**适配**

**适配**:百分百还原设计图（等比）

没有加viewport meta 标签时：等比，但是字太小的

加了viewport meta之后就不等比了，占据的实际尺寸 (英寸)一样

**基础普及：**

* chrome默认的fontSize 最大为16px 最小12px

1. em参照于自己的fontSize，最小16，比他小的都要按照16去渲染
2. rem参照于根标签的fontSize，所以最大超过16px的按照16px，小于12px的只能用12px 就是不能大于16小于12

1rem=16px，

1. ie6最小高度为19px，在ie6下任何高度小于19px都会被渲染为19px，可以给父元素 fontSize为0，只能解决到2px

em参照于自己的fontSize，最小16，比他小的都要按照16去渲染

**rem适配**

* 在任何情况下都占据满屏
* 想让一个1rem的像素宽度在任何情况下都占据满屏
* 1rem最终渲染的值最终所渲染的px的值为布局视口
* 根标签的fontSize变为布局视口所代表的值

(function(){

var styleNode = document.createElement("style");

var width= document.documentElement.clientWidth/16;

styleNode.innerHTML="html{font-size:"+width+"px!important}"

document.head.appendChild(styleNode);

})()

创建style标签，获得当前视口的宽度，在标签中增加innerHtml

就是把宽度给html的fontsize，让1rem等于视口的大小，这样在写宽度的时候就可以与1比例参照，后面/16是为了让他没有小数

rem参照于根标签的fontSize-->1rem=16px-->1rem=视口宽度/16

**此时：**整个视口的宽度就为16em

我们设置样式时 就可以按照比例来赋值

比如width = 10em,此时宽度就占据总宽度的10/16。

本质：改变每个元素在不同设备上锁占据的css像素的个数

优点：使用的完美视口

缺点：rem的转化比较复杂

**viewport适配**

**基础普及：**

2x—是从750上面截下来的图

3x---是从1224截下来的图

一般要750的设计图

**需求：**

**当设计图就是320时，把所有的设备布局视口都调成320**

**这样测量了多少就可以写多少，等比方案动的是mate 的sclae缩放视口**

**viewport适配 将每个设备的布局视口变成设计图的尺寸**

**优点：所量既所得**

**缺点：没有使用完美视口 在大项目可能会出错**

(function(){

var targetW = 640;

var scale = document.documentElement.clientWidth/targetW;

var metaNode = document.querySelector("meta[name='viewport']");

metaNode.setAttribute("content","initial-scale="+scale+",user-scalable=no");

})

// targetW这个是设计图给的图片所要适配的尺寸

// scale获得当前所要缩放的的大小，要用ClientWidth获取视口大小的的前提，先在头上写好理想视口才能用，在后面再修改

//获得这个属性，然后在后面设置这个属性

**实现1物理像素的适配方案**

(function(){

//主体的适配使用rem适配方案

var dpr = window.devicePixelRatio||1;

var styleNode = document.createElement("style");

var width = document.documentElement.clientWidth\*dpr/16;

styleNode.innerHTML="html{font-size:"+width+"px!important}";

document.head.appendChild(styleNode);

//1物理像素

var scale = 1/dpr;

var metaNode = document.querySelector("meta[name='viewport']");

metaNode.setAttribute("content","width=device-width,initial-scale="+scale+",user-scalable=no");

})()

drp设置了缩放 然后在主体设置的时候把它给恢复，这样宽度的em就可以不变 px实现了像素