**一、CSS定位的工作原理**

默认情况下，块级元素的内容宽度是其父元素的宽度的100％，并且与其内容一样高。内联元素高宽与他们的内容高宽一样。不能对内联元素设置宽度或高度——它们只是位于块级元素的内容中。 如果要以这种方式控制内联元素的大小，则需要将其设置为类似块级元素 display: block;。

**页面**[**布局技术**](https://developer.mozilla.org/zh-CN/docs/Web/CSS/Layout_mode)**的细节：**

* 浮动
* 定位
* CSS 表格
* 弹性盒子
* 网格

**正常的布局流**：是指在不对页面进行任何布局控制时，浏览器默认的HTML布局方式。

要使某个元素上的特定类型的定位，使用[position](https://developer.mozilla.org/zh-CN/docs/Web/CSS/position)属性。

**定位技术**

定位技术(position)允许我们将一个元素从它在页面的原始位置准确地移动到另外一个位置。

**有四种主要的定位类型：**

静态定位(Static positioning)是每个元素默认的属性——它表示“将元素放在文档布局流的默认位置——没有什么特殊的地方”。

相对定位(Relative positioning)允许我们相对元素在正常的文档流中的位置移动它——包括将两个元素叠放在页面上。这对于微调和精准设计(design pinpointing)非常有用。

用[top](https://developer.mozilla.org/zh-CN/docs/Web/CSS/top), [bottom](https://developer.mozilla.org/zh-CN/docs/Web/CSS/bottom), [left](https://developer.mozilla.org/zh-CN/docs/Web/CSS/left), 和 [right](https://developer.mozilla.org/zh-CN/docs/Web/CSS/right) 来精确指定要将定位元素移动到的位置。

（需要考虑一个看不见的力，推动定位的盒子的一侧，移动它的相反方向。 例如，如果你指定 top: 30px;一个力推动框的顶部，使它向下移动30px。）

绝对定位(Absolute positioning)将元素完全从页面的正常布局流中移出，类似将它单独放在一个图层中. 我们可以将元素相对于页面的 <html> 元素边缘固定，或者相对于离元素最近的被定位的祖先元素(ancestor element)。绝对定位在创建复杂布局效果时非常有用，例如通过标签显示和隐藏的内容面板或者通过按钮控制滑动到屏幕中的信息面板.

元素的位置改变——top，bottom，left和right以不同的方式在绝对定位。解析为：位于从“包含元素”的顶部\*\*px，从左边\*\*px。

包含元素：默认情况下，它是[<html>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/html)元素——定位的元素是被嵌套在[<body>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/body)中的HTML源代码，但在最终的布局，它离页面边缘的顶部和左侧30px距离，这是[<html>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/html)元素。 这更准确地称为元素的**定位上下文**。

z-index：当元素开始重叠时，决定哪些元素出现在其他元素的顶部。

z-index 值影响定位元素位于该轴上的位置；正值将它们移动到堆栈上方，负值将它们向下移动到堆栈中。默认情况下，定位的元素都具有z-index为auto，实际上为0。

z-index只接受无单位索引值；你不能指定你想要一个元素是Z轴上23像素—— 它不这样工作。 较高的值将高于较低的值，这取决于您使用的值。 使用2和3将产生与300和40000相同的效果。

固定定位(Fixed positioning)与绝对定位非常类似，除了它是将一个元素相对浏览器视口固定，而不是相对另外一个元素。 在创建类似页面滚动总是处于页面上方的导航菜单时非常有用。

**二、弹性盒子**

详细：http://www.runoob.com/w3cnote/flex-grammar.html

灵活实现：

* 在父内容里面垂直居中一个块内容。
* 使容器的所有子项占用等量的可用宽度/高度，而不管有多少宽度/高度可用。
* 使多列布局中的所有列采用相同的高度，即使它们包含的内容量不同。

指定元素的布局为 flexible：

section {

display: flex;

}

**三、css布局**

在网页中，元素有三种布局模型：  
1、流动模型（Flow） 默认的  
2、浮动模型 (Float)  
3、层模型（Layer）

内容：

1、display

2、margin: auto;

例子：

#main {

width: 600px;

margin: 0 auto;

}

<div id="main">

设置块级元素的 width 可以防止它从左到右撑满整个容器。然后你就可以设置左右外边距为 auto 来使其水平居中。元素会占据你所指定的宽度，然后剩余的宽度会一分为二成为左右外边距。唯一的问题是，当浏览器窗口比元素的宽度还要窄时，浏览器会显示一个水平滚动条来容纳页面。

3、max-width

例子：

#main {

max-width: 600px;

margin: 0 auto;

}

<div id="main">

在这种情况下使用 max-width 替代 width 可以使浏览器更好地处理小窗口的情况。这点在移动设备上显得尤为重要，调整下浏览器窗口大小检查下吧！

4、box-sizing：当你设置一个元素为 box-sizing: border-box; 时，此元素的内边距和边框不再会增加它的宽度。

把以下CSS代码放在他们页面上，确保所有的元素都会用这种更直观的方式排版：

\* {

-webkit-box-sizing: border-box;

-moz-box-sizing: border-box;

box-sizing: border-box;

}

4、position

5、float

6、clear：被用于控制浮动。

7、清除浮动（clearfix hack）





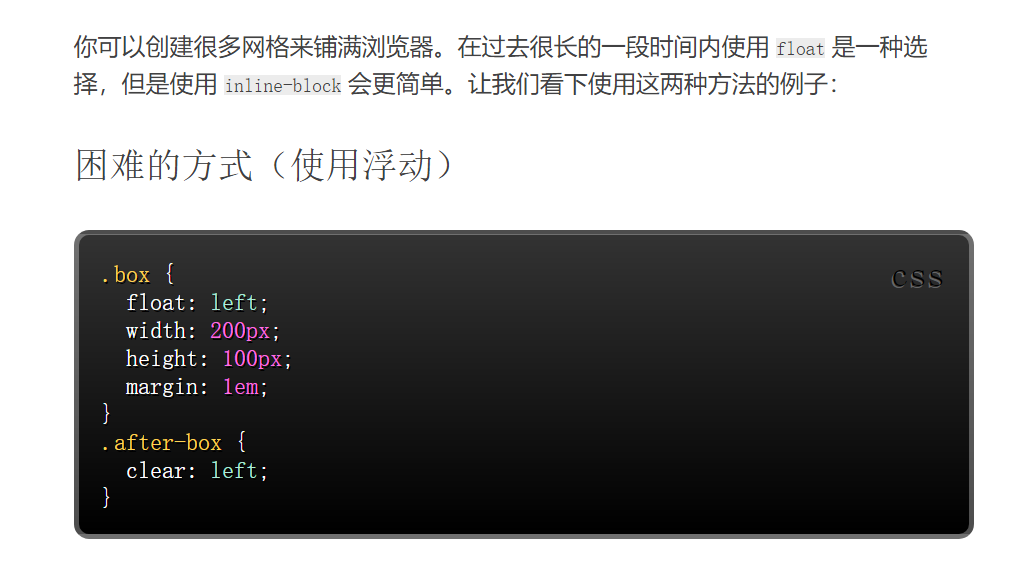


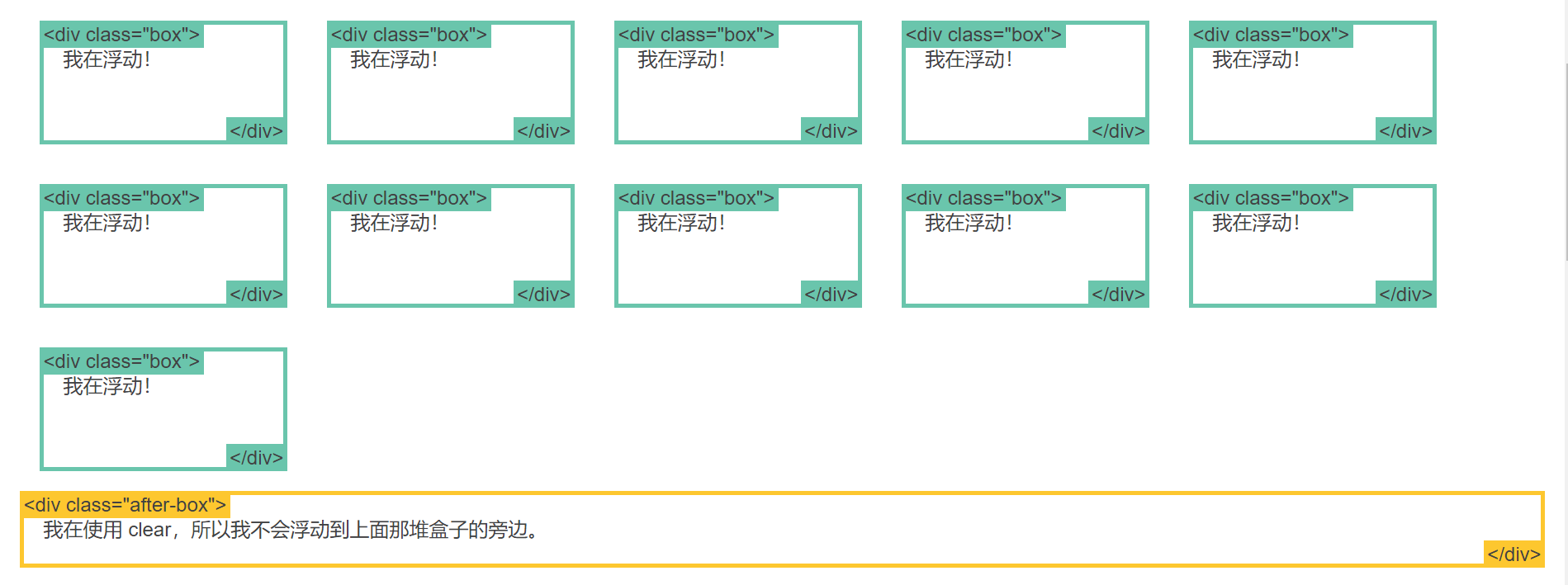
8、百分比宽度

9、媒体查询

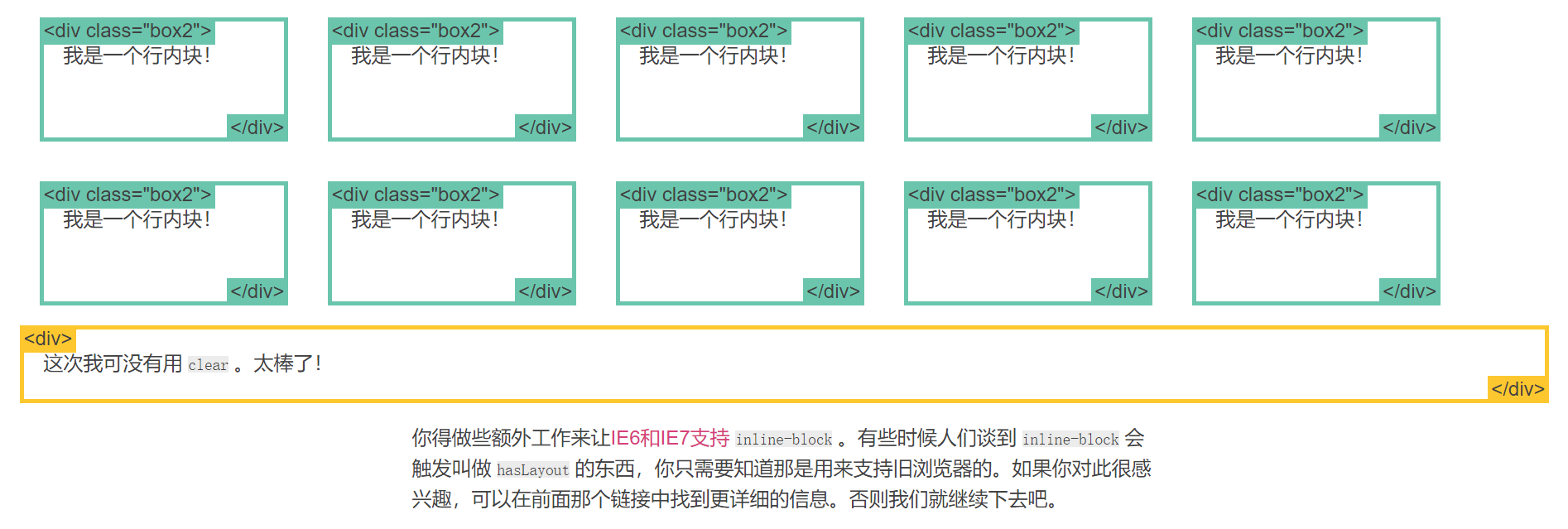


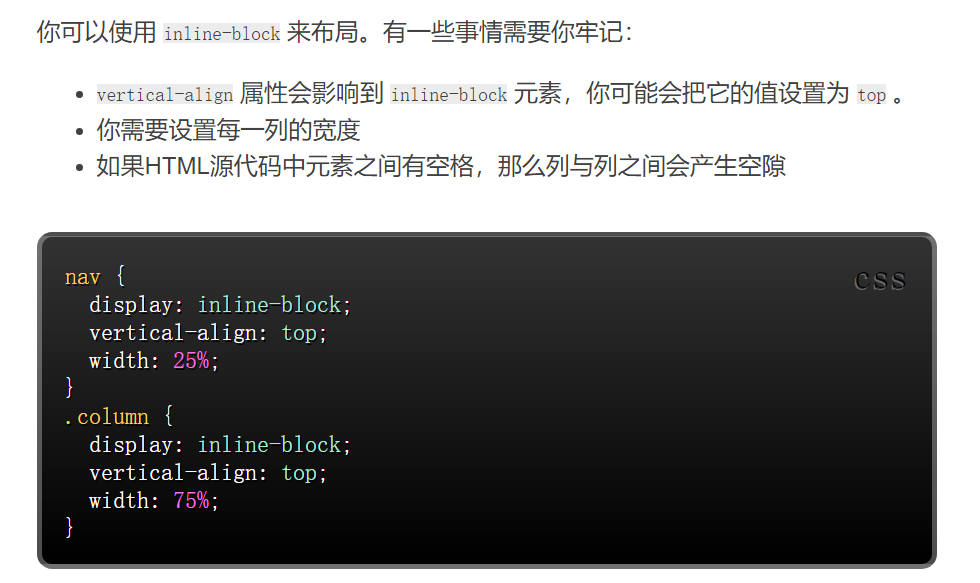
10、inline-block











11、column

12、flexbox