1. 盒子模型的理解

|  |
| --- |
| 盒子模型都具有的属性padding,margin,border,content |
| 每个盒子都有：边界，边框，填充，内容四个部分 |
| 每个属性都包括四个部分：上下左右，这四个部分可同时设置，也可分别设置 |

1. 响应式布局与自适应布局的区别

|  |  |
| --- | --- |
| 响应式 | <http://www.youzhan.org/crazycall/> bootstrap案例 |
| 自适应布局 | <https://wap.sogou.com/> 搜狗 |

|  |  |
| --- | --- |
| 自适应布局 | 以rem为单位的网站，是自适应布局网站的，随着屏幕分辨率的缩小而缩小 |
| 响应式布局 | 当屏幕分辩率，到达某个设置的分辩界限，就响应相应的单位 |
|  |  |

1. 采用了非标准的字体

|  |
| --- |
| 看能不能用图片代替 |
| webfonts (字体服务例如：Google Webfonts , TypeKit 等等。) |
| 跟设计师沟通好，尽量使用默认的字体 |

1. 使用 \*{box-sizing:content-box;} 能够统一IE和非IE浏览器之间的差异

|  |
| --- |
| IE6以前的版本中，IE对盒子模型的解析出现了一些问题，将border与padding都包含在了width之中，而另外一些浏览器则与之相反，是不包含从容的让和padding的。使用之后IE浏览器与非IE浏览器刚好相反。 |

1. 定位 static /relative /absolute /fixed的区别

|  |  |
| --- | --- |
| static | 默认的，没有定位，元素出现在正常的流中 |
| relative | 相对定位 |
| absolute | 绝对定位，默认以static定位以外的第一个父元素进行定位 |
| fixed | 绝对定位，相对于浏览器窗口进行定位 |
| 出static外，都能够进行z-index进行层次分离，通过top/right/left/bottom进行位置设置 | |
|  | |

|  |  |
| --- | --- |
| static |  |
| relative | 脱离正常的文本流中，但其在文本流中的位置依然存在 |
| absolute | 脱离正常的文本流中，但其在文本流中的位置会让灰色背景补全 |
| fixed |  |
|  | |

1. 使用预处理器的有缺点
2. 使用过那种栅格系统 网格系统
3. css 的hacks

|  |
| --- |
| \_width针对于ie6。\*width,+width针对于ie6,7。 |
| color: red\9;/\*IE8以及以下版本浏览器\*/（但是测试可以兼容到ie10。 |
| \*+html与\*html是IE特有的标签, firefox暂不支持.而\*+html又为IE7特有标签（但是测试\*html兼容ie6-10。\*+兼容ie7-10）。 |
| !important 在IE中会被忽视，ie6,7,8不识别，ie9+（包括ie9）是识别的。 |

9.浮动布局

|  |  |
| --- | --- |
| 浮起来后移动 |  |
| 把网页看层两层结构，加了浮动元素后会从底层（文档流位置）浮起来，浮起来后底层就没有东西了，导致后面的内容自动挤上来，被栏住（栏不住里面的内容）。 | |

1. get与post提交的区别
2. cookie，session，sessionSorage和localStorage的区别

|  |  |  |
| --- | --- | --- |
| cookie | 保存在浏览器端，如果不设置过期时间，会保存到内存中，声明周期随浏览器的关闭而结束，如果设置了过期时间，会保存到硬盘中，关闭浏览器后，数据依然存在，直到过期时间结束才消失。 | |
| session | 保存在服务器端 | |
| sessionSorage | 临时存储 | 会为每一个数据源维持一个存储区域，在打开，包括重新加载的时候都存在，在关闭页面后即被清空 |
| localSorage | 长期存储 | 浏览器关闭后，数据会一直存在 ，除非主动删除数据，否者数据是永远不会过期 ，也可以设置消失时间 |
|  |  |  |
| 都是用来存储东西 | | |
| sessionSrorage/localStorage/cookie 都是保存在浏览器端，且同源 | | |
| cookie存储的内容不能超过4k。session大小没有限制。localStorage可以设置大小。sessionStorage不能超过5M。 | | |
|  | | |

1. XHTML的局限

|  |
| --- |
| 放弃了一些不好的标签，但必须有head,body且每个都要闭合，一些老的浏览器不支持。 |

1. http地址请求到加载到屏幕上的整个流程

|  |
| --- |
|  |
|  |
|  |
|  |

1. 网站资源文件的优化
2. 语义化

|  |
| --- |
| 1. 我们可以让机器的理解能力越来越接近人类，人能看懂、听懂的东西，机器也能理解； |
| 1. 我们应该在发布内容的时候，就用机器可读的、被广泛认可的语义信息来描述内容，来降低机器处理 Web 内容的难度（HTML 本身就已经是朝这个方向迈出的一小步了）。 |