

# 珠峰培训CSS从入门到精通第六天

## 1、认识标准文档流及特性

**文档流** 指的是元素在文档结构中的排列方式；

**标准文档流** 网页在解析的时候，遵循于从上向下，从左向右的一个顺序，而这个顺序就是来源于标准文档流；

**脱离文档流**：打破原本的从上到下，从左到右的排列方式，比如重叠、叠加、覆盖等非正常的文档流的排列布局情况；

## 2、CSS中的三种基本的定位机制

CSS 有三种基本的定位机制：常规流、浮动和绝对定位。

### 1) 常规流

- 除非专门指定，否则所有框都在普通流中定位。普通流中元素框的位置由元素在(X)HTML中的位置决定。块级元素**从上到下**依次排列，框之间的垂直距离由框的垂直margin计算得到。行内元素在一行中**从左向右**

水平布置。

- 常规流不脱离文档流。
- 浮动、绝对定位和固定定位脱离文档流。

## 2) 浮动流

### 一、浮动的基础语法

**浮动概念：** float 属性指定一个盒子（元素）是否应该浮动。

**\*\*浮动特性：**

**特性一：**浮动的框可以向左或向右移动，直到它的外边缘碰到包含框或另一个浮动框的边框为止。

**特性二：**由于浮动框不在文档的普通流中，所以文档的普通流中的块框表现得就像浮动框不存在一样。

### 浮动属性值

- none：设置对象不浮动
- left：设置对象浮在左边
- right：设置对象浮在右边

float 属性定义元素在哪个方向浮动。以往这个属性总应用于图像，使文本围绕在图像周围，不过在 CSS 中，任何元素都可以浮动。浮动元素会生成一个块级框，而不论它本身是何种元素。

### 二、浮动应用：图文环绕

float最初被设计的时候就是用来完成文字环绕的效果的。

浮动元素后面的块级元素的内容会向此浮动元素的外边距靠齐，但是边框和背景却忽略浮动元素而向上一个（实例中为父元素）任意非浮动元素靠齐；

## 浮动特性

- 块级元素浮动将并排显示，不再独占一行
- 内联样式浮动将支持设置宽高
- 浮动元素宽高由内容决定
- 浮动元素脱离正常的文档流
- 浮动元素提升半层级
- 浮动不会继承也不会传递
- 浮动元素之前的元素不会受当前浮动元素的影响，如果你想让两个块状元素并排显示，必须要让两个块状元素都应用float属性。

## 缺点

- 会导致父元素的高度失效 值为0，后面的元素识别不了它的高度，就会往前占领它的位置；

## 3 ) 定位流

后续讲

## 3、BFC与IFC

## BFC的含义

- BFC ( Block Formatting Context ) 叫做“块级格式化上下文”；

## BFC的表现原则

- 如果一个元素具有BFC，内部子元素再怎么翻江倒海、翻云覆雨，都不会影响外部的元素。

## 如何触发BFC

默认情况（普通文档流下）叫FC

- 1.我们的 `html` 根元素就是一个 BFC；
- 2.float的属性不为 `none`；即值为 `left` 和 `right`；
- 3.overflow为 `auto`、`scroll`、`hidden`；
- 4.display为 `inline-block`，`table-cell`，`table-caption` 中的任何一个可以触发 BFC；
- 5.position为 `absolute` 或 `fixed` 时可以触发 BFC；

## BFC布局规则

- 1.内部的盒子会在垂直方向，一个个地放置；

- 2.盒子垂直方向的距离由margin决定，属于同一个BFC的两个相邻Box的上下margin会发生重叠；
- 3.每个元素的左边，与包含的盒子的左边相接触，即使存在浮动也是如此；
- 4.BFC的区域不会与float重叠；
- 5.BFC就是页面上的一个隔离的独立容器，容器里面的子元素不会影响到外面的元素，反之也如此；
- 6.计算BFC的高度时，浮动元素也参与计算；  
\*\*可以形成一个封闭的区域，它里面的内容不会影响到外面的元素。  
**可以形成一个封闭的区域，它里面的内容不会影响到外面的元素。**

### 3、清除浮动方法（面试题）

#### 方法一：给父盒子加固定高度

给前面浮动元素的父元素设置一个固定高度

注意点：实际项目开发中，我们能不写高度就不写高度，所以这种方式用得很少

#### 方法二：清除浮动的方法之隔墙法与内墙法

**外墙法：**在两个盒子中间添加一个额外的块级元素，给这个额外添加的块级元素设置clear: both;属性

注意点：外墙法它可以让第二个盒子使用margin-top属性；外墙法不可以让第一个盒子使用margin-bottom属性

**内墙法：**在第一个盒子中所有子元素最后添加一个额外的块级元素，给这个额外添加的块级元素设置clear: both;属性

注意点：内墙法它可以让第二个盒子使用margin-top属性，内墙法它可以让第一个盒子使用margin-bottom属性

// 具备两个条件：

// 1. 具有块特点的标签，就算不是块级标签也需要转换为块级标签

// 2. 给这个块级标签添加一个属性clear: both

```
<div class="main">
  <div class="right">2</div>
  <div class="left">1</div>
  <div style="clear: both;"></div>
</div>
```

```
<div class="main">
  <div class="right">2</div>
  <div class="left">1</div>
  <a style="display: block; clear: bot
```

```
h;"></a>  
</div>
```

### 方法三：给父元素加overflow:hidden;

- overflow : visible : 对溢出内容不做处理，内容可能会超出容器。
- overflow : hidden : 隐藏溢出容器的内容且不出现滚动条。
- overflow : scroll : 隐藏溢出容器的内容，溢出的内容将以滚动滚动条的方式呈现。
- overflow : auto : 当内容没有溢出容器时不出现滚动条，当内容溢出容器时出现滚动条，按需出现滚动条。此为body对象和textarea的默认值。
- overflow的效果等同于overflow-x + overflow-y。

### 方法四：清除浮动的方法之clearfix法

利用clearfix 去清除浮动(实际项目中最常用的方法)

我是一块砖 哪里需要 就往哪里搬

```
<style type="text/css">  
.clearfix:after{  
    content:'';
```

```
        display:block;
        clear:both;
    }
</style>
<div class="main clearfix">
    <div class="right">2</div>
    <div class="left">1</div>
</div>
<div class="main1 clearfix">
    <div class="right">2</div>
    <div class="left">1</div>
</div>
```

// 最完整clearfix清除浮动的方法:

```
.clearfix:after{
    content:'';
    display:block;
    clear:both;
    width:0;
    height:0;
    line-height:0;
    font-size:0;
    visibility: hidden;
    overflow:hidden;
}
.clearfix{
    zoom:1;
}
```



## 4、CSS让元素消失在视野中发挥想象力(面试题)

- display: none 让元素消失
- font-size : 0 字体大小为0
- width:0; overflow:hidden 宽度为0 而且溢出隐藏
- height:0; overflow:hidden 高度为0 而且溢出隐藏
- line-height:0; overflow:hidden 行高为0 而且溢出隐藏
- visibility: hidden; 可见性的元素隐藏消失
- opacity: 0; 透明度为0
- color / background: transparent 字体颜色或者背景颜色为透明
- text-indent: -9999px; 首段缩进足够大的负值
- position: absolute; left/top/bottom/right :-9999px; 定位 + 方位的足够大的负值
- margin-top/bottom/left/right: -9999px; 的四个方位的足够大的负值
- transform: translateX(-9999px) / translateY(-9999px); 位移坐标值足够大的负值
- transform: scale(0); 缩放比为0

## 5、单行文本和多行文本出现省略号的条件(面试题)

- 单行文本出现省略号的条件(四个) :
  - width 设置框的宽度

- `white-space: nowrap;` 禁止文字内容折行
- `overflow: hidden;` 超出部分溢出隐藏
- `text-overflow: ellipsis;` 文字以省略号的方式隐藏

```
<style>
  p {
    width: 200px; /*宽度*/
    white-space: nowrap; /*禁止文字内容折行*/
    overflow: hidden; /*超出部分溢出隐藏*/
    text-overflow: ellipsis; /*文字以省略号的方式隐藏*/
  }
</style>
<p>单行文本出现省略号单行文本出现省略号单行文本出现省略号</p>
```

- 多行文本出现省略号的条件(四个) :
  - `display: -webkit-box;` 弹性盒模型
  - `-webkit-box-orient: vertical;` 文字垂直排列
  - `-webkit-line-clamp: 2;` 文字显示的行数
  - `overflow: hidden;` 超出部分溢出隐藏

```
<style>
```

```
p {  
    display: -webkit-box; /* 弹性盒模  
型*/  
    -webkit-box-orient: vertical; /* 文  
字垂直排列 */  
    -webkit-line-clamp: 2; /*文字显示的行  
数*/  
    overflow: hidden; /*超出部分溢出隐  
藏*/  
}  
</style>  
<p>多行文本出现省略号多行文本出现省略号多行文  
本出现省略号多行文本出现省略号</p>
```