


《京程一灯》精英班第五讲试卷 姓名：

请按要求作答，并工整书写试卷。

- 1.请简述一下OO CSS及CSS分层理论SMACSS、BEM、SUIT、ACSS、ITCSS的概念。(5分)
答：

 这个题是你的CSS是否杂乱无章，请结合自己实战项目经验自己的描述。

1.OO CSS的概念解读：众多开发者忽视了CSS的表现（认为他太过简单，是一种机械的工作），而把更多关注在Javascript的性能上或者其他方面。OO CSS将页面可重用元素抽象成一个类，用Class加以描述，而与其对应的HTML即可看成是此类的一个实例。

2.SMACSS(Scalable and Modular Architecture for CSS): 可扩展的模块化架构的CSS，像OO CSS一样以减少重复样式为基础，但SMACSS通过使用一套五个层次来划分CSS使项目更结构化。

3.BEM:和SMACSS非常类似，主要用来给项目命名。一个简单命名更容易让别人一起工作。

4.SUIT: Suit起源于BEM，但是它对组件名使用驼峰式和连字号把组件从他们的修饰和子孙后代中区分出来。

5.ACSS: 考虑如何设计一个系统的接口。

6.ITCSS: ITCSS是一个完全不同于SMACSS的方法，它创造了一系列的层次来管理依赖关系、促进可扩展性。基础的层次包含通用和广泛的选择器，顶部的层次包含局部模块具体化的选择器。

- 2.请简述一下双飞翼布局实现思路(8分)
答：

 请兄弟姐妹们默写出来。

1.描述圣杯布局。

2.在圣杯布局的基础上如何实现双飞翼布局。

3.好了，你要利用你所有的CSS2的核心技巧 float margin(负边距) position 盒子模型。

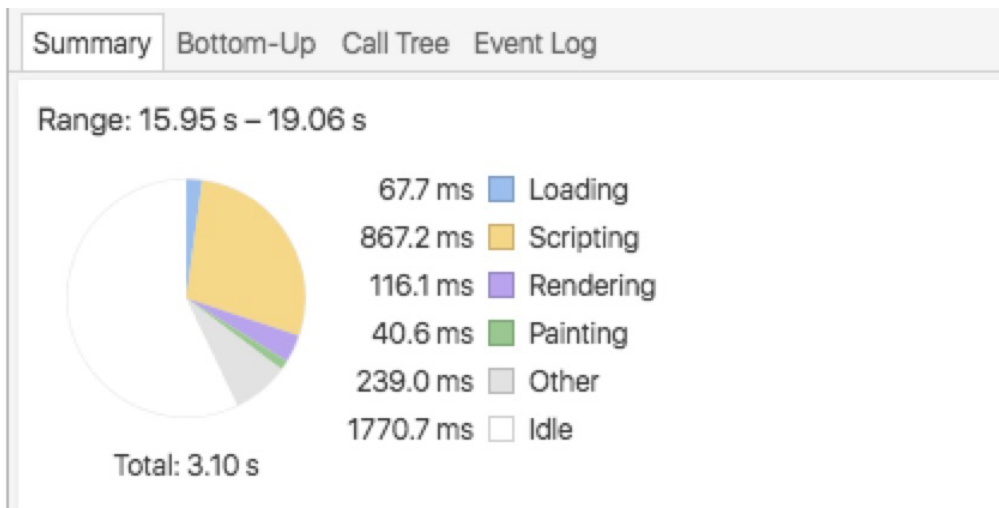
4.同时也希望大家能对CSS 更新的属性有认知<https://developer.mozilla.org/zh-CN/docs/Web/CSS/Reference>。

- 3.请写出你所知道的CSS3绘制特殊图形的技巧。(12分)

答：

🍊 这个题是考查的是基础属性的灵活应用，要熟练掌握。

- 1.border && border-radius 造就万千可能。
- 2.box-shadow是可以定义为任意颜色的，并且同一个元素可以投射出不同的box-shadow。
- 3.神奇borders构建的三角形。
- 4.别忘了还有CSS的裁切 clip:rect(0px,60px,200px,0px)。



- 4.如下是网页渲染的阶段汇总，请以此描述每一步代表意义。(5分)


- 答：

🍊 老弟 老妹 画个小球动画 点下performance然后录制，然后实战下行不。

- 1.loading 加载阶段 网络通信和html解析
- 2.scripting javascript执行
- 3.rendering 重排阶段 样式计算和布局 css
- 4.painting 重绘阶段
- 5.other 其他事件花费的时间
- 6.idle 空闲时间

- 5.请描述开发中如何减少或者避免网页重排和重绘 (20分)

- 答：

 高频考题。

1.重排必定会造成重绘 (如下条件都会造成重排)

- a、添加或者删除可见的DOM元素
- b、元素位置改变
- c、元素尺寸改变
- d、元素内容改变 (例如：一个文本被另一个不同尺寸的图片替代)
- e、页面渲染初始化 (这个无法避免)
- f、浏览器窗口尺寸改变

2.如果DOM变化仅仅影响的了背景色等等非几何属性

此时就发生了重绘 (repaint) 而不是重排

因为布局没有发生改变

3.开启动画加速的元素为什么就不会引起重排了呢？如下是合成层的步骤。

1.DOM子树渲染层 (RenderLayer) ->RenderObject->GraphicsContext

(根元素、position、transform、半透明、CSS滤镜、Canvas2D、video、溢出)


2.Compositor->渲染层子树的图形层 (GraphicsLayer) -> RenderLayer ->RenderObject

Compositor将所有的拥有compositing layer进行合成，合成过程GPU进行参与。合成完毕就能够将纹理映射到一个网格几何结构之上——在视频游戏或者CAD程序中，这种技术用来给框架式的3D模型添加“皮肤”。Chrome采用纹理把页面中的内容分块发送给GPU。纹理能够以很低的代价映射到不同的位置，而且还能够以很低的代价通过把它们应用到一个非常简单的矩形网格中进行变形。这就是3D CSS的实现原理。

(CSS3D透视变换、video、webgl、transform动画、加速CSS滤镜、叠加在已经触发合成层)

- 6.请描述一下BFC IFC GFC FFC具体描述和应用场景。 (10分)

- 答：

 高频考题。

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。常见的 Formatting context 有 BFC、IFC、GFC和FFC。

1.BFC(Box Formatting context): Box 是 CSS 布局的对象和基本单位，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。以下5种情况会生成 BFC：根元素、float属性不为none、position为absolute或fixed、overflow不为visible、display为inline-block, table-cell, table-caption, flex, inline-flex。

2. IFC(Inline Formatting Contexts)直译为“内联格式化上下文”，IFC的line box（线框）高度由其包含行内元素中最高的实际高度计算而来（不受到竖直方向的padding/margin影响）。会生成成为inline-level box:display 属性为 inline, inline-block, inline-table 的元素。


3.FFC(Flex Formatting Contexts)直译为"自适应格式化上下文"，display值为flex或者inline-flex的元素将会生成自适应容器（flex container）。

4.GFC(GridLayout Formatting Contexts)直译为"网格布局格式化上下文"，当为一个元素设置display值为grid的时候，此元素将会获得一个独立的渲染区域，我们可以通过在网格容器（grid container）上定义网格定义行（grid definition rows）和网格定义列（grid definition columns）属性各在网格项目（grid item）上定义网格行（grid row）和网格列（grid columns）为每一个网格项目（grid item）定义位置和空间。

5.那么问题来了BFC都能具体的到底解决哪些问题呢？？

- 7.请写出能够监测陀螺仪变化的事件？(15分)

- 答：

 这个题能体现出你平时做的CSS项目到底怎么样。

1. deviceorientation 设备的物理方向信息，表示为一系列本地坐标系的旋角。
2. devicemotion 提供设备的加速信息。
3. compassneeds Calibration 用于通知Web站点使用罗盘信息校准上述事件。


- 8.请写出你所知道的Next CSS的书写规则？（5分）

- 答：

 请默写。。。🚢

- 9.请解释一下预处理器和前处理器的作用和实现方式(10分)

- 答：

 这个题是基础。

1.预处理器处理特定格式源文件到目标css的处理程序

2.后处理主要是对CSS进行一些压缩以及浏览器之间的兼容处理。

3.当下最火的Post CSS前后通吃。

4.那么到底怎么配置嘞???请动一动你的手指。

- 10.请写出实现如下全景图的技术思路。

- 答：

 这个题考查的是css3d模型的构建。



1.计算好大小，分割图片。

2.根据公式进行拼接、旋转。

3.详解见实战五。具体的计算公式在这了。

- 附加题.请写出你对CSS Houdini的认识？

- 答：

