**task1 九宫格**

**日报：**

写在前面：年前到今天自己花了不少时间，但是毫无章法的乱学，学了不少概念，却大多只是走马观花，更是写得不熟练，因此希望通过修真院能重新按部就班，一步一步学习，巩固基础。

电脑上下了各种编辑器，IDE。觉得一般的页面sublime text3 和huilder 比较好用。

九宫格涉及到：

1. **布局**：

float布局或inline-block布局都可以，inline-block布局需要父元素设置font-size为0消除间隙。

1. **随浏览器大小变化的正方形:**

长宽需要相等，且要基于浏览器宽度变化(),宽度单位可以100%，height不行

需要用到padding，padding和margin的单位为%比时，都是基于父级的width。

1、自适应，viewport介绍

2、适配移动设备

4、怎么让div高度随浏览器宽度变换而变化

首先，如何让一个div的高度自动延伸到浏览窗口100%的高度,在布局中

html ,body{height:100%;}  
#wrapper（div）{height:100%;}

5

5、浏览器能拖拽的最小宽度

不同浏览器都有不同的最小宽度限制，不需要给body加min-width

1. **九宫格的布局你还能想到哪些办法实现？它们各自的优势劣势是什么？**

答：1、布局上,方格使用：

1）float

2）inline-block,设定父元素或body的font-size为0消除间隙

2、方格的高度:

**2.IDE是什么？它和文本编辑器相比有什么优缺点？**

IDE是用于提供程序开发环境的应用程序，一般包括代码[编辑器](http://baike.baidu.com/view/499644.htm)、[编译器](http://baike.baidu.com/view/487018.htm)、[调试](http://baike.baidu.com/view/322913.htm)器和[图形用户界面](http://baike.baidu.com/view/185360.htm)等工具

1. **HTML文件里开头的Doctype有什么作用**

DOCTYPE标签是一种标准通用标记语言的文档类型声明，它的目的是要告诉标准通用标记语言解析器，它应该使用什么样的文档类型定义（DTD）来解析文档4.如何理解盒模型

**5.常见的inline元素、block元素、inline-block元素有哪些？它们之间有什么区别**

**6.什么是****自适应，怎么去自适应？**

**7.加和不加meta的viewport有什么区别？**

**本例中，加不加没有区别**

**一、viewport的概念**

viewport是浏览器用来显示网页的那部分区域，但是viewport的大小又不局限于浏览器可视区域的大小，可能比浏览器的可视区域大，也可能小。

一般来讲，移动设备的viewport都要大于浏览器可视区域，

**二、css中的1px并不等于设备的1px**

css中的像素只是一个抽象的单位，在不同的设备或不同的环境中，css中的1px所代表的设备物理像素是不同的。

在早先的移动设备中，屏幕像素密度都比较低，如iphone3，它的分辨率为320x480，在iphone3上，一个css像素确实是等于一个屏幕物理像素的。后来随着技术的发展，移动设备的屏幕像素密度越来越高，从iphone4开始，苹果公司便推出了所谓的Retina屏，分辨率提高了一倍，变成640x960，但屏幕尺寸却没变化，这就意味着同样大小的屏幕上，像素却多了一倍，这时，一个css像素是等于两个物理像素的。

还有一个因素也会引起css中px的变化，那就是用户缩放。例如，当用户把页面放大一倍，那么css中1px所代表的物理像素也会增加一倍；反之把页面缩小一倍，css中1px所代表的物理像素也会减少一倍。

在移动端浏览器中以及某些桌面浏览器中，window对象有一个devicePixelRatio属性，它的官方的定义为：设备物理像素和设备独立像素的比例，也就是 devicePixelRatio = 物理像素 / 独立像素。css中的px就可以看做是设备的独立像素，所以通过devicePixelRatio，我们可以知道该设备上一个css像素代表多少个物理像素。

**三、PPK的关于三个viewport的理论**

ppk把这个浏览器默认的viewport叫做 layout viewport。这个layout viewport的宽度可以通过 document.documentElement.clientWidth 来获取。

  然而，***layout viewport***的宽度是大于浏览器可视区域的宽度的，所以我们还需要一个viewport来代表 浏览器可视区域的大小，ppk把这个viewport叫做 **visual viewport**。visual viewport的宽度可以通过window.innerWidth 来获取，但在Android 2, Oprea mini 和 UC 8中无法正确获取。

但浏览器觉得还不够，因为现在越来越多的网站都会为移动设备进行单独的设计，所以必须还要有一个能完美适配移动设备的viewport。所谓的完美适配指的是，首先不需要用户缩放和横向滚动条就能正常的查看网站的所有内容；第二，显示的文字的大小是合适，比如一段14px大小的文字，不会因为在一个高密度像素的屏幕里显示得太小而无法看清，理想的情况是这段14px的文字无论是在何种密度屏幕，何种分辨率下，显示出来的大小都是差不多的。当然，不只是文字，其他元素像图片什么的也是这个道理。ppk把这个viewport叫做 ideal viewport，也就是第三个viewport——移动设备的理想viewport。

ppk把移动设备上的viewport分为layout viewport 、 visual viewport 和 ideal viewport 三类，其中的ideal viewport是最适合移动设备的viewport，ideal viewport的宽度等于移动设备的屏幕宽度，只要在css中把某一元素的宽度设为ideal viewport的宽度(单位用px)，那么这个元素的宽度就是设备屏幕的宽度了，也就是宽度为100%的效果。ideal viewport 的意义在于，无论在何种分辨率的屏幕下，那些针对ideal viewport 而设计的网站，不需要用户手动缩放，也不需要出现横向滚动条，都可以完美的呈现给用户。

**四、利用meta标签对viewport进行控制**

<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0">

width=device-width和initial-scale=1.0

meta viewport 有6个属性(暂且把content中的那些东西称为一个个属性和值)，如下：

width 设置layout viewport 的宽度，为一个正整数，或字符串"width-device"

initial-scale 设置页面的初始缩放值，为一个数字，可以带小数

minimum-scale 允许用户的最小缩放值，为一个数字，可以带小数

maximum-scale 允许用户的最大缩放值，为一个数字，可以带小数

height 设置layout viewport 的高度，这个属性对我们并不重要，很少使用

user-scalable 是否允许用户进行缩放，值为"no"或"yes", no 代表不允许，yes代表允许