

1:移动端你们一般采用什么布局？移动端设计稿是多大的尺寸？

- 定宽布局
- 一般移动端设计稿是640或者750的尺寸

2:em和rem的区别

- em相对父级元素设置的font-size来设置大小 如果父元素没有设置font-size，则继续向上查找，直至有设置font-size元素
- rem直接参照html标签字体大小，并且所有使用rem单位的都是参照html标签

3:移动端用过那些meta标签？

```
<!--1:设置视口宽度 缩放比例-->
<meta content="width=device-width, initial-scale=1.0, maximum-scale=1.0,
user-scalable=0" name="viewport">

<!--2:忽略将数字变为电话号码-->
<meta content="telephone=no" name="format-detection">

<!--3:忽略识别邮箱-->
<meta name="format-detection" content="email=no" />

<!--4:IOS中Safari允许全屏浏览-->
<meta content="yes" name="apple-mobile-web-app-capable">
```

4:移动端click 300毫秒延迟原因？

移动端浏览器会有一些默认的行为，比如双击缩放、双击滚动。这些行为，尤其是双击缩放，主要是为桌面网站在移动端的浏览体验设计的。而在用户对页面进行操作的时候，移动端浏览器会优先判断用户是否要触发默认的行为。

5:移动端zepto中的tap事件点透问题?

问题点击穿透问题：点击蒙层（mask）上的关闭按钮，蒙层消失后发现触发了按钮下面元素的click事件

zepto的tap事件是绑定到document的，所以一般点击tap事件都会冒泡到document才会触发。当点击隐藏蒙层的时候默认也会手指触发到蒙层下面的元素 执行事件

1. github上有个fastclick插件，用来规避click事件的延时执行

2. 用touchend代替tap事件并阻止掉touchend的默认行为preventDefault()

```
//tap事件出现点透问题
$("#id").on("tap", function (event) {
    //很多处理比如隐藏什么的
    event.preventDefault();
});

//touchend事件解决点头问题
$("#id").on("touchend", function (event) {
    //很多处理比如隐藏什么的
    event.preventDefault();
});
```

3:给tap事件里面的隐藏蒙层方法执行的方法300毫秒延迟

```
$("#id").on('tap',function(ev){
    setTimeout(function(){
        $("#id").hide();
    },320)
})
```

6:固定定位布局 键盘挡住输入框内容?

1:通过绑定窗口改变事件,监听键盘的弹出。然后去改变固定定位元素的位置。默认键

盘的宽度应该是页面的2分之一。所以我们位移的距离改成键盘的二分之一就可以

```
window.onresize = function(){
    //$(".mian")就是固定定位的元素
    if($(".mian").css('top').replace('px', '') != 0){
        $(".mian").css('top', 0);
    }else{
        var winHeight = $(window).height();
        $(".mian").css('top', -(winHeight/4));
    }
}
```

2:通过定时器实时监听是否触发input。如果触发input框 就把固定定位, 改变成静态定位。这样就会浏览器会总动把内容顶上去。

```
function fixedWatch(el) {
    //activeElement 获取焦点元素
    if(document.activeElement.nodeName == 'INPUT') {
        el.css('position', 'static');
    } else {
        el.css('position', 'fixed');
    }
}

setInterval(function() {
    fixedWatch($(".mian"));
}, 500);
```

7: 移动端兼容问题?

1:移动端默认的input和按钮样式还有点击出现的样式不一样 需要默认去清除一些默认样式

```
//去掉webkit的滚动条——display: none;
//其他参数
::-webkit-scrollbar //滚动条整体部分
::-webkit-scrollbar-thumb //滚动条内的小方块
::-webkit-scrollbar-track //滚动条轨道
::-webkit-scrollbar-button //滚动条轨道两端按钮
::-webkit-scrollbar-track-piece //滚动条中间部分, 内置轨道
```

```

::-webkit-scrollbar-corner          //边角，两个滚动条交汇处
::-webkit-resizer                   //两个滚动条的交汇处上用于通过拖动调整元素大小的小控件

// 禁止长按链接与图片弹出菜单
a,img { -webkit-touch-callout: none }

// 禁止ios和android用户选中文字
html,body { -webkit-user-select:none; user-select: none; }

// 改变输入框placeholder的颜色值
::-webkit-input-placeholder { /* WebKit browsers */
color: #999; }
:-moz-placeholder { /* Mozilla Firefox 4 to 18 */
color: #999; }
:-moz-placeholder { /* Mozilla Firefox 19+ */
color: #999; }
:-ms-input-placeholder { /* Internet Explorer 10+ */
color: #999; }
input:focus::-webkit-input-placeholder{ color:#999; }

// android上去掉语音输入按钮
input::-webkit-input-speech-button {display: none}

// 阻止windows Phone的默认触摸事件
/*说明：winphone下默认触摸事件使用e.preventDefault是无效的，可通过样式来禁用，如：*/
html { -ms-touch-action:none; } //禁止winphone默认触摸事件

//ios用户点击一个链接，会出现一个半透明灰色遮罩，如果想要禁用，可设置-webkit-tap-highlight-color的alpha值为0去除灰色半透明遮罩；
//android用户点击一个链接，会出现一个边框或者半透明灰色遮罩，不同生产商定义出来效果不一样，可设置-webkit-tap-highlight-color的alpha值为0去除部分机器自带的效果；
//winphone系统,点击标签产生的灰色半透明背景，能通过设置<meta name="msapplication-tap-highlight" content="no">去掉；
//特殊说明：有些机型去除不了，如小米2。对于按钮类还有个办法，不使用a或者input标签，直接用div标签
a,button,input,textarea {
    -webkit-tap-highlight-color: rgba(0,0,0,0);
    -webkit-user-modify:read-write-plaintext-only; // -webkit-user-modify有个副作用，就是输入法不再能够输入多个字符
}
// 也可以

```

```
* { -webkit-tap-highlight-color: rgba(0,0,0,0); }  
//winphone下  
<meta name="msapplication-tap-highlight" content="no">
```

2:手机拍照和上传图片

```
/*  
*IOS有拍照、录像、选取本地图片功能,  
*部分Android只有选择本地图片功能。  
*Winphone不支持  
*/  
<input type="file" accept="images/*" />
```

3:消除transition闪屏

```
.css {  
  -webkit-transform-style: preserve-3d;  
  -webkit-backface-visibility: hidden;  
  -webkit-perspective: 1000;  
}
```

=====一下内容与上无关=====

//目前，像Chrome/Filefox/Safari/IE9+以及最新版本Opera都支持硬件加速，当检测到某个DOM元素应用了某些CSS规则时就会自动开启，从而解决页面闪白，保证动画流畅。

```
.css {  
  -webkit-transform: translate3d(0,0,0);  
  -moz-transform: translate3d(0,0,0);  
  -ms-transform: translate3d(0,0,0);  
  transform: translate3d(0,0,0);  
}
```

4:固定定位相关问题?

1. 固定定位会产生 键盘弹出遮住固定元素里面元素的input框
2. 固定定位的遮罩层弹出时 滚动到最底部 qq和uc浏览器会出现遮罩层下面的样式露出来

5: 在ios上写有邮箱文本点击这个邮箱地址不会有反应，安卓手机上点击邮箱会自动打开本地邮箱软件。