1. 如何设置标签的背景颜色？

--在CSS中有一个background-color: 属性，就是专门用来设置标签的背景颜色的

--取值：

--具体单词

--rgb

--rgba

--十六进制

1. 如何设置背景图片

--在CSS中有一个叫做background-image: url(); 的属性，就是专门用于设置背景图片的

--注意点：

--图片的地址必须放在url()中，图片的地址可以是本地的地址，也可以是网络的地址

--如果图片的大小没有标签的大小大，那么会自动在水平和和垂直方向平铺和填充

--如果网页中出现了图片，那么浏览器就会再次发送请求获取图片

1. 如何控制背景图片的平铺方式？

--在CSS中，有一个background-repeat属性，就是专门用于控制背景图片的平铺方式的

--取值：

--repeat 默认，在水平和垂直都需要平铺

--no-repeat 在水平和垂直都不平铺

--repeat-x 只在水平方向平铺

--repeat-y 只在垂直方向平铺

--应用场景：

--可以通过背景图片的平铺来降低图片的大小，提升网页的访问速度

1. 如何控制背景图片的位置？

--在CSS中有一个叫做background-position: 属性，就是专门用于控制背景图片的位置的

--格式：

--background-position: 水平方向 垂直方向;

--取值：

--具体方位名词

--水平方向：left center right

--垂直方向：top center bottom

--具体的像素

--例如：50px 100px 200px

--一定要写单位px，不写px无效。可以取负值，负值就是将图片往外移

--简写：

--background: 背景颜色 背景图片 平铺方式 关联方式 定位方式;

如：background: red url(…) no-repeat left bottom

--注意点：

--background属性中，任何一个属性都可以被省略

--什么是背景关联方式？

--默认情况下背景图片会随着滚动条的滚动而滚动。如果不想让背景图片随着滚动条的滚动而滚动，那么我们可以修改背景图片和滚动条的关联方式

--如何修改背景图片的关联方式

--在CSS中，有一个叫做background-attachment的属性，这个属性就是专门用于修改关联方式的

--格式：

--background-attachment: scroll;

--取值：

--scroll 默认值，会随着滚动条的滚动而滚动

--fixed 不会随着滚动条的滚动而滚动

1. 背景图片和插入图片的区别

--背景图片和文字可以在同一个容器中相互融合

插入图片和文字不会占用位置

--背景图片有定位属性，所以可以很方便的控制图片的位置

插入图片没有定位属性，所以控制图片的位置不太方便

--插入图片的语义比背景图片的语义要强，所以在企业开发中，如果你的图片想被搜索引擎收录，那么推荐使用插入图片

1. CSS精灵图

--什么是CSS精灵图

--CSS精灵图是一种图像合成技术

--精灵图的作用

--可以减少请求的次数，以及可以降低服务器的处理压力

--如何使用CSS精灵图

--CSS的精灵图需要配合背景图片和背景定位来使用

1. 边框属性

--什么是边框属性

--边框就是环绕在标签宽度和高度周围的线条

--边框属性的格式

--连写（同时设置四条边框）

--border: 边框的宽度 边框的样式 边框的颜色;

--注意点：

--在连写格式中，颜色属性可以省略，省略后默认是黑色

--在连写格式中，样式不能省略，省略后就看不到边框了（具体样式在w3cschool中查找）

--在连写格式中，宽度可以省略，省略后还是会看到一条很窄的边框

--连写（分别设置四条边框）

--border-top: 边框的宽度 边框的样式 边框的颜色;

--border-right: 边框的宽度 边框的样式 边框的颜色;

--border-left: 边框的宽度 边框的样式 边框的颜色;

--border-bottom: 边框的宽度 边框的样式 边框的颜色;

--连写（分别设置四条边框）

--border-width: 上 右 下 左;

--border-style: 上 右 下 左;

--border-color: 上 右 下 左;

--注意点：

--这三个属性的取值是按照顺时针来赋值的，也就是按照上右下左来赋值，而不是日常生活中的上下左右

--缺省规律：

--只缺少某一条边的取值，那这条边的取值会跟它的对边取值相同（如：缺省右边，则右边的取值跟左边一样。上下边同理）。若缺省了三条边，那这三条边的取值就会跟上边的取值一样

--*将某一条边的值设为none，就会有去掉该条边的效果*

1. 内边距（文字到边框的距离）

--非连写

--padding-top: ;

--padding-right: ;

--padding-bottom: ;

--padding-left: ;

--注意点：

--给标签设置内边距之后，标签占有的宽度和高度会发生变化

--连写

--padding: 上 右 下 左;

--注意点：

--内边距可以显示背景颜色

1. 外边距（边框与边框之间的距离）

--非连写

--margin-top: ;

--margin-right: ;

--margin-bottom: ;

--margin-left: ;

--连写

--margin: 上 左 右 下;

--缺省值的效果和前面讲过的一样

--注意点：

--外边距是不会显示背景颜色的，也就是在div中设置的颜色不会在外边距中显示

--在默认布局的垂直方向上，默认情况下外边距不会叠加，会出现合并现象，谁的外边距大就听谁的。水平方向上可以叠加。

1. CSS盒子模型

--CSS盒子模型仅仅是一个形象的比喻，HTML中所有的标签都是盒子

--在HTML中所有的标签都可以设置

--宽度/高度 == 指定可以存放内容的区域

--内边距

--边框

--外边距

--内容的宽度和高度

--就是通过width/height属性设置的宽度和高度

--元素的宽度和高度

--宽度 = 左边框 + 左内边框 + width + 右边框

--高度 同理可证

--元素空间的宽度和高度

--宽度 = 左外边框 + 左边框 + 左内边框 + width + 右内边框 + 右边框 + 右外边框

--高度 同理可证

1. 清空默认边距

--为什么要清空默认边距（外边距和内边距）

--在企业开发中，为了更好的控制盒子的宽高和计算盒子的宽高等，在编写代码之前，做的第一件事情就是清空默认的边距

--如何清空默认边距

--格式：

\*{

margin: 0;

padding: 0;

}

--注意点：

--通配符选择器会找到（遍历）当前所有的标签，效率比较低。因此，在企业开发中通常会采用另外一种清空方式

body,div,dl,dt,dd,ul,ol,li,h1,h2,h3,h4,h5,h6,pre,code,form,fieldset,legend,input,textarea,p,blockquote,th,td{

margin: 0;

padding: 0;

}

参考网址：<http://yui.yahooapis.com/3.18.1/build/cssreset/cssreset-min.css>

1. 行高和字号

--什么是行高

--在CSS中所有的行都有自己的行高

--注意点

--行高和盒子高不是同一个概念

--规律：

--文字在行高中默认是垂直居中的

--在企业开发中，经常将盒子的高度和行高设置为一样，这样盒子里的内容就能实现居中的效果

--如果一个盒子中有多行文字，那么我们就不能使用设置行高等于盒子高来实现让文字垂直居中，只能通过设置padding属性来让文字居中

1. 还原字体和行高

--通过fireworks软件测量出网页的文字字体行高等

--注意点：

--在企业开发中，如果一个盒子中存储的是文字，那么一般情况下我们会以盒子左边的内边距为基准，不会以右边的内边距为基准，因为这个右边的内边距有误差

--右边的内边距的误差从何而来？

--因为右边如果放不下一个文字，那么文字就会换行显示，所以文字和内边距之间的距离就会有误差

--顶部的内边距并不是边框到文字顶部的距离，而是边框到行高顶部的距离

