

# 移动端全兼容的flexbox速成班

恬恬

说起flexbox，都算是件陈年旧事了，它是2009年W3C提出的一种全新的可伸缩的CSS布局方式。依赖flexbox，我们可以更简单，高效的完成可伸缩式页面的布局。

业界与flexbox的相关教程文章也是各式各样，新旧交替，很多小伙伴对flexbox想用又不敢用，究其原因也就是即分不清它各个版本的编写规范，又苦恼于大家总是挂在嘴边的一句“flexbox兼容性不好”。所以今儿前来总结一个精华干货贴，回顾那些楼主在项目里用过的flexbox，来谈谈那些不用苦恼“兼容”的flexbox最佳实例（本文只聊移动端）。

## Part1 先聊聊历史：

在2009年最早版本的Flexbox规范中，我们编写为“display:box;”，

中期版本的Flexbox；我们编写为“display:flexbox;”

而目前的规范版本，我们定义“display:flex;”。flexbox规范的制定可谓是艰辛百变，

但目前使用的最新[版本草案](#)已于今年3-1日正式进入W3C的[候选推荐标准（Candidate Recommendation）](#)阶段，这也就意味着关于flexbox的最终规范即将落地。

## Part2 版本VS兼容：

如下图所示，除了Opera mobile12，移动端的各大浏览器都是支持flexbox的旧版语法的，但不包含flex的wrap属性。

## Part3 信手拈来的flexbox最佳实例

(以下代码片段均遵循flexbox移动端全兼容性写法，已验证机型列表见文章最后) 我们先来回顾一下flexbox的属性体系：直白的解释，我们常说的“flexbox”其实包含“父元素”，“子元素”2个部分，将“父元素”定义为一个flexbox，则在"父元素"里的“子元素们”就被赋予了可以自由伸缩的能力。 赋予神奇能力的代码片段如下：

```
33▼ .flex-cont{
34     /*定义为flexbox的"父元素"*/
35     display: -webkit-box;
36     display: -webkit-flex;
37     display: flex;
38 }
39▼ .flex-item{
40     /*给"子元素"赋予自由伸缩的能力*/
41     -webkit-box-flex: 1;
42     -webkit-flex: 1;
43     flex: 1;
44▼ }
```

1.用flex做提示icon

这里用到了flex父元素的“align-items”属性。 指定元素沿侧轴对齐方式 **align-item: flex-start | flex-end | center | space-between | space-around | stretch;**



一句属性设置，就可以完成顶部对齐，居中对齐或是底部对齐的切换，简单方便值得拥有。

【Demo Link】：<http://jsfiddle.net/tikizzz/2zuthdap/>

2.用flex做列表元素

同样，只需要用flex父元素的“align-items”属性，就可以制作灵活多变的列表元素了。如下图所示，3种状态的变形都不需要修改CSS文件，只需要按需隐藏DOM结构就搞定。

这里多注意1个Tips：像下图例子中的左侧图片，右侧按钮这些固定宽度的子元素，不需要增加“flex:1”的能力，维持原样即可。



【Demo Link】：<https://jsfiddle.net/tikizzz/ztdfq5dw/>

### 3.用flex做tab

做均分的tab，应该是flexbox最常见的一个功能了，实现原理很简单，只需要给“父元素”，“子元素”分别赋予“display：flex”，"flex：1"即可。子元素的宽度不会根据内容的长短而发生改变。如下图所示，前台同学增加或删减tab数量，只需要增减DOM结构即可，无需样式的修改。

3.用flex做tab			
title	title	long title	title
long title	title	title	

★重点兼容TIPS: 在旧版的规范中，使用比例伸缩布局时，子元素的内容长短不同会导致无法“等分”，这个时候，我们需要给子元素设置一个“width:0%”来解决问题。 ★重点兼容TIPS: 不要给flexbox里的子元素设置“margin:auto”的属性，在部分安卓机下，它会导致该元素的宽度撑开到100%占位

【Demo Link】：<https://jsfiddle.net/tikizzz/ztdfq5dw/>

4.用flex做导航(只适合三项的布局) 一样也是利用align-items的属性即可，就可以轻松完成flexbox的导航制作。

4.用flex做导航(只适合三个布局)		
btn	The big title information	btn
btn	The big title	btn

但是导航变成单按钮布局的时候，会导致标题栏的位位移，不是特别的推荐。

btn	The big title information
btn	The big title

【Demo Link】：<https://jsfiddle.net/tikizzz/g2Lj417p/>

5.用flex做搜索条 利用align-items的属性，还可以轻松完成flexbox的搜索条制作。



【Demo Link】 :<https://jsfiddle.net/tikizzz/yut2qv9b/>

## 6.用flex做垂直居中(单/多)

前面说的5个实例，其实都只用到了flexbox的基础属性+“align-items”属性。千万不要以为结束了，flexbox还可以做更多的事。例如一直很困扰我们的“垂直居中”：flexbox可以轻松实现不定宽高，不限数量的元素“水平+垂直居中”在屏幕中。

这部分用到了flexbox的另外一个属性：指定元素沿主轴对齐方式 **justify-content: flex-start | flex-end | center space-between | space-around;** 将“沿着主轴对齐方式+沿着侧轴对齐的方式”设置为居中，无论子元素是什么形态，都可以随时随地的“水平垂直居中”了。

【Demo Link】 :<https://jsfiddle.net/tikizzz/zq8cdkfg/>

## 7.用flex做垂直弹性布局

顶部栏，底部栏fixed，中间的元素支持滚动条，这是移动端常见的页面结构模型，我们熟称为“垂直弹性布局”。使用传统的fixed写法总是会给一些安卓机带来无法避免的烦人bug。其实只要巧妙利用flexbox的转换方向的属性，就可以轻松实现这个结构模型了。如下图，其实也就是一个横着的弹性伸缩模型，倒转垂直的弹性伸缩模型了。

这部分用到了flexbox的方向属性：指定主轴的伸缩流方向 **flex-direction: row | column** ; 这里注明一下，box-orient, box-direction是最老版本flex-box控制方向的2种写法，默认方向均为横向，为保证兼容性，我们需要将它们2条都写全。

```
.page .flex-ver{
  /*指定主轴的伸缩流方向是纵向的*/
  -webkit-box-orient:vertical;
  -webkit-box-direction: normal;
  -webkit-flex-direction: column;
  flex-direction: column;
  width: 100%;
  height: 100%;
}
```

【Demo Link】 :<https://jsfiddle.net/tikizzz/obLp1mga/>

最后，附赠一个全DEMO的合集：<http://115.159.36.96/tikizheng/flextest/demo.html>



## Part4 其他

希望以上的demo对移动端开发的小伙伴们有用。当然，flexbox还有一些很帅气的属性，例如order，wrap等等，它们只是暂时还不被移动端所有的系统兼容，咋们的x5内核已经刚刚表示全面兼容flexbox的所有属性。关于flexbox的未来我们敬请期待就好！ 另，附赠 <<[移动端属性全兼容自查表](#)>>

### 已测机型

#### 备注：

- 1.以上机型是根据友盟，应用宝，微信及腾讯云运营活动数据统计整合得出的Top7机型。
- 2.Top8-10的华为， oppo，魅族机型挑选于各品牌的占比最高机型，为品牌差异性测试而存在。
- 3.黄色标注的分辨率为Top6的热门分辨率， 占比超过iOS市场87%， Android市场的50%（安卓分辨率太多，剩余50%分辨率占比较为均分零散，故暂不列入必测范围）。
- 4.iOS8.0+， Android4.0+涵盖了移动端90%的系统，其中iOS9.0+占比超过65%， Android4.4+占比超过60%，测试用例不强行要求涵盖各机型所有版本的系统，以最新版本为准，若因老版本出现bug次数大于3次，再加设为必测的问题版本。

Tags: [CSS3](#), [flexbox](#), [h5布局](#)

