爱创课堂前端培训

# CSS

第4天课堂笔记（本课程共6天）

班级：北京前端训练营12期

讲师：冯楠娜

日期：2018年1月12日

爱创课堂官网 ：[www.icketang.com](http://www.icketang.com)

# 目录

[CSS 1](#_Toc26915)

[目录 2](#_Toc17078)

[复习 3](#_Toc17544)

[一、标准文档流 3](#_Toc23965)

[1.1 标准文档流特征 3](#_Toc6193)

[1.2 分类 4](#_Toc29356)

[二、 浮动 5](#_Toc12679)

[2.1 浮动概述 5](#_Toc832)

[2.2 浮动的性质 5](#_Toc14245)

[2.3 浮动存在问题 8](#_Toc24693)

[2.4 解决办法 9](#_Toc3070)

[2.5 overflow 10](#_Toc28585)

# 复习

文字属性：

|  |
| --- |
| 1. font-size:20px; 2. Font-family:”Arial”,”Microsoft Yahei”; 3. Font-weight:bold; 4. Font-style:italic; 5. Font:italic bold 20px/40px ”Arial”,”Microsoft Yahei”; |

|  |
| --- |
| 1. text-align:center; 2. Text-decoration:none; 3. Text-indent:2em; |

盒模型：

内容区域：width,height

盒子占有区域：盒子占有宽 = width + padding \* 2 + border \* 2

盒子占有高 = height + padding \* 2 + border \* 2

背景渲染区域：border以内（padding可以渲染背景）

Css书写第一步：清除默认样式

|  |
| --- |
| 1. \*{padding:0;margin:0;} |

盒子水平居中：margin: 0 auto;

父子盒模型：子盒子的实际占有宽度不能超过父盒子内容区域。（內减）

# 一、标准文档流

标准文档流，类似于word文档。光标位置决定书写位置，当光标前面的文字字号变大或者图片变大，光标会往下移动。书写顺序只能是从左到右，从上到下书写。

word文档：是标准文档流，书写从上到下，从左到右。

网页制作：也是标准文档流，书写是上到下。

## 标准文档流特征

* 文本内容有空白折叠现象

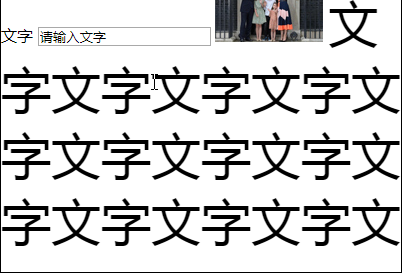
|  |
| --- |
| 1. <p>文字 文字文字 文字文字文字文字文字文字文字文字文字文字文字文 2. 字文字文字文字文字文字文字文字文字文字文字文字文字文字文字文字文字文字文字文字文字文字</p> |



* 文本高低不齐底边对齐的效果（文字，input,img,b）



* 文本内容超过父盒子宽度时自动换行。



## 分类

标准文档流标签的分类：块级元素和行内元素。

块级元素：所有的容器级标签都是块级元素，p标签也还是块级元素。

行内元素：所有的文本级标签都是行内元素，p标签除外。

|  |
| --- |
| 1. 块级元素： p,div,h1,h2……h6,ul ,li,ol,dl等 2. 行内元素：a, span , b, u ,i ,img ,input等 |

块级元素特征：

* 可以设置宽和高。
* 块级元素不设置宽，会撑满父盒子的内容宽。（width:100%）
* 块级元素独占一行。

行内元素特征：

* 不能设置宽高
* 行内元素的宽，高是由内容撑开。
* 行内元素不独占一行，有其他行内元素会并排显示。

块级元素和行内元素可以相互转换：display(显示模式)

属性值：block(行内元素转为块级元素)

inline(块级元素转为行内元素)

inline-block(行内块)即可以设置宽高有可以并排显示。

|  |
| --- |
| 1. .teshu{ 2. width: 300px; 3. height: 300px; 4. background-color: pink; 5. /\*转换为块级元素\*/ 6. display: block; 7. } |

标准文档流制作网页非常不灵活，不能制作精美的网页。只能让元素脱离标准文档流（脱标）。

脱标的方法：浮动，绝对定位，固定定位。

# 浮动

## 2.1 浮动概述

Float:浮动

属性值：left(左浮动)

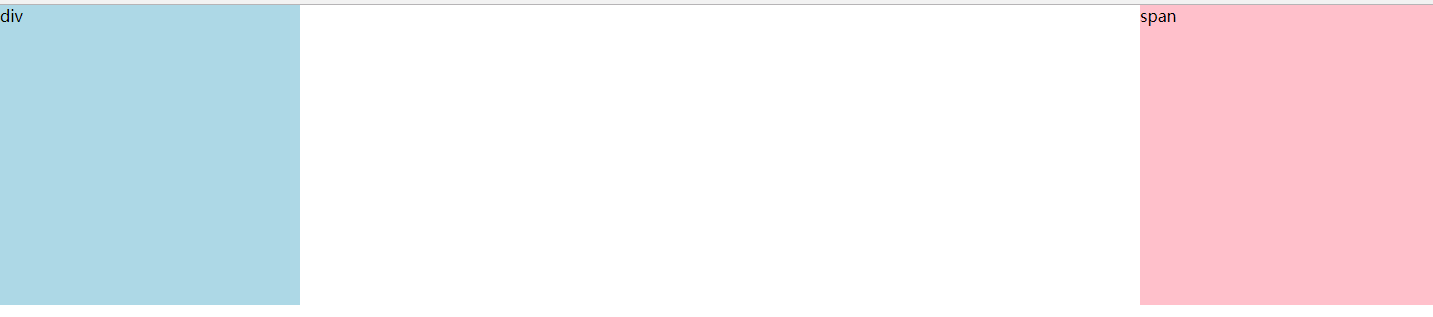
Right(右浮动)

浮动的元素会脱离标准文档流。脱标的元素不在分区块级元素和行内元素。脱离标准文档流的元素既可以设置宽高又可以并排显示。

|  |
| --- |
| 1. div{ 2. width: 300px; 3. height: 300px; 4. background-color: lightblue; 5. float:left; 6. } 7. span{ 8. width: 300px; 9. height: 300px; 10. background-color: pink; 11. float:right; 12. } |

## 2.2 浮动的性质

* 浮动的元素会脱标



* 浮动的元素会依次贴边显示

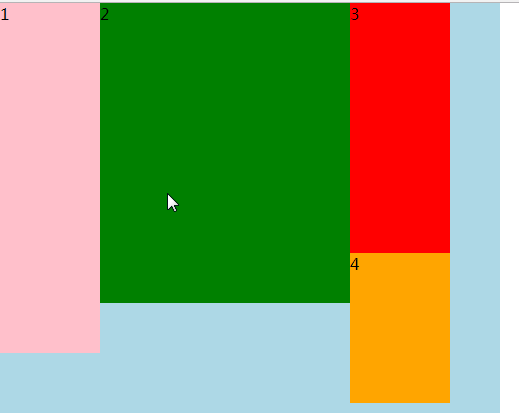
以左浮动为例

父盒子的左侧 → 子盒子1 → 子盒子2 → 子盒子3 → 子盒子4

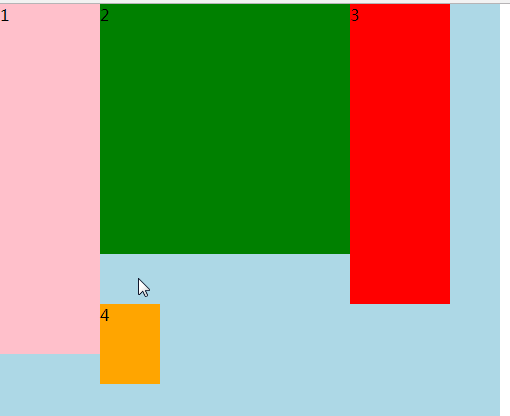
|  |
| --- |
| 1. .box p{ 2. /\*float:right;\*/ 3. float:left; 4. width:100px; 5. height: 100px; 6. background-color: pink; 7. } |



如果子盒子的宽度超过父盒子的宽，继续向前一个元素贴边显示。

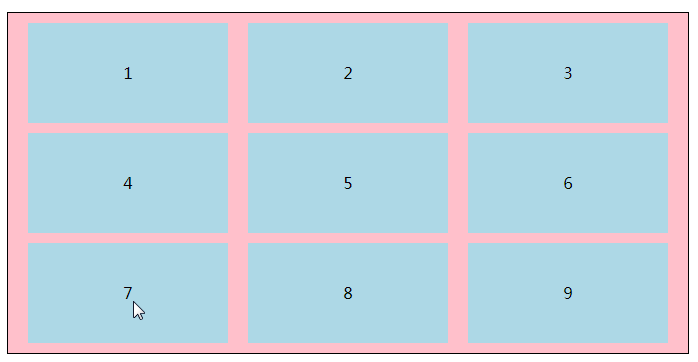


浮动的元素不会钻盒子



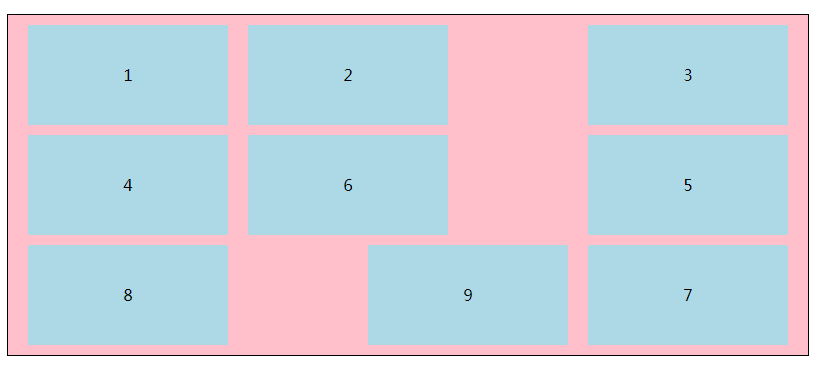
表格项：

|  |
| --- |
| 1. .box{ 2. width:680px; 3. height:340px; 4. border: 1px solid #000; 5. margin:50px auto; 6. background-color: pink; 7. } 8. .box ul{ 9. list-style: none; 10. padding-left:20px; 11. padding-top: 10px; 12. } 13. .box ul li{ 14. width: 200px; 15. height: 100px; 16. background-color: lightblue; 17. float: left; 18. margin-right: 20px; 19. margin-bottom: 10px; 20. text-align: center; 21. line-height: 100px; 22. } |



* 浮动的元素双向贴边显示

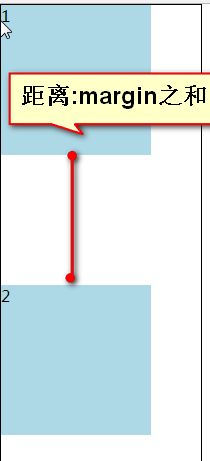
|  |
| --- |
| 1. <div class="box"> 2. <ul> 3. <li>1</li> 4. <li>2</li> 5. <li class="fr">3</li> 6. <li>4</li> 7. <li class="fr">5</li> 8. <li>6</li> 9. <li class="fr">7</li> 10. <li>8</li> 11. <li class="fr">9</li> 12. </ul> 13. </div> |



* 浮动的元素没有margin的塌陷

模型：两个垂直排列的浮动盒子，上盒子有一个下margin，下盒子有一个上margin。两个盒子距离就是margin之和没有塌陷。

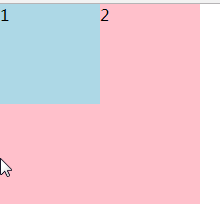
|  |
| --- |
| 1. .box p.no1{ 2. margin-bottom: 50px; 3. } 4. .box p.no2{ 5. margin-top: 80px; 6. } |



* 浮动的元素会让出标准流的位置

模型：两个同级盒子，一个盒子浮动，一个盒子标准流，浮动的盒子会让出标准流的位置。

|  |
| --- |
| 1. .box1{ 2. float: left; 3. width: 100px; 4. height: 100px; 5. background-color: lightblue; 6. } 7. .box2{ 8. width: 200px; 9. height: 200px; 10. background-color: pink; 11. } |

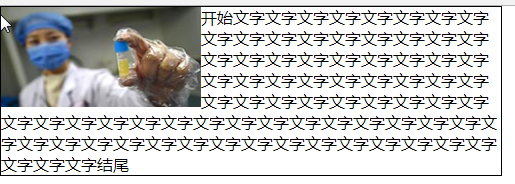


盒子1让出标准流的位置，形成“压盖效果”。但是压盖效果不用浮动书写，以后会学定位。

同级的盒子要浮动都浮动，要不浮动都不浮动。

* 字围效果

同级元素，浮动的盒子会压盖住标准流的盒子，但是文字不会有压盖效果，会在浮动的元素周围显示。

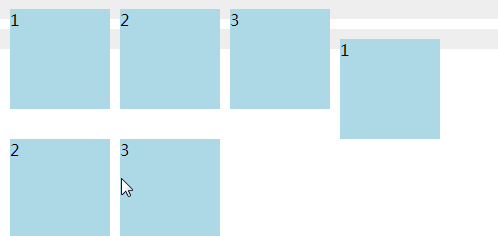


## 2.3 浮动存在问题

①浮动的元素不能撑高父盒子（高度自适应）。

②浮动的元素会影响后面浮动的元素。

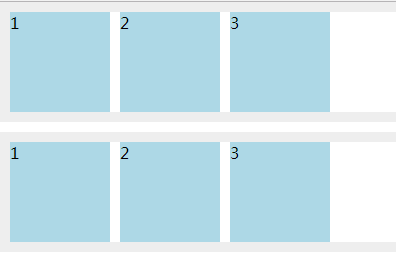
|  |
| --- |
| 1. .box1 p,.box2 p{ 2. float: left; 3. width: 100px; 4. height: 100px; 5. background-color: lightblue; 6. margin-right: 10px; 7. } |



## 2.4 解决办法

* 直接给父盒子加高度

|  |
| --- |
| 1. height:100px; |



解决：

对后面浮动元素的影响。Margin生效

没解决：高度自适应

* 给后面浮动的盒子加clear属性

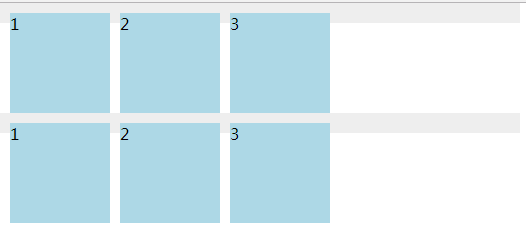
Clear：清除浮动的影响。

属性值：left(清除左浮动的影响)

Right(清除有浮动的影响)

Both(清除左右浮动的影响)

|  |
| --- |
| 1. .box2{ 2. margin-top: 10px; 3. /\*clear:left;\*/ 4. clear:both; 5. } |



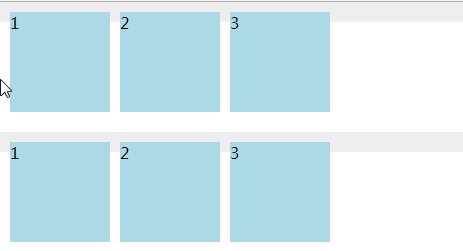
解决：对后面浮动元素的影响

没有解决：高度自适应，margin失效

* 外墙法

在两个父盒子之间加一堵有高度并且清除了浮动的墙。

|  |
| --- |
| 1. <div class="box1"> 2. <p>1</p> 3. <p>2</p> 4. <p>3</p> 5. </div> 6. <div class="cl"></div> 7. <div class="box2"> 8. <p>1</p> 9. <p>2</p> 10. <p>3</p> 11. </div> |



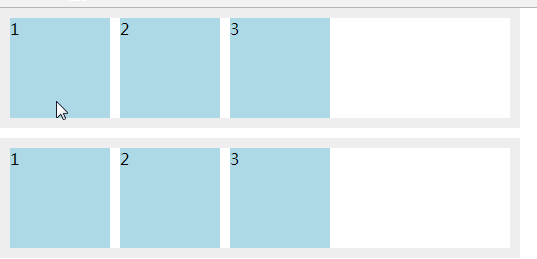
解决：对后面浮动元素的影响，视觉有了间隔

没解决：高度自适应

* 内墙法

在浮动元素的最后添加一堵清除了浮动的墙。

|  |
| --- |
| 1. <div class="box1"> 2. <p>1</p> 3. <p>2</p> 4. <p>3</p> 5. <div class="cl"></div> 6. </div> 7. <div class="box2"> 8. <p>1</p> 9. <p>2</p> 10. <p>3</p> 11. <div class="cl"></div> 12. </div> |



解决：都解决了

虽然解决了所有的问题，但是不使用该方法。因为给页面增加了很多无意义的标签。

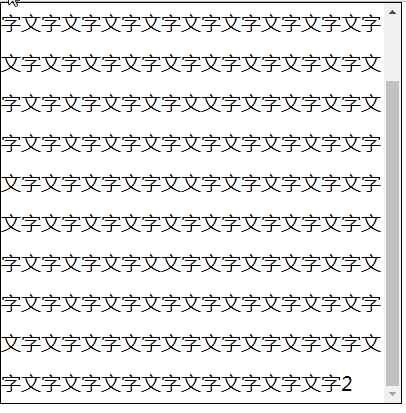
## 2.5 overflow

* Overflow属性

overflow:溢出处理的方法

属性值：auto(溢出滚动）当文字的内容超过盒子的高度时，会出现滚动条，拉动滚动可以查看全部的文字。

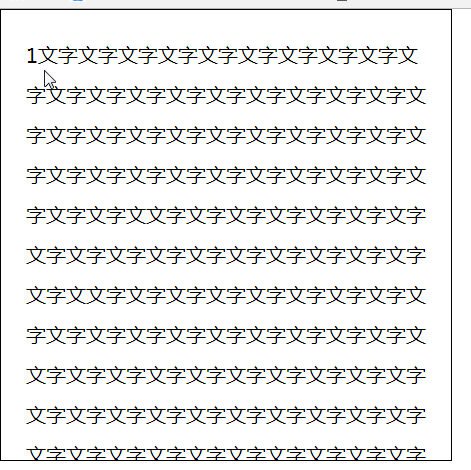
|  |
| --- |
| 1. overflow: auto; |



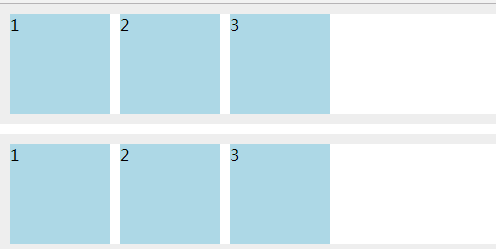
属性值：hidden;(溢出隐藏)当文字的内容超过盒子的高度时，直接隐藏超过盒子高度范围外的文字。

特别的当盒子有padding时，下padding区域不会隐藏文字，只有border边框外的文字才会隐藏。

|  |
| --- |
| 1. .box{ 2. width: 400px; 3. height: 400px; 4. border: 1px solid #000; 5. font-size: 20px; 6. line-height: 40px; 7. padding:25px; 8. /\*overflow: auto;\*/ 9. overflow: hidden; 10. } |



|  |
| --- |
| 1. .box1,.box2{ 2. width: 500px; 3. border: 10px solid #eee; 4. overflow: hidden; 5. } |



解决：解决了所有的问题，同时没有增加任何标签。

overflow：hidden有俩个引申含义：

①overflow：hidden会强制让父盒子去检测子盒子的高度，将检测的子盒子的高度自动撑开自己的高度。（只要看到盒子高度自适应，只需要给父盒子加overflow：hidden）

②overflow：hidden；会强制检测子盒子的浮动，会让父盒子强制管理住自己的浮动的子盒子，不让子盒子影响别人同时也不让别人影响自己。

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |

|  |
| --- |
|  |