如图所示的布局我们经常见到，传统的方法是写上float || display: inline-block来实现，现在我们尝试引入CSS3 flex新布局方式，相比较传统的定位、浮动方式，flex提供了更强的灵活性，但是由于规范一直在变动，直到12年才成标准，在此之前还是存在一些浏览器支持旧的规范的，所以这里梳理一下这些知识，更好的在移动端应用这些技术。

## HTML

首先编写 HTML：

<div class="wrapper">

<div class="box"><a href="" class="link">box1</a></div>

<div class="box"><a href="" class="link">box2</a></div>

<div class="box"><a href="" class="link">box3</a></div>

</div>

## 一、float layout

float是我们再熟悉不过的css语法了，给一个元素设置float那么它就脱了文档流，自身的宽度随内容大小而定，因此会带来的一系列问题比如要清除浮动（有关float的细节请查看[W3C SPEC](http://www.w3.org/TR/CSS21/visuren.html" \l "floats)，这里就不多说了）

.wrapper {

min-width: 320px;

.clearfix();

}

.box {

float: left;

width: 33.3%;

}

## 二、inline-block layout

使用inline-block避免了float脱离文档流的影响，但是会受到字体行高的影响导致两个元素之间存在3px左右的空隙，有很多方法解决了，比如修改HTML结构。

.box {

display: inline-block;

width: 33.3%;

}

## 三、table

## 四、css3 flex

flex自出现开始，规范就一直在变动：

Flexbox Specification Timeline:

* July 2009 Working Draft (display: box;)
* March 2011 Working Draft (display: flexbox;)
* November 2011 Working Draft (display: flexbox;)
* March 2012 Working Draft (display: flexbox;)
* June 2012 Working Draft (display: flex;)
* September 2012 Candidate Recommendation (display: flex;)

所以存在早期浏览器支持旧规范的情况，在兼容旧的规范之前，先解释一下现在处于“CR”的规范。

### CR SPEC

我们知道现在有block layout, inline layout, table layout, position layout，而flex对应的则是flex layout(弹性布局)。简言之我们只要知道以下几个属性就知道如何使用flex了：

// 设置 Flex Formatting Context

display: flex || inline-flex

// flex-direction 主轴方向

flex-direction: row || row-reverse || col || col-reverse

// flex-wrap 是否折行显示

flex-wrap: wrap || no-wrap(default)

// flex-flow 上面两个的简写属性

flex-flow: floex-direction flex-wrap;

// 对齐方式

justify-content:

align-items:

align-self:

align-content:

// 弹性程度

flex-grow:

flex-shrink:

flex-basis:

// flex 上面三个的简写属性

flex: flex-grow flex-shrink flex-basis

demo就不写了，这里有一个[自动生成flexbox的工具](http://the-echoplex.net/flexyboxes/)，可以体验一下上面几个属性的用法。

### 浏览器的兼容

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 21+ (new) | 3.1+ (old) | 2-21 (old) |  | 10 (tweener) | 2.1+ (old) | 3.2+(old) |
| 20- (old) | 6.1+ (new) | 22+ (new) | 12.1+ (new) | 11+ (new) | 4.4+ (new) | 7.1+(new) |
| **Chrome** | **Safari** | **Firefox** | **Opera** | **IE** | **Android** | **iOS** |

PS：

* old对应display: box
* tweener对应display: flexbox
* new对应display: flex

所以从上表可以看出，一般我们写上old, tweener, new就基本上能覆盖掉大多数的浏览器：

.wrapper {

display: -webkit-box;

display: -moz-box;

display: -ms-flexbox;

display: -webkit-flex;

display: flex;

}

.box {

-webkit-box-flex: 1;

-webkit-flex: 1;

flex: 1;

}

## 五、css3 column

## 六、css3 grid