
```

```
//返回第一个字符
```

```
alert(stmp.slice(0,1));//返回"r"
```

```
alert(stmp.substring(0,1));//返回"r"
```

```
//在上面的例子中我们可以看出slice()和substring()的用法是相同的
```

```
//返回的值也是一样的，但当参数为负数时，他们的返回值却不一样，看下面的例子
```

```
alert(stmp.slice(2,-5));//返回"i"
```

```
alert(stmp.substring(2,-5));//返回"rc"
```

```
//从上面两个例子可以看出slice(2,-5)实际上是slice(2,3)
```

```
//负5加上字符串长度8转换成正3(若第一位数字等于或大于第二位数字,则返回空字符串);
```

```
//而substring(2,-5)实际上是substring(2,0),负数转换为0,substring总是把较小的数作为起始位置。
```

```
alert(stmp.substring(1,5))//从第2个字符开始，到第5个字符；返回"cinn"
```

```
alert(stmp.substr(1,5));//从第2个字符开始，截取5个字符；返回"cinn."
```

```
</script>
```

substr 和 substring方法的区别

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

```
var str = "0123456789";//
```

```
alert(str.substring(0));//-----"0123456789"
```

```
alert(str.substring(5));//-----"56789"
```

```
alert(str.substring(10));//-----""
```

```
alert(str.substring(12));//-----""
```

```
alert(str.substring(-5));//-----"0123456789"
```

```
alert(str.substring(-10));//-----"0123456789"
```

```
alert(str.substring(-12));//-----"0123456789"
```

```
alert(str.substring(0,5));//-----"01234"
```

```
alert(str.substring(0,10));//-----"0123456789"
```

```
alert(str.substring(0,12));//-----"0123456789"
```

```
alert(str.substring(2,0));//-----"01"
```

```
alert(str.substring(2,2));//-----""
```

```
alert(str.substring(2,5));//-----"234"
```

```
alert(str.substring(2,12));//-----"23456789"
```

```
alert(str.substring(2,-2));//-----"01"
```

```
alert(str.substring(-1,5));//-----"01234"
```

```
alert(str.substring(-1,-5));//-----""
```

```
alert(str.substr(0));//-----"0123456789"
alert(str.substr(5));//-----"56789"
alert(str.substr(10));//-----""
alert(str.substr(12));//-----""
alert(str.substr(-5));//-----"0123456789"
alert(str.substr(-10));//-----"0123456789"
alert(str.substr(-12));//-----"0123456789"
alert(str.substr(0,5));//-----"01234"
alert(str.substr(0,10));//-----"0123456789"
alert(str.substr(0,12));//-----"0123456789"
alert(str.substr(2,0));//-----""
alert(str.substr(2,2));//-----"23"
alert(str.substr(2,5));//-----"23456"
alert(str.substr(2,12));//-----"23456789"
alert(str.substr(2,-2));//-----""
alert(str.substr(-1,5));//-----"01234"
alert(str.substr(-1,-5));//-----""
</script>
```

函数: split()

功能: 使用一个指定的分隔符把一个字符串分割存储到数组

例子:

```
str="jpg|bmp|gif|ico|png";
arr=theString.split("|");
//arr是一个包含字符值"jpg"、"bmp"、"gif"、"ico"和"png"的数组
```

函数: join()

功能: 使用您选择的分隔符将一个数组合并为一个字符串

例子:

```
var delimitedString=myArray.join(delimiter);
var myList=new Array("jpg","bmp","gif","ico","png");
var portableList=myList.join("|");
//结果是jpg|bmp|gif|ico|png
```

功能: 返回字符串中匹配子串的第一个字符的下标

```
var myString="JavaScript";
var w=myString.indexOf('v');w will be 2
```

```
var x=myString.indexOf("S");x will be 4
```

```
var y=myString.indexOf("Script");y will also be 4
```

```
var z=myString.indexOf("key");z will be -1
```

在网上看到另一种非常简单的方法，代码如下：

```
function func(s, n) {
```

```
    return s.replace(/([^\x00-\xff])/g, "$1a").slice(0, n).replace(/([^\x00-\xff])a/g, "$1");
```

```
}
```