**Q：一个优秀的前端开发人员需要具备哪些特质？**

**A：**从个人能力来讲，需要具备较强的学习能力，解决问题的能力和沟通能力。从代码层次来讲，要求开发人员能够写出较为规范性且易于维护的代码。

**Q：css预加载的使用？**

**A:**

**相同点：**他们是动态的样式语言，是CSS预处理器,CSS上的一种抽象层。他们是一种特殊的语法/语言而编译成CSS。

**不同点：**

- 变量符不一样，less是@，而Sass是$;

- Sass支持条件语句，可以使用if{}else{},for{}循环等等。而Less不支持;

- Sass是基于Ruby的，是在服务端处理的，而Less是需要引入less.js来处理Less代码输出css到浏览器

补充链接：<https://juejin.im/post/5a2bc28f6fb9a044fe464b19>

**Q:关于懒加载和预加载**

**A:** <https://www.jianshu.com/p/4876a4fe7731>

回答套路，按3W1H模式回答。即：①是什么What？；②在何处使用Where?；③为什么Why？；④如何使用How？

基于上述四点回答完之后，可适当延伸到相关知识点来讲。

**Q：常见的解决浏览器兼容性问题的方式有哪些？**

**A:** <https://blog.csdn.net/xustart7720/article/details/73604651/>

**常见：**

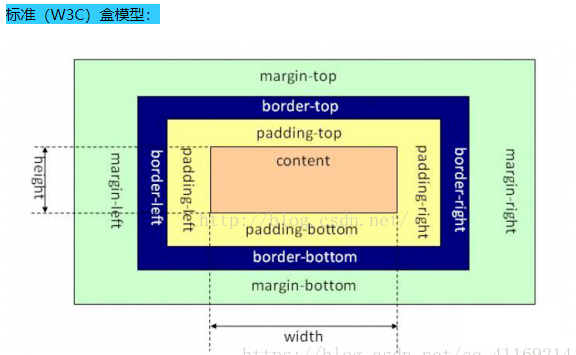
①不同浏览器的标签默认的外补丁和内补丁不同：css里 \*{margin:0;padding:0;}

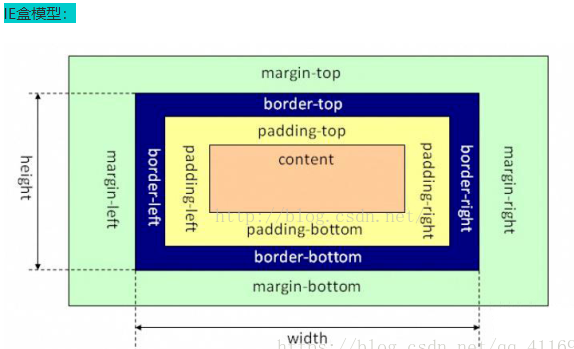
②块属性标签float后，又有横行的margin情况下，在ie6显示margin比设置的大：在float的标签样式控制中加入 display:inline;将其转化为行内属性

③设置较小高度标签（一般小于10px），在ie6，ie7，遨游中高度超出自己设置高度：给超出高度的标签设置overflow:hidden;或者设置行高line-height 小于你设置的高度。

**Q：CSS盒子模型**

**A：**分为两种：标准盒子模型和IE的盒子模型。标准的盒子模型的计算方式是 内外补丁+边框+内容区高(height)/宽(width)，而IE的盒子模型的width/height包括了内补丁和边框。





通过设置 box-sizing:content-box(W3C)/border-box(IE)就可以达到自由切换的效果；

**Q：****Ajax能够解析的数据类型**

**A：**XML、JSON、字符串等。

**①XML**

**优点：**XML 是一种通用的数据格式，不必把数据强加到已定义好的格式中，而是要为数据自定义合适的标记，利用 DOM 可以完全掌控文档。

**缺点：**如果文档来自于服务器，就必须得保证文档含有正确的首部信息。若文档类型不正确，那么 responseXML 的值将是空的，当浏览器接收到长的 XML 文件后， DOM 解析可能会很复杂

**②JSON**

**优点：**作为一种数据传输格式，JSON 与 XML 很相似，但是它更加灵巧。JSON 不需要从服务器端发送含有特定内容类型的首部信息。

**缺点：**语法过于严谨、代码不易读、eval 函数存在风险

**Q：如何解决跨域的问题**

**A：**两种方法：CORS和JSONP

**CORS：**服务器端对于CORS的支持，主要就是通过设置Access-Control-Allow-Origin来进行的。如果浏览器检测到相应的设置，就可以允许Ajax进行跨域的访问。

**JSONP：**动态插入script标签，通过script标签引入一个js文件，这个js文件载入成功后会执行我们在url参数中指定的函数，并且会把我们需要的json数据作为参数传入。

**Q：解释一下this**

**A：**this是一个关键字，总是指向当前调用的对象。（按3W1H模式回答，还可以讲讲替换this的几种方法）

**Q：什么是闭包**

**A：**

**定义：**一个方法调用了不属于自己的局部变量的现象。

**作用：**既重用变量，又可以保护变量不被污染

**闭包形成的原因：**外层函数的作用域对象在调用后无法释放；但也因此闭包比普通函数多占用内存

**创建闭包：**创建闭包的步骤主要为以下三步：

> 用外层函数包裹要保护的变量和使用变量的内层函数

> 外层函数将内层函数返回到外部

> 使用者调用外层函数，获得内层函数对象，保存在变量中反复使用