

1、说说盒模型

html中的元素不论怎么放置，形状都是矩形的、规规矩矩水平放置的，也就是通常意义上盒子的形象。

为了给文档书中的各个元素排版定位（布局），浏览器会根据渲染模型（样式position、float等）为每个元素生成四个嵌套的矩形框，分别称作content box、padding box、border box和margin box，它们是不可分割的，但可能会重合，这就是CSS规范中描述的“盒模型”，也就是以CSS的角度去看一个元素被渲染后的抽象形态。是讲一个元素自身的构成部分，不同于布局：多个元素在页面上的定位。

2、css reset和normalize.css有什么区别？

答：reset和normalize都是对浏览器默认样式的统一改造处理。

reset是传统版本，旨在删除所有内置的浏览器样式，这样更易于保持各浏览器渲染的一致性。normalize是兼容了css3的版本，旨在使浏览器内置的浏览器样式保持一致，尽量保留浏览器的默认样式，不进行太多的重置。

Normalize.css只是一个很小的css文件，但它在默认的HTML元素样式上提供了跨浏览器的高度一致性，相比于传统的CSS reset，Normalize.css是一种现代的、为HTML5准备的优质替代方案。

Normalize.css是一种CSS reset的替代方案。创造normalize.css的目的：1.保护有用的浏览器默认样式而不是完全去掉它们；2.一般化的样式 为带部分HTML元素提供；3.修复浏览器自身的bug并保证各浏览器的一致性；4.优化CSS可用性 用一些小技巧；5.解释代码 用注释和详细的文档来。

Normalize.css支持包括手机浏览器在内的超多浏览器，同时对HTML5元素、排版、列表、嵌入的内容、表单和表格都进行了一般化。尽管这个项目基于一般化的原则，但我们还是在合适的地方使用了更实用的默认值。

3、如何居中？

答：水平居中主要是text-align：center和margin:0 auto；垂直居中主要是padding、transform+absolute、flexbox、margin+top+absolute

4、选择器优先级如何确定？

答：不同类型间：html标签属性 > id选择器 > 类选择器 > 元素选择器

同类型间：CSS规则写在最后面的生效！

!important：样式最高优先级，无视优先级，在样式的分号前加上，使该样式起作用。

5、BFC是什么？

答：Block Formatting Context块格式化上下文

对一个元素设置CSS，首先需要知道这个元素是block还是inline类型。而BFC就是用来格式化块级盒子的，同样管理inline类型的盒子还有IFC，以及其他的FC。

Formatting Context：指页面中的一个渲染区域，并且拥有一套渲染规则，他决定了其子元素如何定位，以及与其他元素的相互关系和作用。

BFC：块级格式化上下文，它是指一个独立的块级渲染区域，只有Block-level Box参与，该区域拥有一套渲染规则来约束块级盒子的布局，且与区域外部无关。

Box：CSS布局的基本单位。Box是CSS布局的对象和基本单位，直观点来说，就是一个页面是由很多个Box组成的。元素的类型和display属性，决定了这个Box的类型。不同类型的Box，会参与不同的Formatting Context（一个决定如何渲染文档的容器），因此Box内的元素会以不同的方式渲染。

首先看一下有哪些盒子：

块级盒子：display:block/list-item/table

行内盒子：display:inline/inline-block/inline-table

run-in盒子：会根据上下文作为块级元素或内联元素显示

Formatting context：Formatting context是W3C CSS2.1规范中的一个概念。它是页面中的一块渲染区域，并且有一套渲染规则，它决定了其子元素将如何定位，以及和其他元素的关系和相互作用。最常见的Formatting Context有Block formatting content（简称BFC）和Inline formatting context（简称IFC）。

CSS2.1中只有BFC和IFC，CSS3中还增加了GFC和FFC。

BFC定义：BFC（Block formatting context）直译为“块级格式化上下文”。它是一个独立的渲染区域，只有Block-level box参与，它规定了内部的Block-level Box如何布局，并且与这个区域外部毫不相干。

BFC布局规则：

内部的Box会在垂直方向，一个接一个地放置；

Box垂直方向的距离由margin决定。属于同一个BFC的两个相邻Box的margin会发生重叠；

每个元素的margin box的左边，与包含块border box的左边相接触（对于从左往右的格式化，否则相反）。即使存在浮动也是如此。

BFC的区域不会与float box重叠。

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。反之亦然。

计算BFC的高度时，浮动元素也参与计算

哪些元素会生成BFC

根元素

float属性不为none

position为absolute或fixed

display为inline-block、table-cell、table-caption、flex、inline-flex

overflow不为visible

BFC的作用及原理

- 1、自适应两栏布局
- 2、清除内部浮动

3、防止垂直margin重叠

6、如何清除浮动？

答：在非IE浏览器（如Firefox）下，当容器的高度为auto，且容器的内容中有浮动（float为left或right）的元素，在这种情况下，容器的高度不能自动伸长以适应内容的高度，使得内容溢出到容器外面而影响（甚至破坏）布局的现象。这个现象叫浮动溢出，为了防止这个现象的出现而进行的CSS处理，就叫CSS清除浮动。

浮动元素的性质：

1、浮动元素会脱离正常的文档流，按照其外边距指定的位置相对于它的上一个块级元素（或父元素）显示；2、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；3、浮动元素后面的内联元素会向此浮动元素的外边距靠齐。

清除浮动：

1.清除浮动一定要在浮动标签完成浮动布局以后添加，否则会影响到浮动标签布局；2.清除浮动必须与前面的标签属于同级关系。

①clear属性定义了元素某个方向上不允许出现浮动元素：none（允许两边都可以有浮动对象）/left（.....）/right（不允许两边有浮动对象）/both（不允许有浮动对象）。

②空标签清除浮动：在HTML页面中加入一个空标签用来清除浮动。（空标签清除浮动会增加多余的代码）

③overflow：在需要清除浮动的父级标签中加入overflow即可。

④伪元素after清除浮动

⑤子标签浮动时，给父标签浮动

⑥下一标签直接清除浮动：兄弟标签浮动时，下一标签直接写入清除clear:both;就可以了。

⑦使用position:absoluted;清除浮动

总结：清除浮动的方法可以分成两类——1.利用clear属性，包括在浮动元素末尾添加一个带有clear:both属性的空div来闭合元素，其实利用:after伪元素的方法也是在元素末尾添加一个内容为一个点并带有clear:both属性的元素实现的。2.触发浮动元素父元素的BFC，使得该父元素可以包含浮动元素。

