

02

OPEN ORIENTED

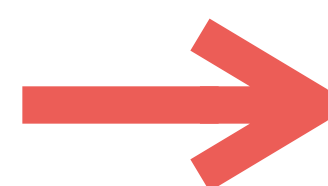
凹凸实验室

关于视口单位，能带来些什么

Tingglelao

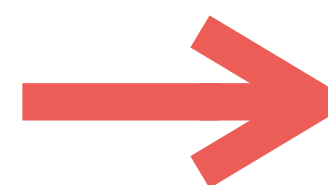
适配手法

Media Queries

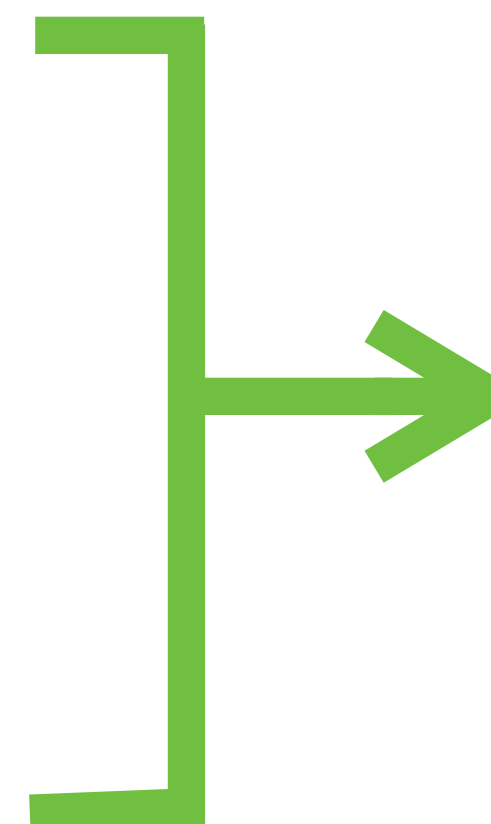


“断层”

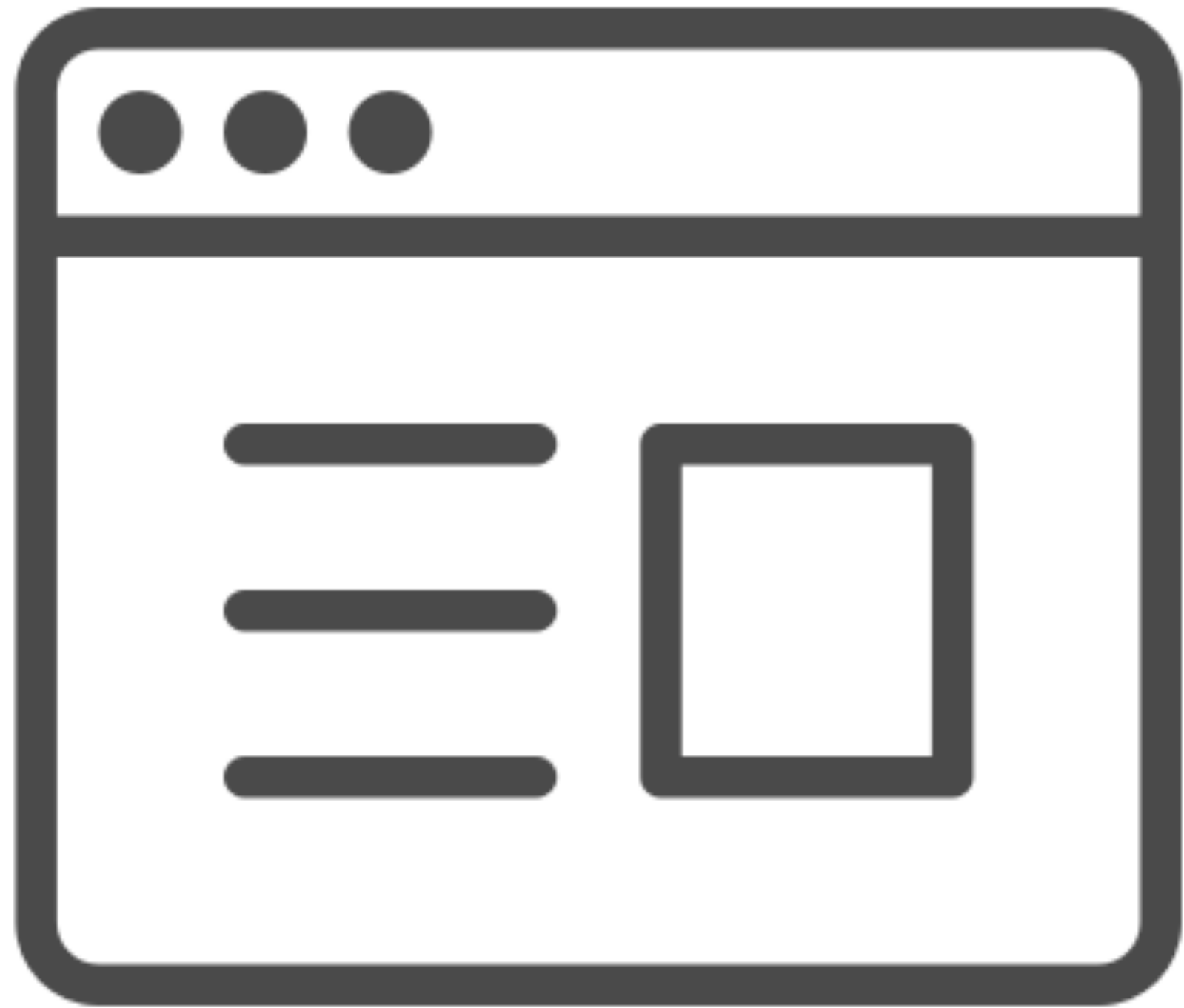
REM



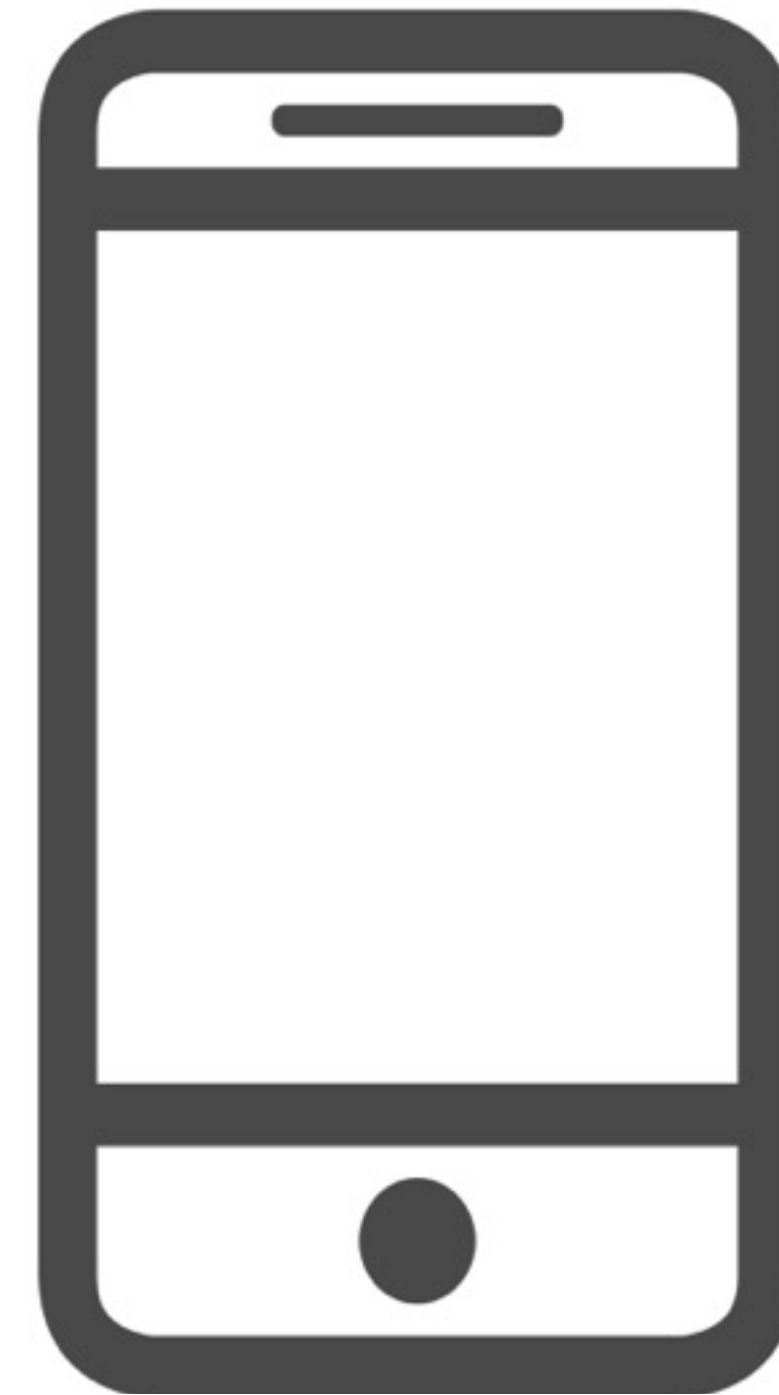
内嵌脚本



视口单位



桌面端 — 浏览器可视区域



移动端 — Layout Viewport

认识视口单位

视口单位

vw

1vw等于视口宽度大小的1%

vh

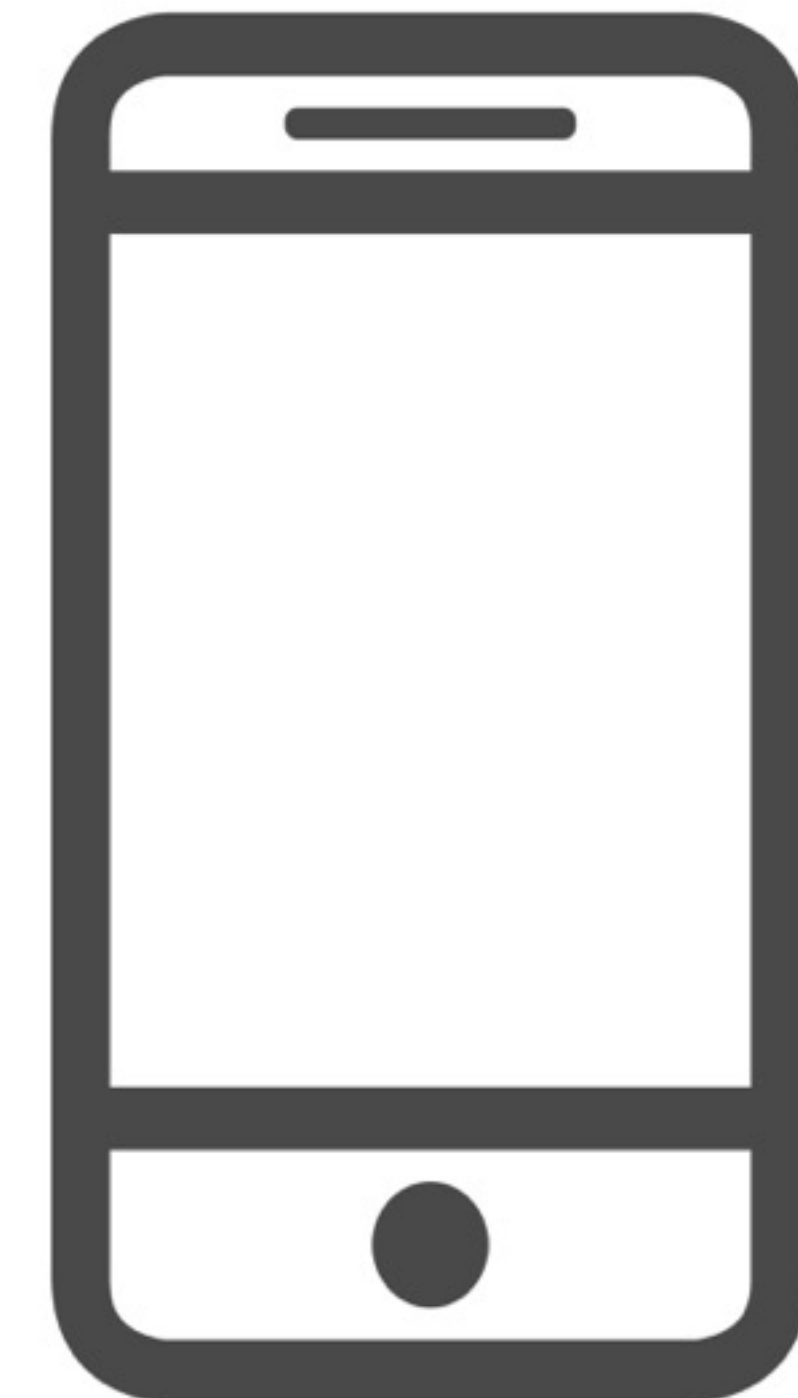
1vh等于视口高度大小的1%

$vmin$

选取100vw和100vh中最小的那个值

$vmax$

选取100vw和100vh中最小的那个值



100vh

100vw

[illegible]

视口单位的兼容性- X5 CAN I USE

Viewport units
40
42
6.0
6.1
6.2
6.3
6.6
1.4
2.0

利用视口单位实现布局 — 只用vw

1

无论是文本还是布局高宽、间距等都使用vw作为单位

2




1物理像素线采用transform属性scale实现

3

对于需要保持高宽比的图，改用padding-top实现。

```
1 $vm_base: 375; // 设计稿基准
2 @function vw($px) {
3     @return ($px / 375) * 100vw;
4 }
```

利用视口单位实现布局 — 只用vw

-  根据视口大小而自动缩放，但也无最大最小宽度限制
-  违背大屏下展示更多文字信息的原则
-  同样地也带来小数像素问题

利用视口单位实现布局



DEMO

利用视口单位实现布局 – vw搭配rem

1

<html>根元素字体大小使用vw作为单位

2

无论是文本还是布局高宽、间距等都使用rem作为单位

3

1物理像素线采用transform属性scale实现

4

对于需要保持高宽比的图，改用padding-top实现。

```
1 $vw_base: 375;
2 $vw_fontsize: 75;
3 // 通用的rem单位换算
4 @function rem($px) {
5     @return ($px / $vw_fontsize) * 1rem;
6 }
7
8 // 根元素大小
9 html {
10     font-size: ($vw_fontsize / $vw_base) * 100vw;
11 }
12
13 // 最大宽度和最小宽度限制
14 html {
15     @media screen and (max-width: 320px) {
16         font-size: 64px;
17     }
18     @media screen and (min-width: 540px) {
19         font-size: 108px;
20     }
21 }
22
23 body {
24     max-width: 540px;
25     min-width: 320px;
26 }
```

利用视口单位实现布局 — vm搭配rem



~~根据视口大小而自动缩放，但也无最大最小宽度限制~~



违背大屏下展示更多文字信息的原则



同样地也带来小数像素问题

利用视口单位实现布局



DEMO

给我们带来什么



如何无缝切换

02

Sass 函数替换



```
// 媒体查询
// 使用: @include query(320px) {...};
@mixin query( $limit ) {
  @media screen and ( min-width: $limit ) {
    & {
      @content;
    }
  }
}

// rem 函数
// 使用: width: rem(100);
$rem_grid: 20 !default;
@function rem($val){
  $rem : $val / $rem_grid;
  @if $rem == 0 {
    @return #{ $rem };
  }@else{
    @return $rem / ($rem * 0 + 1) * 1rem; // 带不带px单位都支持
  }
}

// rem 媒体查询
// 使用: html {@include rem_query; 或 @include rem_query(320);}
// _mod_cross 项目无需引用
@mixin _mod_cross( $width, $page_w ) {
  @include query( $width * 1px ) {
    font-size: $width / $page_w * $rem_grid * 1px;
  }
}

@mixin rem_query($page_width: 375) {
  font-size: $rem_grid * 1px;
  @include _mod_cross(320, $page_width); // iPhone5
  @include _mod_cross(360, $page_width); // Nexus5
  @include _mod_cross(375, $page_width); // iPhone6
  @include _mod_cross(384, $page_width); // Nexus4
  @include _mod_cross(411, $page_width); // Nexus6
  @include _mod_cross(414, $page_width); // iPhone6+
  @include _mod_cross(480, $page_width); // Samsung Galaxy Note 4
  @include _mod_cross(540, $page_width); // Sony Xperia C4
}
```

```
html {
  @include rem_query;
}
```

```
1 $vw_base: 375;
2 $vw_fontsize: 75;
3 // 通用的rem单位换算
4 @function rem($px) {
5     @return ($px / $vw_fontsize ) * 1rem;
6 }
7
8 // 根元素大小
9 html {
10     font-size: ($vw_fontsize / $vw_base) * 100vw;
11 }
12
13 // 最大宽度和最小宽度限制
14 html {
15     @media screen and (max-width: 320px) {
16         font-size: 64px;
17     }
18     @media screen and (min-width: 540px) {
19         font-size: 108px;
20     }
21 }
22
23 body {
24     max-width: 540px;
25     min-width: 320px;
26 }
```

如何无缝切换



DEMO

体验优化



DEMO

T H A N K S
FOR YOUR WATCHING

