

# element.childNodes怎么数个数

一开始接触到这个子节点个数的时候，每次调用length出现的数字总和我数的不一样。在网上查找了一圈解释都是一脸懵逼，只好自己自己动手试验了。

- 第一回合

先写一个块元素

`<div></div>` 这个块元素有1个子节点，子节点类型为3（文本类型）。

这样

`<div>dvdv</div>`

这样

`<div>dvdv`

`</div>`、

还是这样

`<div>`

`dvdvdv`

`</div>`

childNodes都只有1个文本类型的子节点。

- 第二回合

现在开始仔细数

`<div>dvdv<p>p1</p></div>`这样的话在原来的div元素里又添加了一个p的元素节点所以childNodes.length是2

但是这样

`<div>dvdv<p>p1</p>`

`</div>`

在p元素的结束标签后面加上换行，div元素的childNodes.length就会多一个文本类型的子节点。

这样

`<div>dvdv`

`<p>p1</p>`

`</div>`

和上面的是一样的。

- 第三回合

`<div>dvdv<p>p1</p><p>p2<span>span1</span></p></div>`

子节点数为3 ----蒙了吧。（div元素有两个p1和p2并列的子元素节点和一个本身的文本节点，至于p2

里的span元素，它不是和p1，p2并列的元素，所以不能算是div的子节点，可能是孙子节点。而且p1节点的末尾标签后面没有换行，p2节点的末尾标签后面也没有换行所以是---一个div本身的文本节点，和两个元素节点一共3个）。

\*\*\*\*现在加上换行

```
<div>dvdv
  <p>p1</p><p>p2</p>
</div>
```

这样的话节点数就变为了4个啊，首先dvdv后面有个换行这个换行是不计数的，一般情况下换行会被计为一个文本节点，dvdv就是一个文本节点所以dvdv后面的换行是不作为文本节点计数的。比上个例子多出来的一个节点就是p2便签末尾的那个换行，记为文本节点。

\*\*\*\*最后来个实战

```
<div>
  <p>p1</p>
  <p id = 'p2'>
    <span>---span---</span>
  </p>
</div>
```

来看看这个div块共有几个子节点

首先div本身有个文本节点（不管它会不会像上面的例子那样有显示的dvdv文本），接着p1一个元素节点，p1标签末尾的换行是又一个文本节点，还有id为p2的元素节点，以及p2末尾的换行又是一个文本节点，至此div的子节点已经数完了。一共5个子节点，至于p2内部的span那是div的孙子节点了不能算在div的子节点里的。

- 如果标签的嵌套违反了嵌套规则，那么在计算子节点的时候可能怎么算都算不对的（：

