

HTML textarea cols,rows属性和宽度高度关系研究

这篇文章发布于 2016年02月4日, 星期四, 00:53, 归类于 CSS相关。 阅读 40831 次, 今日 11 次 13 条评论

by zhangxinxu from <http://www.zhangxinxu.com>

本文地址: <http://www.zhangxinxu.com/wordpress/?p=5240>

一、关于textarea元素的cols和rows属性

`<textarea>` 元素, 俗称“文本域”, 或者“多行文本框”, 其自带原生的HTML属性 `rows` 表示行的意思, 可以改变 `<textarea>` 的可视区域高度, `cols` 表示列, 可以改变 `<textarea>` 的可视区域宽度。例如:

```
<textarea></textarea>
<textarea cols="30" rows="5"></textarea>
```

会发现, 下面的尺寸明显比没有 `cols` 和 `rows` 属性值的要大。

下面问题来了, 我想很多小伙伴知道 `cols` 和 `rows` 可以影响文本域的尺寸, 那 `cols` 和 `rows` 不同的数值和最终展示的像素尺寸之间有没有什么关系呢? 可不可以通过公式计算呢? 最终表现是否还受到其他CSS属性的影响呢?

我想, 以上的问题就很少有人知道了, 包括我在内, 在写此文之前, 也是模模糊糊的了解。

下面要扯点废话了, 写这篇文章的时候, 我其实内心是孤独的。这种感觉和自己一人去荒野寻找钓场的经历很像, 你知道前方有一条河流, 至于那里适不适合钓鱼, 是不知道的, 你要知道结果, 就需要亲自去确认。但是, 前往河岸的道路之前没有一个人走过, 也就是没有道路, 而拦在自己面前的是齐人高的荒草地, 它们是那么高那么远, 似乎看不到头, 你不知道草地中有什么危险, 毒蛇野猪随时可能出现。然而, 追随自己的热情的内心, 还是毅然前去开拓者未知的道路, 草木长得是那么的密集, 自己只能用手用力拨开他们, 狠狠踩在脚下才能前行, 长着倒刺的茎藤, 刀片般的叶子, 在手上脸上留下一个个伤痕, 当你前进了一半, 到了整个荒草片野中间的时候, 四周没有一个人, 只有轻轻的风抚动草叶交织的嗦嗦声, 一股孤独的感觉油然而生, 在这一刻, 再强大的内心也会犹豫, 我要不要继续前进……

对啊, 我要不要继续前进呢! 最近一段时候, 有一句话对我内心冲击很大, 而且反复被提及, 就是“你做这件事情的目的是什么?”

“拜托, 不要搞这么功利, 别搞这么累好不好, 我就是追寻内心, 我就是兴趣使然!”我心里的第一反应是这样的。



然而，当突然抬头，发现周围没有一个人，满是荆棘的时候，内心就会犹豫，好比这篇文章，如果非要讲功利，讲目的，真的没什么，有价值吗？有人在意吗？花出去的时间值得吗？为什么搞些无人问津的东西？为何不去关注时髦的对项目有帮助的东西？为何别人在天上飞，你要在荆棘丛中孤独前行呢？（下面视频：路人在街头黑板上写下他们这辈子最大的遗憾）



So, 统统out去吧，我继续我的前进，穿越未曾穿越过的草地，前往那召唤等待的河流，发现自己向往的垂钓处女地。

二、cols属性值和宽度

您可以狠狠地点击这里：[textarea文本域cols属性和宽度关系demo](#)

界面如下：

我们可以选择下拉修改文本域的文字尺寸，字体以及字符间距等，看看会不会对文本域的宽度造成影响。结果结论如下：

Chrome
IE
FireFox
font-size
✓
✓
✓

font-family
×
×
✓
letter-spacing
×
✓
✓

可以看见，Chrome浏览器只受到字符大小影响，对字体和字符间距视而不见；而Firefox浏览器则每一个都会影响之；而IE浏览器不受字符间距影响。

那最关心的问题来了，`cols` 的值和最终呈现的宽度有没有什么比如的关系或者公式呢？

在simsum字体，也就是宋体下，我们可以很容易得看出之间的关系为：

- **Chrome:** $8\text{px} * \text{cols} + 17\text{px}$
- **IE:** $8\text{px} * \text{cols} + 17\text{px}$
- **FireFox:** $8\text{px} * \text{cols} + 29\text{px}$

由于CSS,HTML这些东西是老外发明的，因此，`cols` 每个单位相对的宽度是相对于英文字符而言的，在宋体下，可以认为是 `8px`，后面的 `17px` 其实很好理解，表示滚动条的宽度。在window系统下，默认，所有浏览器的滚动条所占据的宽度都是17像素。所以，对于Chrome和IE浏览器而言，文本域最终的宽度（`padding` 和 `border` 尺寸暂不考虑）就是单字符宽度* `cols` 值+滚动条宽度。然而，FireFox浏览器下，增加的宽度是 `29px`，这个不就搞不清楚问什么是 `29px` 了。

在IE浏览器下，如果 `<textarea>` 的 `overflow` 属性值为 `hidden`，则没有后面增加的 `17px`，而其他浏览器没有此现象。

然后，当我们使用其他字体的时候，最终的宽度表现则比较微妙了。因此，单个 `cols` 对应的单位就不是整数了，可能是 `8.1px` 也有可能是 `7.7px~7.8px` 这样子的。

总结一下就是：

貌似在Chrome浏览器下才有点实用价值，因为其他浏览器下的宽度波动太明显，任何其他环境的变化都会导致宽度不一样，这个其实在网页布局中是很糟糕的。因为，网页是自上而下流式的，高度可以很长很长，但是，宽度一般是固定的，资源比较紧张，要求布局精确，因此，一旦出现宽度在各个浏览器下不一样的情况，势必大大影响其实用价值。因此，在实际web制作的时候，除非一些对宽度要求不高的情况，其余时候，还是使用CSS `width` 属性来控制文本域的宽度。

三、rows属性值和高度

您可以狠狠地点击这里：[textarea文本域rows属性和高度关系demo](#)

界面如下：

我们可以选择下拉修改文本域的文字尺寸，字体以及行高等，看看会不会对文本域的高度造成影响。结果结论如下：

Chrome
IE
FireFox

font-size
✗
✓
✗
font-family
✗
✓
✗
line-height
✓
✗
✓

可以看见，Chrome浏览器和FireFox浏览器只受到行高大小影响，对字体和字符大小视而不见；而IE浏览器却是完全相反的处理，对 `line-height` 行高视而不见，但是，却对字符大小和字体有影响。

那最关心的问题来了，`rows` 的值和最终呈现的高度有没有什么比如的关系或者公式呢（先不考虑 `padding` 和 `border` 的影响）？

在Chrome浏览器下，最终的高度就是 `rows` 的值和 `line-height` 行高的乘积。正如上面的demo截图所示，`rows` 值为 2，`line-height` 为 20px 的时候，最后的高度是 40px，就是他们的乘积。

在FireFox浏览器下也是类似的，但是有一点不同，就是高度需要再把滚动条的尺寸计算上去，因此，是下面这幅光景，高度都比Chrome浏览器高 17px：



在IE浏览器下，高度与行高没关系，更是有文本的 `content area` 高度(inline box模型中的概念)决定的，因为当不同字体切换的时候，高度也会变高，例如，使用微软雅黑字体就会变高很明显，而微软雅黑的 `content area` 高度就是比一般字体要高。因此，要想确定文本域高度其与字符之间的关系，关键很难确定那个计算系数：高度=rows*系数。例如，16px 的微软雅黑字体的系数近似 21（见下图），宋体simsun则是18.2~18.5之间等。



总结一下就是：

兼容性还是一团糟，没有任何两个浏览器是完全同一表现。但是最终表现设计要比 `cols` 要好那么一点点。更关键的问题是，高度这个东西一般要求没那么严，所以，我们有时候不想重新搞个CSS确认文本域的高度，使用 `rows` 属性值搞一下也是可以的。

四、最后的结束语

最后研究的结果是，兼容性很差，想要各个浏览器下表现一模一样成本很高。因此，如果对文本域的布局要求比较严格，还是使用CSS `width` / `height` 属性精确控制。

唯一的收获就是，知道了各个浏览器如何兼容性不一，以及最终尺寸呈现的计算方法，相关因素等，说不定，对于日后某个功能实现提供了不一样的思路呢！

En, that's all!



(本篇完) // 想要打赏? 点击[这里](#)。有话要说? 点击[这里](#)。



« CSS font关键字属性值的简单研究

CSS镂空图片transition过渡初加载背景色块问题解决 »

猜你喜欢

- CSSOM视图模式(CSSOM View Module)相关整理
- canvas HTML属性尺寸和CSS尺寸多个细节深入
- textField - jQuery文本域操作集插件展示
- div模拟textarea文本域轻松实现高度自适应
- 基于HTML模板和JSON数据的JavaScript交互
- 有意思: textarea resize属性下纯CSS交互效果
- CSS流体(自适应)布局下宽度分离原则
- 翻译: CSS中的槽粕
- 理解CSS3 max/min-content及fit-content等width值
- 深入理解CSS的width:auto
- 内容loading加载后高度变化CSS3 transition体验优化

分享到: [+](#) [微信](#) [QQ](#) [微博](#) [豆瓣](#) [贴吧](#) [收藏](#) [0](#)

标签: cols, height, rows, textarea, width

发表评论 (目前13条评论)

名称 (必须)

邮件地址(不会被公开) (必须)

网站

提交评论

1. 南山老么说道:
2016年10月27日 20:12
为精神点个赞
[回复](#)



2.

妙莲说道:

2016年04月19日 10:08

非常感恩您把知识分享给我们这样学识还很浅薄的初学者, 让我不断进步, 在学完的基础知识, 不知道怎么走下去的时候, 找到了您的文章。如获珍宝...

回复

Joe说道:

2016年11月17日 09:11

En, that's right!

回复


3.

wingmeng说道:

2016年03月9日 19:12

你的分享令人耳目一新, 拓展了思路 and 眼界, 你用自己的实际行动使同行很多人受益匪浅, 你我他, 我们都记得你这位兴趣使然的英雄, 心怀感激! 为你点赞!

另外, 不要停, 不要停, 请继续.....前进

回复


4.

exkaede说道:

2016年02月26日 10:12

喜欢你的这句话, 我就是兴趣使然

我有时候也会突然想起来有个东西总是模模糊糊, 就会去搞清楚, 不过, 写博客就没你那么有才华啦

回复


5.

伊利鑫牛奶说道:

2016年02月23日 22:50

感觉学CSS这会子百度啥迷糊的问题都会首页list鑫哥的博客=。=list多了就记得了=。=前来拜读哈哈喜欢生活小鸡汤加技术干货的文风

回复


6.

晴雯说道:

2016年02月23日 15:57

赞!!!

回复


7.

zj说道:

2016年02月22日 09:16

布满荆棘, 仍毅然向前, 赞一个

回复


8.

yangzj1992说道:

2016年02月15日 16:34

为兴趣使然手动点赞

回复


9.

dzxczx说道:

2016年02月13日 18:34



加油！

[回复](#)

10. **Asiaidc.net**说道：
2016年02月4日 10:50

已拜读,对于浏览器的兼容性问题在每个开发者的心中都是个永远的伤痛~

[回复](#)



11. **meepo**说道：
2016年02月4日 10:46

兴趣使然。赞

[回复](#)



12. **Humphry**说道：
2016年02月4日 09:51

作为替换元素的textarea等，其实在CSS2.1内部就规定了其内容区域不在CSS的管辖范围内，所以看天气看系统看浏览器看心情~

顺带一提，对于这类替换元素，chrome可以用-webkit-appearance: none;来试着兼容一下.....

关于对表单元素的兼容，mozilla有写过，可以参考一下~：https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/Forms/Advanced_styling_for_HTML_forms#Dealing_with_the_select_nightmare

[回复](#)



最新文章

- » [常见的CSS图形绘制合集](#)
- » [粉丝群第1期CSS小测点评与答疑](#)
- » [分享三个纯CSS实现26个英文字母的案例](#)
- » [小tips: 纯CSS实现打字动画效果](#)
- » [CSS/CSS3 box-decoration-break属性简介](#)
- » [CSS :placeholder-shown伪类实现Material Design占位符交互效果](#)
- » [从天猫某活动视频不必要的3次请求说起](#)
- » [CSS vector-effect与SVG stroke描边缩放](#)
- » [CSS ::backdrop伪元素是干嘛用的？](#)
- » [周知：CSS -webkit-伪元素选择器不再导致整行无效](#)

今日热门

- » [常见的CSS图形绘制合集](#) ⁽¹⁹⁰⁾
- » [未来必热：SVG Sprite技术介绍](#) ⁽¹¹⁹⁾
- » [粉丝群第1期CSS小测点评与答疑](#) ⁽¹¹⁵⁾
- » [HTML5终极备忘大全（图片版+文字版）](#) ⁽⁹³⁾
- » [让所有浏览器支持HTML5 video视频标签](#) ⁽⁸⁶⁾
- » [Selectivizr-让IE6~8支持CSS3伪类和属性选择器](#) ⁽⁸²⁾
- » [CSS3下的147个颜色名称及对应颜色值](#) ⁽⁷⁹⁾
- » [视区相关单位vw, vh..简介以及可实际应用场景](#) ⁽⁷⁶⁾
- » [写给自己看的display: flex布局教程](#) ⁽⁷⁶⁾
- » [小tips: 纯CSS实现打字动画效果](#) ⁽⁷⁶⁾



今年热议

- » 《CSS世界》女主角诚寻靠谱一起奋斗之人 ⁽⁷⁶⁾
- » 不借助Echarts等图形框架原生JS快速实现折线图效果 ⁽⁶⁴⁾
- » 看，for..in和for..of在那里吵架！ ⁽⁶⁰⁾
- » 是时候好好安利下LuLu UI框架了！ ⁽⁴⁷⁾
- » 原来浏览器原生支持JS Base64编码解码 ⁽³⁵⁾
- » 妙法攻略：渐变虚框及边框滚动动画的纯CSS实现 ⁽³³⁾
- » 炫酷H5中序列图片视频化播放的高性能实现 ⁽³¹⁾
- » CSS scroll-behavior和JS scrollIntoView让页面滚动平滑 ⁽³⁰⁾
- » windows系统下批量删除OS X系统.DS_Store文件 ⁽²⁶⁾
- » 写给自己看的display: flex布局教程 ⁽²⁶⁾

猜你喜欢

- CSSOM视图模式(CSSOM View Module)相关整理
- canvas HTML属性尺寸和CSS尺寸多个细节深入
- textField - jQuery文本域操作集插件展示
- div模拟textarea文本域轻松实现高度自适应
- 基于HTML模板和JSON数据的JavaScript交互
- 有意思：textarea resize属性下纯CSS交互效果
- CSS流体(自适应)布局下宽度分离原则
- 翻译：CSS中的糟粕
- 理解CSS3 max/min-content及fit-content等width值
- 深入理解CSS的width:auto
- 内容loading加载后高度变化CSS3 transition体验优化