

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，
 3. 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 2pxem 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");
})();
```

dpr 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