

移动端讲义案例

移动端开发

把我们做好的产品放在手机(PAD)上观看 => 响应式
布局开发

- 基于现有的PC端产品做样式和布局的调整，让其在手机端也可以很好的观看（一般情况下PC和移动端共用一套项目）
- PC端和移动端是两套项目，PC就是PC的，不需要做响应式，而移动端需要做响应式也是一套单独的项目
 - 基于手机的浏览器输入网址或者扫描二维码访问我们开发的页面
 - 在微信/QQ/微博等第三方平台中运行
 - 嵌入到自己公司的APP中（ hybrid混合开发模式 ）

[https://mp.weixin.qq.com/wiki?
t=resource/res_main&id=mp1421141115](https://mp.weixin.qq.com/wiki?t=resource/res_main&id=mp1421141115)

HTML5

HTML的第五套版本，在传统的HTML基础上新增加或者更新了一些新特性 (HTML5秘籍第二版)

- 对语义化标签的处理
 - 新增加语义化标签：header、footer、nav、section、main、article、aside、figure、figcaption、hgroup、progress、time、mark(高亮选中)、datalist、summary...
 - 对传统标签的语义修改：small(附属细则)、strong(重点朗读)、hr(区域分割线)....
 - 删除一些标签：font、big...

```
1. /*以上标签不兼容，不兼容的情况下我们引入 ht  
   ml5.min.js来处理兼容*/  
2. <!--[if lt IE 9]>  
3. <script src='js/html5.min.js'></scrip  
   t>  
4. <![endif]-->
```

- 给传统的表单元素新增加了很多的类型
 - 传统类型：text、password、radio、checkbox、submit、button、reset、file、hidden...
 - 新增加类型(不兼容IE低版本)：search、email、tel、number、range、color、date...

```
1.  /*
2.   * 新增加类型的优势
3.   * 1、在手机端可以调取出最符合用户输入的键盘(例如：类型是number调取出来的就是数字键盘...)
4.   * 2、内置的验证机制(对于邮箱电话类型等，浏览器有内置的正则验证)
5.   * 3、提供了强大的功能组件，例如：日历、滑动杆、颜色选择面板等
6.   * 4、placeholder属性也是新增加的，默认显示提示文字
7.  */
8.  input:valid{
9.      //->合法的
10. }
11. input:invalid{
12.     //->非法的
13. }
14.
15. 移动端的表单，没有keydown/keyup等事件（因为移动端的键盘是虚拟键盘），我们统一使用input事件来代替上述事件操作
16. oInput.addEventListener('input',function(){
17.     //<js code> 用户输入的过程中触发这个事件
18. },false);
```

- 新增音视频处理
 - 音频:audio
 - 视频:video
 - 在IE低版本中依然采用传统的FLASH播放
 - 真实项目中的音视频播放，我们都是采用别人写好的播放器插件来处理的(兼容VIDEO/兼容FLASH)
- canvas / svg
- 一些新兴的API(JS)
 - 本地存储：webStorage (localStorage、sessionStorage)
 - 检测地理位置信息的
 - web socket
 - web works
 - 离线存储
 - ...

CSS3

CSS第三个版本，在传统的CSS基础上新增加很多的属性和内容 (图解CSS3 ->大漠)

<http://www.w3cplus.com/>

- 新增选择器
 - 结构伪类选择器: `:nth-child`、`:nth-last-child`、`:nth-of-type`、`:nth-last-of-type`、`:not...`

- 目标伪类选择器：`:target`
- 属性选择器：`[attr=xxx]`、`[attr!=xxx]`、`[attr^=xxx]`、`[attr$=xxx]`、`[attr*=xxx]`、`[attr~=xxx]...`
- ...
- 新增一些基本的样式属性
 - `border-radius`
 - `box-shadow`
 - `text-shadow`
 - ...
- 对背景的改革
 - `background-size`
 - `background: linear-gradient / radial-gradient...`
 - `background-clip`
 - `background-origin`
 - `background-attachment: fixed` (背景图铺在第一屏幕下，页面滚动背景图始终定位在当前窗口下)
 - ...
- `transform`
 - `transform-style: preserve-3d`
 - `transform-origin`
 - `translate(X|Y|Z)`
 - `rotate(X|Y|Z)`
 - `scale`
 - `skew`

- matrix
- ...
- transition
 - transition-property : all...
 - transition-duration
 - transition-timing-function : linear、ease、ease-in、ease-out、ease-in-out...
 - transition-delay
- animation
 - @keyframes 设置运动的轨迹
 - animation-name
 - animation-duration
 - animation-timing-function
 - animation-delay
 - animation-iteration-count : **infinite**无限次
 - animation-fill-mode : **forwards**(运动完成后停留在最后一帧) **backwards**(在延迟等待时间内, 元素始终处于当前动画的第一帧位置) **both**(同时具备以上两个特点)
 - animation-play-state : running、paused
 - ...
- 新盒子模型
 - box-sizing : border-box、content-box
 - columns
 - flexbox

- @font-face : 字体图标(导入外部字体)
- @media : 媒体查询
- ...

viewport

常用手机的尺寸

苹果 : 320(<=iphone5s)、 375(iphone6)、
414(iphone6 plus)...

安卓 : 360、 480、 540、 640...

默认情况下手机上的HTML页面宽度默认都是
980(1024)的宽度，和手机多宽没有关系

如果想在320的手机上展示980的页面，需要把页面
整体缩小大概三倍左右（需要放大自己使用手指缩
放）

```
1. <meta name="viewport"  
2. content="  
3.     width=device-width,  
4.     user-scalable=no,  
5.     initial-scale=1.0,  
6.     maximum-scale=1.0,  
7.     minimum-scale=1.0">
```

真实项目中我们不想让页面缩小，也不想让用户手指缩放，此时我们需要让HTML页面的宽度和手机宽度保持一致：设置页面的 `viewport` (layout viewport布局视口)

`width=device-width`: 当前HTML页面的宽度等于手机设备的宽度

`user-scalable=no`: 禁止用户手指手动缩放

`initial-scale=1.0/maximum-scale=1.0/minimum-scale=1.0`: 初始、最大、最小缩放比例都是1:1，也就是当前页面既不会放大也不会缩小(兼容部分安卓机器)

以后编写的HTML页面如果想在手机上访问，必须加 `viewport`