



传智播客 · 前端与移动开发学院

<http://web.itcast.cn>

前端基本功—javascript 第七天

目录

目录.....	2
1.1 复习.....	3
1.2 根据字符返回位置.....	3
1.2.1 返回前面起第一个字符的位置.....	4
1.2.2 返回后面起第一个字符的位置.....	4
1.3 网址编码.....	5
1.4 操作字符串.....	6
1.4.1 concat() 连接字符串.....	6
1.4.2 slice().....	6
1.4.3 substr().....	7
1.5 保留小数位数.....	8
1.6 大小写转换.....	8
1.7 无缝滚动.....	9
1.8 缓动动画.....	10

1.1 复习

1. 按钮不可用 `disabled = "disabled" || true`
2. `setTimeout` 只执行一次 `setInterval` 执行很多次
3. 递归调用 : 函数自己调用自己 函数递归调用 不提倡用函数名
而喜欢用的是 `arguments.callee` 返回的是 正在执行的函数 本身

4. 逻辑运算符 与 `&&` 或 `||` 非 `!` `!> && > ||`

5. `2&&0` `0` `1&&8` `0&&3` `0` `1||0`

6. 转换为 字符型

```
var num = 12345;
```

```
num+ ""
```

```
String(num);
```

```
num.toString(); 10 进制
```

```
num.toString(2) 二进制
```

7. 根据字符位置返回字符

```
charAt(索引号)
```

```
var txt = "abcde";
```

```
txt.charAt(3); d
```

```
charCodeAt(3) unicode 编码 97 100
```

1.2 根据字符返回位置

跟 `charAt()` 相反的 根据 字符 返回的是 位置

1.2.1 返回前面起第一个字符的位置

```
indexOf(“字符”);
```

它是从 前面开始数（从左边开始数），而且只找第一个，然后返回改字符的位置，索引号都是从 0 开始的。返回的是个数值。

```
var txt = “abcdef”;
```

```
alert(txt.indexOf(“d”))
```

 结果就是 3

如果找不到该字符 返回 -1

1.2.2 返回后面起第一个字符的位置

lastIndexOf(参数: 索引字符串)

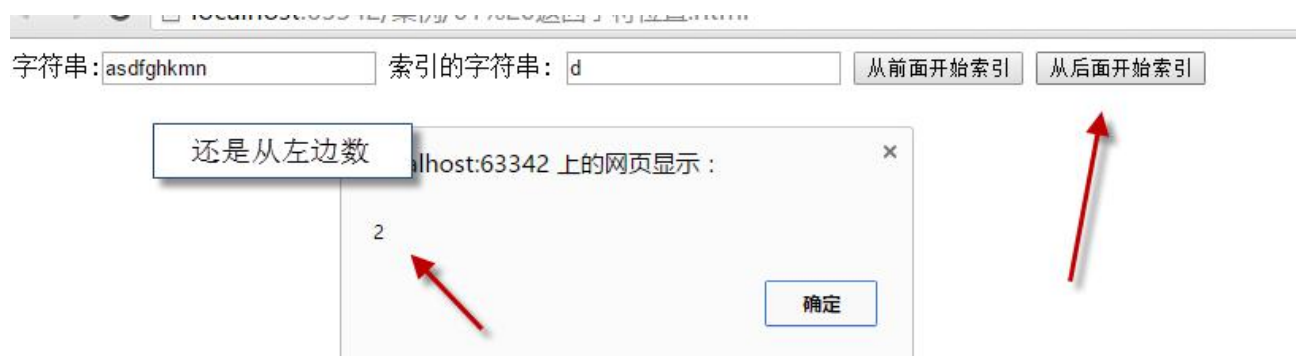
从后面开始数 同上

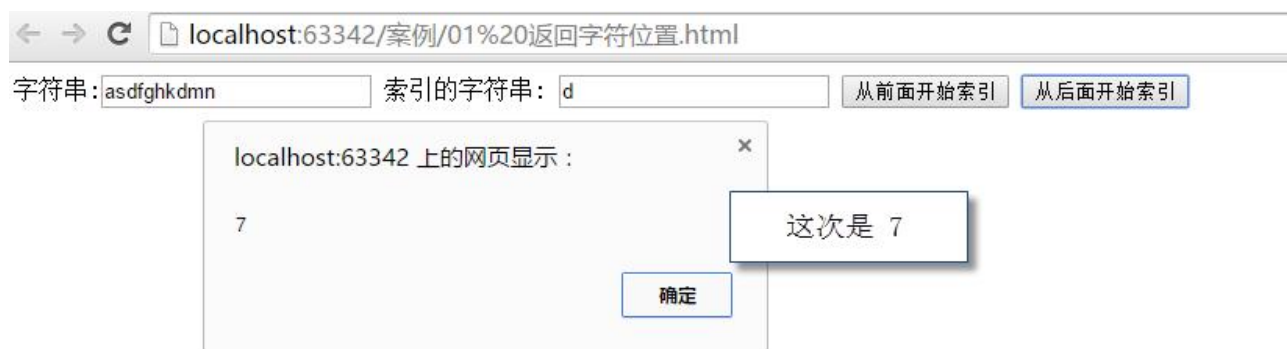
```
var txt = “abcdef”;
```

```
txt.lastIndexOf(“d”);
```

 3

返回的值，还是从 左边开始 数的 索引号 。





1.3 网址编码

我们知道一个网址 自己的网址， 不同页面也有自己 id 网址， 我们经常会做一些， 把网址送入到后台。 但是后台再处理的 不认识比如 换行啊 等特殊符号的 ？

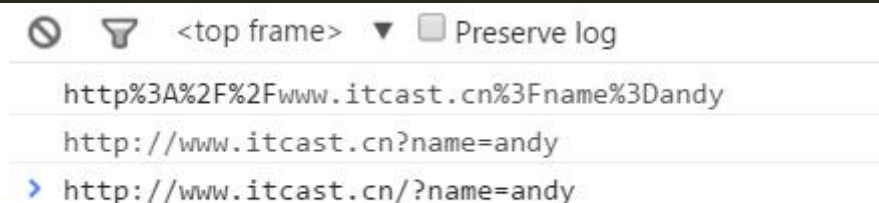
```
var url = "http://www.itast.cn?name=cz"
```

所以我们要实现编码，然后再传到后台。

`encodeURIComponent()` 函数可把字符串作为 URI 组件进行编码

`decodeURIComponent()` 函数可把字符串作为 URI 组件进行解码

```
var url = "http://www.itcast.cn?name=andy";  
console.log(encodeURIComponent(url)); // 编码  
var afterUrl = encodeURIComponent(url);  
console.log(decodeURIComponent(afterUrl)); // 解码
```



结果：

1.4 操作字符串

合并字符串 取字符串

1.4.1 concat() 连接字符串

```
var txt1 = "abc";  
var txt2 = "123";  
console.log(txt1.concat(txt2));    "abc123";
```

1.4.2 slice()

slice("取字符串的起始位置", [结束位置]); [] 可选的

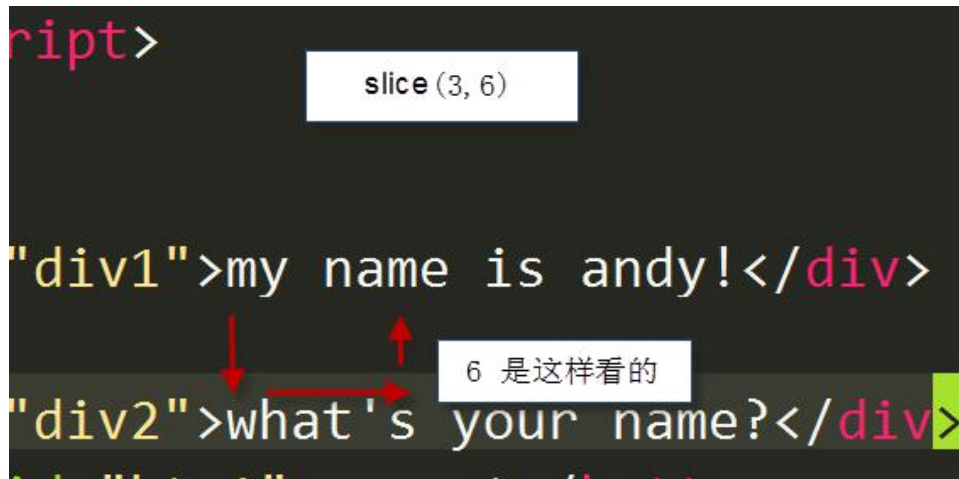
起始位置一定要有 结束位置可以省略

```
var txt = "abcdef";
```

txt.slice(3); 从 txt 里面字符的 第 3 (索引号) 个开始取 结束位置省略, 一直取到最后一个。

总结 slice(3) 从第 3 个开始取, 一直取到最后

slice(3,6) 3 从 第 3 个开始 取 6 取到第 6 索引号的位置, 还是从左边的第 0 个开始数。 但是不包 6 。



起始位置可以是负数，如果是负数，则是从右边往左边开始取。

```
var txt = "asdf";
```

```
txt.slice(-1) 结果是 f
```

1.4.3 substr()

substr(起始位置,[取的个数])

同上。

不写取的个数，默认从起始位置一直取到最后。

取的个数：是指从起始位置开始，往后面数几个。

```
var txt = "abcdefghijk";
```

```
txt.substr(3,4);
```

从第 3 个（d）开始 数 4 个 defg

substr(-1) 少用 ie678 报错。尽量少用

兼容性的写法：

```
onBtnClick("btn7",div1.substr(div1.length-1,1)); // 兼容的写法
```

substring 同 slice 一样的 但是有一点不同

substring(3,6)

substring 始终会把 小的值作为 起始位置 大的值作为结束位置

例如: substring(6,3) 实际中 自动变成 substring(3,6);

1.5 保留小数位数

122340.12345 保留两位有效小数 122340.12

substr(0, .+3)

```
1. console.log(str.substr(0, str.indexOf(".") + 3));
```

通过 indexOf 返回小数点的位置 截取字符串

```
2 console.log(parseInt(Pi*100) / 100);
```

先乘以 100 取整 然后 除以 100

```
3 console.log(Pi.toFixed(2));
```

pi.toFixed(2) 保留 2 位 小数

1.6 大小写转换

asdf = ASDF

ASDF -- asdf

1 toUpperCase, 转换为大写 (参数: 无)

2 toLowerCase, 转换为小写 (参数: 无)

asdf.toUpperCase() 结果就是: ASDF

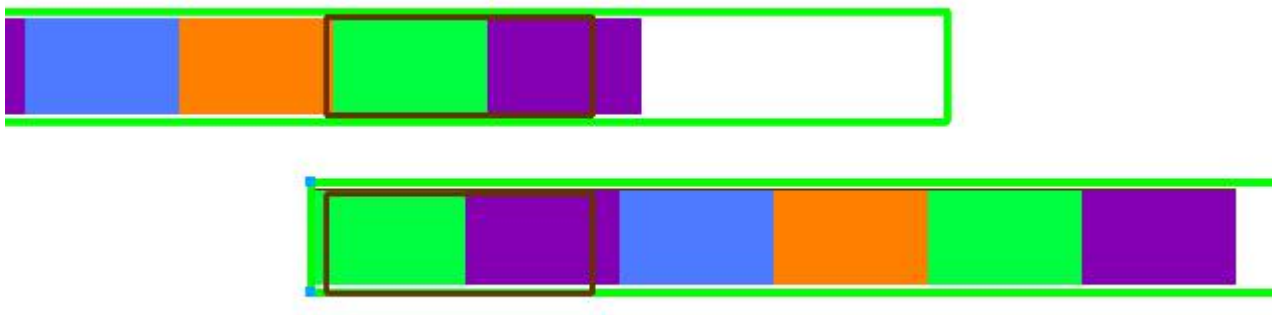

```
$("txt").value.toUpperCase();
```

把 txt 的值 转换为大写

1.7 无缝滚动

原理：

首先先复制 两张图片 （第一张和第二张） 放到最后面 （本质上是第 5.6 张）



ul 是盒子移动的， 如果 ul 的 left 值 大于等于 4 张图片的宽度，就应该快速复原为 0 。

完整代码：

```
1  <script>
2      var scroll = document.getElementById("scroll");
3      var ul = scroll.children[0];
4      var num = 0; //控制左侧值 left
5      var timer = null; // 定时器
6      timer = setInterval(autoPlay,20);
```

```
7      function autoPlay() {  
8          num--;  
9          //console.log(num);  
10         num<=-1200 ? num = 0 : num;  
11         ul.style.left = num + "px";  
12     }  
  
13     scroll.onmouseover = function() { // 鼠标经过大盒子 停止定时器  
  
14         clearInterval(timer);  
15     }  
  
16     scroll.onmouseout = function() {  
  
17         timer = setInterval(autoPlay,20); // 开启定时器  
  
18     }  
19 </script>
```

1.8 缓动动画

匀速动画，看起来，效果太简单。

等差序列 : 2 4 6 8 10



缓动动画公式:

一个盒子初始值 是 0 要走到 400 px 的位置

假如说, 初始值 leader 0 target 400

`box.style.left = xxxx + "px"`

`leader = leader + (target - leader) / 10;`

```
1 btn.onclick = function() {  
2     setInterval(function() {  
3         leader = leader + (target - leader) / 10;
```

```
4         box.style.left = leader + "px";  
5     },30)  
6 }
```

