



黑客派

# 关于列表布局的样式分析

作者: [zjhch123](#)

原文链接: <https://hacpai.com/article/1496934012885>

来源网站: [黑客派](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<h2 id="1--前言">1. 前言</h2>

<script async src="https://pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>

<!-- 黑客派PC帖子内嵌-展示 -->

<ins class="adsbygoogle" style="display:block" data-ad-client="ca-pub-5357405790190342" data-ad-slot="8316640078" data-ad-format="auto" data-full-width-responsive="true"></ins>

<script>

(adsbygoogle = window.adsbygoogle || []).push({});

</script>

<p>在实际项目开发中，比较常见的是如下图这样的列表布局。<br><br>列表布局的好处有很多，如排列整齐，可以很好的聚合同类物品，如果再加上同一色调或者相互搭配的图片，会给人很清爽的感觉。<br></p>

<h2 id="2--布局方式">2. 布局方式</h2>

<p>列表布局的方式有很多，我目前在项目中大量使用的方式有 2 种。<br>一种是使用 <code>display: inline-block;</code> 来达到列表布局的目的。<br>还有一种是使用 <code>float: left / right</code> 来做到列表布局。</p>

<p>更早些的全局 <code>&lt;table/&gt;</code> 的方式目前见的比较少，而当下使用最新的 <code>CSS3</code> 的 <code>display: flex</code> 在一些主流网站上也可以看见其身影。<br>今我来说说前两种方式。</p>

<h3 id="2-1-基于的列表布局">2.1 基于 <code>display: inline-block</code> 的列表布局</h3>

<p><code>block</code> 的意思是块级元素，一个 <code>div</code> 默认的 <code>display</code> 属性值就是 <code>block</code>，它会撑满一整行。<br>而 <code>inline-block</code> 则不同，它会将元素改变为<strong>行内元素</strong>。<code>display: inline-block</code> 的元素会与其他带有 <code>inline</code> 属性的元素一起构成<strong>行</strong>其<strong>行高</strong>由<strong>行内最高的元素</strong>决定。<br>因此，利用其\*\* "行\*\*"的特性，我们可以使用其进行列表布局。<br>如文章开头的那张图片，就是使用 <code>inline-block</code> 制作的列表布局。<br><a href="https://link.hacpai.com/forward?goto=http%3A%2F%2F139.129.132.196%2Fshow%2Ftable%2Finline-block.html" target="\_blank" rel="nofollow ugc">大家可以点击我，查看列表布局。</a><br>可以看出代码里有一点奇怪的地方，在父元素 <code>.g-wrap</code> 内定义了一个奇怪的属性</p>

```
<pre><code class="highlight-chroma">letter-spacing: -30px;
font-size: 0;
```

```
</code></pre>
```

<p>子元素内也相对的定义了</p>

```
<pre><code class="highlight-chroma">letter-spacing: 0;
font-size: 16px
```

```
</code></pre>
```

<p>这是为了去除 <code>inline-block</code> 元素之间的间隙而写的。如果不这样的话各个元之间会空出一定的间隙。<br><a href="https://link.hacpai.com/forward?goto=http%3A%2F%2F139.129.132.196%2Fshow%2Ftable%2Finline-block2.html" target="\_blank" rel="nofollow ugc">大家可以点击我，查看列表布局的间隙。</a><br>点击 <code>加上letter-spacing!</code> 按钮之后，整体布局会发生变化，此时点击 <code>去掉letter-spacing!</code>，可以看出明显的不同。<br>这是因为 <code>inline-block</code> 元素在书写的过程中，元素间的空格/空行导致的。<br>除了使用给父类加 <code>letter-spacing</code> 和 <code>font-size: 0</code> 的方式之外，我们还可以用更简单粗暴的形式来解决：<br><br>我们将代码书写到一行，元素间的间距自然就消失了！<br>此时我们可以查看效果，会发现<br><br>达到了我们想要的效果。<br><a href="https://link.hacpai.com/forward?goto=http%3A%2F%2F139.129.132.196%2Fshow%2Ftable%2Finline-block2.html" target="\_blank" rel="nofollow ugc">大家可以点击我，查看列表布局的间隙。</a>

block3.html" target="\_blank" rel="nofollow ugc">大家可以点击我，查看效果图。</a><br> 而前使用的打包工具均具有压缩代码的功能，可以将所有的 HTML 代码都压缩为一行，所以这个问题实不是很大。</p>

<p>PS: 上文其实可以只使用 <code>font-size: 0</code> 来使元素间距消失。但是在 <code>chrome</code> 中貌似设置了最小字体大小为 <code>12px</code>，所以一般还是使用 <code>letter-spacing: 负值</code> 的方式来消除间距。</p>

<p>因为 <code>inline-block</code> 的"行"的特性，其可以很方便的将高度不一样的元素整理为表布局而不会破坏整体。<br> <br> 我们也可以使用 <code>vertical-align</code> 属性来控制对齐方式。例如 <code>vertical-align: middle</code><br> </p>

<h3 id="2-2-2-基于float的列表布局">2.2.2 基于 float 的列表布局</h3>

<script async src="https://pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>

<!-- 黑客派PC帖子内嵌-展示 -->

<ins class="adsbygoogle" style="display:block" data-ad-client="ca-pub-5357405790190342" data-ad-slot="8316640078" data-ad-format="auto" data-full-width-responsive="true"></ins>

<script>

(adsbygoogle = window.adsbygoogle || []).push({});

</script>

<p><code>float</code> 布局和 <code>inline-block</code> 在普通的效果上差不多，代码差也不大，就是得记住浮动会引起 <code>高度塌陷</code>，要 <code>清除浮动</code>。关于 <code>清除浮动</code>，网络上有许多方式方法，比方说父元素 <code>overflow: hidden; zoom: </code>，也有给父元素加 <code>:after {clear: both; content: ""}</code>，用哪种都无所谓，忘了清除就可以。<br> <br> <a href="https://link.hacpai.com/forward?goto=http%3A%2F%2F139.129.132.196%2Fshow%2Ftable%2Ffloat.html" target="\_blank" rel="nofollow ugc">大家可以点击我，查看 float 布局</a><br> 而 <code>float</code> 最大的缺陷就是不支持高度不同的元素一起排列。这也就限制了列表元素必须定高。<code>定高</code> 这种做法我是相当反感的，除非策划能保证将来不改需求，否则还是远离固定高度这个大坑比较好。<br> <br> 如果不定高，换言之，有元的高度超过了其他元素，会造成整体布局混乱。<br> <br> <a href="https://link.hacpai.com/forward?goto=http%3A%2F%2F139.129.132.196%2Fshow%2Ftable%2Ffloat2.html" target="\_blank" rel="nofollow ugc">大家可以点击我，查看混乱的局</a></p>

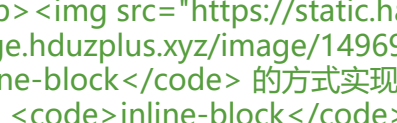
<h2 id="3--如何选择">3. 如何选择</h2>

<p>实际项目中，我们必须根据相关视觉稿来确定我们使用的布局方式。<br> 比如说最简单的元素高定宽列表布局，我们可以直接使用 <code>float</code> 布局的方式直接完成。<code>float</code> 也无需考虑元素间隙，就是不能忘记清除浮动。<br> 然而实际中元素定高定宽的列表布局可是之又少呀！因为谁也不知道下一个超出宽度的元素会是哪个，除非策划能保证将来不改需求。<br> <br> 实际上我比较推荐的还是 <code>inline-block</code> 的方式。<br> 其一、使用 <code>inline-block</code> 可以远离 <code>float</code> 来的高度塌陷问题；<br> 其二、<code>inline-block</code> 的本职工作就是如此，<code>float</code> 的本职工作只是将图片/文字浮动而已；<br> 其三、<code>inline-block</code> 支持化的高度，更能适应当下快节奏的变更需求；<br> 其四、<code>inline-block</code> 配上父元素 <code>text-align: justify</code> 可以实现完美的自适应的两端对齐布局，而这一切都是浏览器帮完成的。（<a href="https://link.hacpai.com/forward?goto=http%3A%2F%2F139.129.132.196%2Fshow%2Ftable%2Finline-block4.html" target="\_blank" rel="nofollow ugc">大家可以点击我查看完美的自适应两端对齐布局</a>）</p>


## 4. 案例

我们通过几个小案例来分析一下使用场景

### 4.1 案例一

 这是最简单的列表布局。推荐使用 `nine-block` 的方式实现。因为其内部元素的文案、标志等极有可能发生更替而导致换行。用 `inline-block` 可以有效避免这点。就不上代码了~

### 4.2 案例二

 类似这样的列表布局的变种，我比较倾向于 `float` 布局。因为看图片的样式可以发现，右边的两块高度之和一定等于左图可以评估一下改需求的可能性不大。我们可以写出这样的代码很轻松的完成布局。

```
<pre><code class="highlight-chroma">&!doctype html&
&html&
&head&
&meta charset="utf-8"&
&style&
* {
  box-sizing: border-box;
}
.g-wrap {
  margin: auto;
  width: 900px;
}
.g-wrap:after {
  display: table;
  content: "";
  clear: both;
}
.m-rect {
  float: left;
  width: 300px;
  border: 1px solid red;
  height: 300px;
}
.m-srect {
  height: 150px;
}
&/style&
&/head&
&body&
&div class="g-wrap"&
  &div class="m-rect"&&/div&
  &div class="m-rect"&&/div&
  &div class="m-rect m-srect"&&/div&
  &div class="m-rect m-srect"&&/div&
&/div&
&/body&
&/html&
</code></pre>
<script async src="https://pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></scr
pt>
<!-- 黑客派PC帖子内嵌-展示 -->
<ins class="adsbygoogle" style="display:block" data-ad-client="ca-pub-5357405790190342"
```

```
data-ad-slot="8316640078" data-ad-format="auto" data-full-width-responsive="true"></in
>
<script>
  (adsbygoogle = window.adsbygoogle || []).push({});
</script>
<p></p>
<h3 id="4-3-案例3">4.3 案例 3</h3>
<p><br> 黑客派的首页的这块地方比较有意思。使
开发者工具查看之后发现是左边 3*3 的方块单独布局，右边的 list 和下方的两块方格单独布局。而且
使用了 display: flex 的方式，估计是为了能更好的适配移动端。<br> 如果只是需
适配设置最小宽度的 pc 端，我们或许可以好好利用 float 的特性来布局。</p>
<pre><code class="highlight-chroma">&lt;!doctype html&gt;
&lt;html&gt;
&lt;head&gt;
&lt;meta charset="utf-8"&gt;
&lt;style&gt;
* {
  box-sizing: border-box;
}
.g-wrap {
  width: 90%;
  min-width: 792px;
  margin: auto;
  background-color: gray;
}
.u-timeline {
  float: right;
  width: 40%;
  height: 492px;
  border: 1px solid black;
}
.u-timeline .ul {
  margin: 0;
  padding: 0;
  height: 100%;
  width: 100%;
  list-style: none;
}
.u-list {
  letter-spacing: -30px;
  font-size: 0;
}
.u-list .rect {
  letter-spacing: 0px;
  font-size: 16px;
}
.rect {
  letter-spacing: 0;
  height: 246px;
  width: 20%;
  display: inline-block;
  vertical-align: top;
```

```
background-color: rgba(100,100,100,.6);
border: 1px solid red;
}
```

```
&lt;/style&gt;
```

```
&lt;/head&gt;
```

```
&lt;body&gt;
```

```
&lt;div class="g-wrap"&gt;
```

```
&lt;div class="u-timeline"&gt;
```

```
&lt;ul class="ul"&gt;
```

```
&lt;li&gt;1&lt;/li&gt;
```

```
&lt;li&gt;2&lt;/li&gt;
```

```
&lt;li&gt;3&lt;/li&gt;
```

```
&lt;li&gt;4&lt;/li&gt;
```

```
&lt;li&gt;5&lt;/li&gt;
```

```
&lt;li&gt;6&lt;/li&gt;
```

```
&lt;li&gt;7&lt;/li&gt;
```

```
&lt;/ul&gt;
```

```
&lt;/div&gt;
```

```
&lt;div class="u-list"&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;div class="rect"&gt;&lt;/div&gt;
```

```
&lt;/div&gt;
```

```
&lt;/div&gt;
```

```
&lt;/body&gt;
```

```
&lt;/html&gt;
```

```
</code> </pre>
```

```
<script async src="https://pagead2.googlesyndication.com/pagead/js/adsbygoogle.js"></script>
```

```
<!-- 黑客派PC帖子内嵌-展示 -->
```

```
<ins class="adsbygoogle" style="display:block" data-ad-client="ca-pub-5357405790190342" data-ad-slot="8316640078" data-ad-format="auto" data-full-width-responsive="true"></ins>
```

```
<script>
```

```
(adsbygoogle = window.adsbygoogle || []).push({});
```

```
</script>
```

```
<p></p>
```

```
<p>PS: 当然了，黑客派的首页不可能像我这样粗糙的布局，这里只是简单的提出一种思路，要想适应更多分辨率或许需要 js 来辅助布局，以及使用其他更现代化的适配方式，这里就不再赘述了。</p>
```