[Bootstrap 1](#_Toc1185)

[jQuery Mobile 7](#_Toc14108)

[jQuery Mobile是什么？ 7](#_Toc31531)

[为什么使用jQuery Mobile？ 7](#_Toc1263)

[过渡效果： 10](#_Toc27772)

[创建按钮： 10](#_Toc21304)

[组合按钮，后退按钮 11](#_Toc784)

[工具栏 13](#_Toc21203)

[向标题右侧添加按钮： 13](#_Toc25430)

[定位页眉页脚 13](#_Toc6485)

[表单 23](#_Toc19437)

[表单，滑块： 26](#_Toc21719)

[事件 27](#_Toc15541)

[html5 35](#_Toc3897)

[coords属性： 36](#_Toc901)

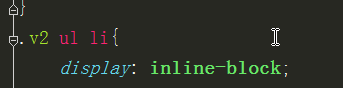
[字体自适应: 42](#_Toc1157)

# Bootstrap

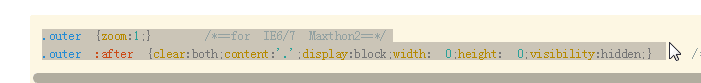
模板：

<**meta** charset=**"UTF-8"**>  
<**meta** http-equiv=**"X-Ua-compatible"** content=**"ie=edge"**>  
*<!--兼容ie8-->*<**meta** name=**"viewport"** content=**"width=device-width,initial-scale=1"**>  
*<!--响应式兼容-->*<**title**>Bootstrap FirstDay</**title**>  
<**link** rel=**"stylesheet"** href=**"bootstrap-3.3.5-dist/css/bootstrap.css"**>  
*<!--引入bootstrap的css文件-->* <**link** rel=**"stylesheet"** href=**"allcss.css"**>  
<**script** src=**"jquery-3.1.0.min.js"**></**script**>  
*<!--引入与bootstrap相匹配的jQuery文件-->*<**script** src=**"bootstrap-3.3.5-dist/js/bootstrap.js"**></**script**>  
*<!--引入bootstrap的js文件-->*

1



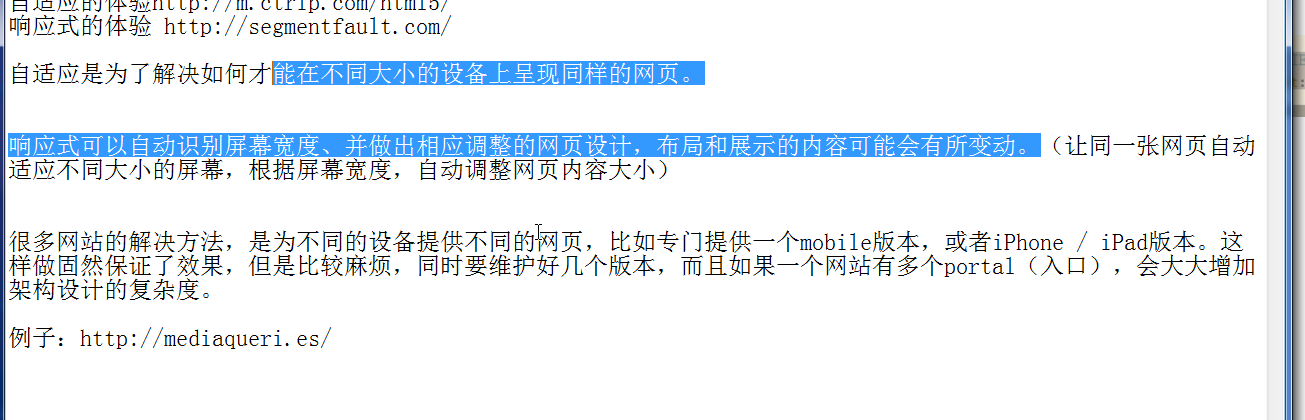
2 清除浮动



3

input 背景颜色

input 边框颜色

4 自适应与响应式

自适应可以自动识别屏幕宽度，并作出相应调整的网页设计，布局和展示的内容可能有所变动。让同一张网页自动适应不同大小的屏幕，

5 bootstrap



6

顺序：jquery 必须在其他js文件之前

解决ie8 兼容

自适应兼容

Bootstrap 的css

bootstrap 的 js

0 所有设置了 .form-control 类的 <input>、<textarea> 和 <select> 元素都将被默认设置宽度属性为 width: 100%;。

0 .container 与 .container-fluid不能互相嵌套

container 居中.

container-fluid 类用于 100% 宽度，占据全部视口（viewport）的容器。

1 “行（row）”必须包含在 .container （固定宽度）或 .container-fluid （100% 宽度）中，以便为其赋予合适的排列（aligment）和内补（padding）。

2 你的内容应当放置于“列（column）”内，并且，只有“列（column）”可以作为行（row）”的直接子元素。

3 可以为“列（column）”，“行(row)”设置 padding 属性和margin属性；

4 如果一“行（row）”中包含了的“列（column）”大于 12，多余的“列（column）”所在的元素将被作为一个整体另起一行排列。

5 超小屏幕（手机，小于 768px）没有任何媒体查询相关的代码，因为这在 Bootstrap 中是默认的（还记得 Bootstrap 是移动设备优先的吗？）

6 栅格参数

通过下表可以详细查看 Bootstrap 的栅格系统是如何在多种屏幕设备上工作的。

|  | **超小屏幕 手机 (<768px)** | **小屏幕 平板 (≥768px)** | **中等屏幕 桌面显示器 (≥992px)** | **大屏幕 大桌面显示器 (≥1200px)** |
| --- | --- | --- | --- | --- |
| **栅格系统行为** | 总是水平排列 | 开始是堆叠在一起的，当大于这些阈值时将变为水平排列C | | |
| .container**最大宽度** | None （自动） | 750px | 970px | 1170px |
| **类前缀** | .col-xs- | .col-sm- | .col-md- | .col-lg- |
| **列（column）数** | 12 | | | |
| **最大列（column）宽** | 自动 | ~62px | ~81px | ~97px |
| **槽（gutter）宽** | 30px （每列左右均有 15px） | | | |
| **可嵌套** | 是 | | | |
| **偏移（Offsets）** | 是 | | | |
| **列排序** | 是 | | | |

\* 唯独xs总是水平排列

其他的 会堆积

7 列都有一个padding:0 15px

8 class="lead" 可以让段落突出显示。

9 <mark>标签添加文本背景

10 <del> 被删除的文本使用，必须被div包裹，否则不会显示（虽然不显示但还是会占空间）

11 <s> 标签，必须被div包裹，否则不会显示（虽然不显示但还是会占空间）

12 <ins> 标签 额外插入的文本使用，必须被div包裹，否则不会显示（虽然不显示但还是会占空间）

13 首字母大写 text-capitalize

全部字母小写text-lowercase

全部字母大写text-uppercase

14 <abbr> 标签： 缩略语

与title属性结合使用；鼠标悬停在缩写和缩写词上时就会显示完整内容

15 .dl-horizontal 属性：

使文本排列在一行，并可以随着屏幕宽度的改变而改变列数

不实现多行排列

16 text-overflow 属性：

水平排列的描述列表将会截断左侧太长的短语。

可以实现多行排列(无论窗口的大小)，同样可以随着屏幕宽度的改变而改变列数

不实现多行排列

17 <code> 标签：

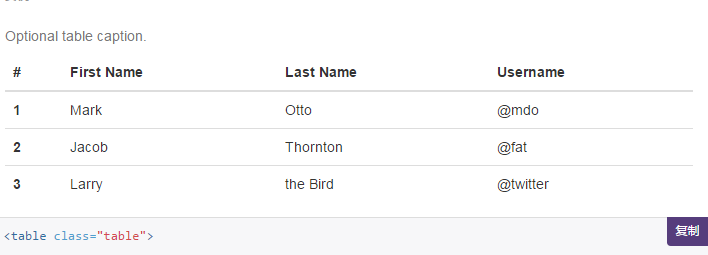
给文本增加淡粉色的背景颜色，以及2px的圆角边框

18 <kbd> 标签：

给文本增加深灰色，接近黑色的背景颜色，以及2px的圆角边框

表格：

19 <table>标签：



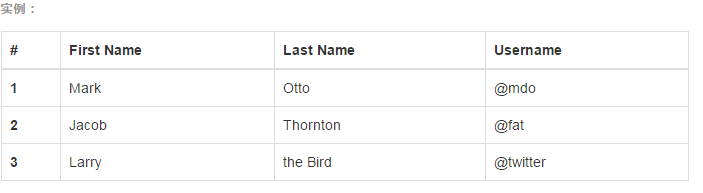
20 .table-striped属性：

给 <tbody> 之内的每一行增加斑马条纹样式。

好似：第一行x色，第二行y色，第三行x色，第四行y色如此轮回

21 .table-bordered 属性：

为表格和其中的每个单元格增加边框。



22 关于表格的其他类:

| **Class** | **描述** |
| --- | --- |
| .active | 鼠标悬停在行或单元格上时所设置的颜色 |
| .success | 标识成功或积极的动作 |
| .info | 标识普通的提示信息或动作 |
| .warning | 标识警告或需要用户注意 |
| .danger | 标识危险或潜在的带来负面影响的动作 |

表单：

23 所有设置了 .form-control 类的 <input>、<textarea> 和 <select> 元素都将被默认设置宽度属性为 width: 100%;。

24 响应式 设置临界最小宽度

25 【!!】<script>

$('.carousel').carousel({

interval: 500

})

</script>

设置轮播时间

问题1   
 <**div** class=**"col-md-4"**>.col-md-4</**div**>

会堆积

但是html中还有其他的col，且页面缩小到992px以下时，找不到  
 <**div** class=**"col-md-4"**>.col-md-4</**div**>

【】自己写的css样式，不能给div加高度，限制了高度的话，超出的元素会被隐藏，

貌似在bootstrap中，所有的元素都必须被包裹在div标签中

？禁止响应式布局

老师的 bootstrap 文档 笔记

# jQuery Mobile

## jQuery Mobile是什么？

jQuery Mobile 是创建移动web应用程序的框架

jQuery Mobile 适用于所有流行的智能手机和平板电脑。

jQuery Mobile 使用HTML5 和CSS3 通过尽可能少的脚本对页面进行布局

## 为什么使用jQuery Mobile？

jQuery Mobile 将”写的更少，做的更多”这一理念提升到了新的层次；

它会自动为网页设计交互的易用外观，并在所有移动设备商保持一致。

不需要为每种移动设备或OS 编写一个应用程序

使用了css，html，js技术

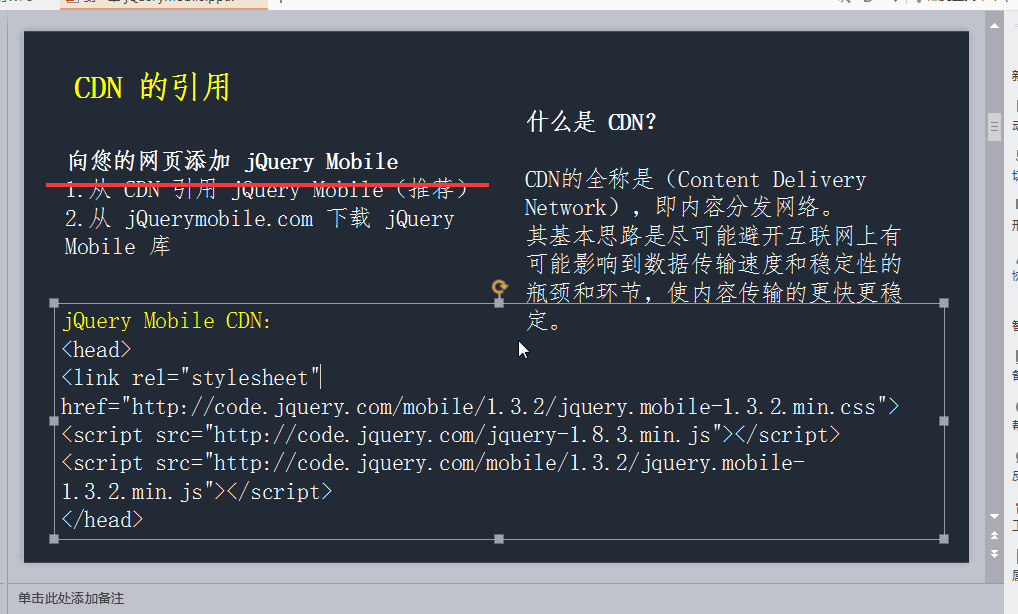
\*在pc上存在兼容性问题，因为css3并不是每个浏览器都支持

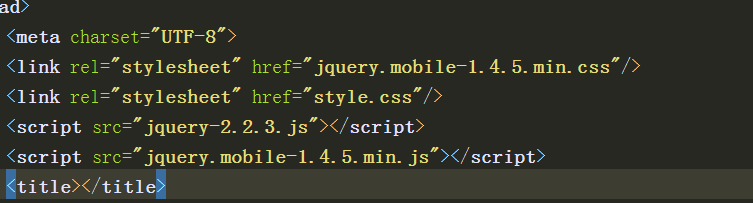
小知识：

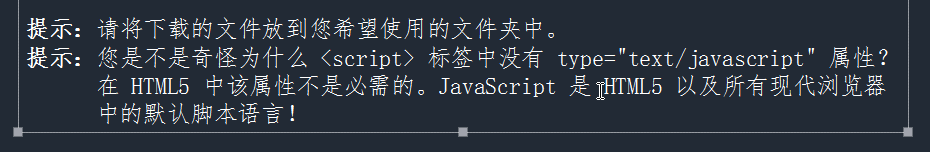
Android 和 Blackberry(黑莓) 用Java 编写

IOS 用 Objective 用C编写

Window Phone 用C# 和 .net 编写

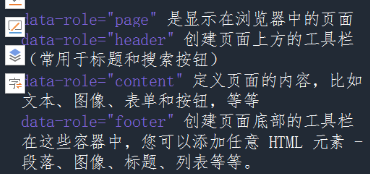


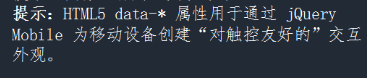




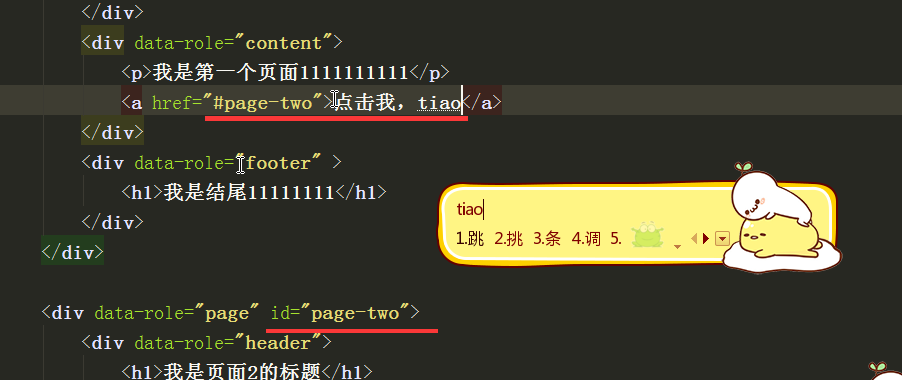
一个页面只能出现一次 （以上四个div）

但是一个HTML可以出现多个页面

date-role = “page|header|content|footer|... ...”







将页面用作对话框：

向a链接 添加一个data-rel:"dialog"

(甚至可以弹出一个对话框形式的页面)

## 过渡效果：

**格式：data-transition = “”**

\*淡入淡出效果是默认的：

fade 默认，淡出淡出到下一页

flip 从后向前翻动到下一页。

flow 抛出当前页面，引入下一页面。

pop 像弹窗口一样转到下一页

slide 从右向左滑动到下一页

slidefade 从右向左滑动并淡入到下一页

slideup 从下到上滑动到下一页

slidedown从上到下滑动到下一页

turn 转向下一页

none 无过渡效果

\*如果希望反向效果:

需要在后面增加data-direction:”reverse”

(这个属性在后退效果上是默认的)

## 创建按钮：

三种方法：

1 <button></>

2 <input type=”button” name=”” value=””>

以上2种用于表单提交；

3 data-role=”button”的<a>元素 —— **推荐**

导航和页脚内的a标签其实默认显示的就是按钮形式，且自适应大小；

其他 的：

默认情况下会占据屏幕的全部宽度，必须加date-role=”button”

如果需要自适应大小显示：

data-inline=”true”



data-corners true | false 规定按钮是否有圆角。

data-mini true | false 规定是否是小型按钮。

data-shadow true | false 规定按钮是否有阴影。

### 组合按钮，后退按钮

组合按钮

将：data-role="controlgroup"与data-type="horizontal|vertical"共同使用

规定水平或者垂直组合按钮；

后退按钮

a标签，使用data-rel="back"属性(会忽略锚的href值)

按钮的属性：

data-corners true|false 规定是否有圆角

data-mini true|false 规定是否有小型按钮

data-shadow true|false 规定按钮是否有阴影



属性值 描述

data-icon="arrow-l" 左箭头

data-icon="arrow-r" 右箭头

data-icon="arrow-u" 上箭头

data-icon="arrow-d" 下箭头

data-icon="plus" 加

data-icon="minus" 减

data-icon="delete" 删除

data-icon="check" 检查

data-icon="home" 首页

data-icon="info" 信息

data-icon="grid" 网格

data-icon="gear" 齿轮

data-icon="search" 搜索

data-icon="back" 后退

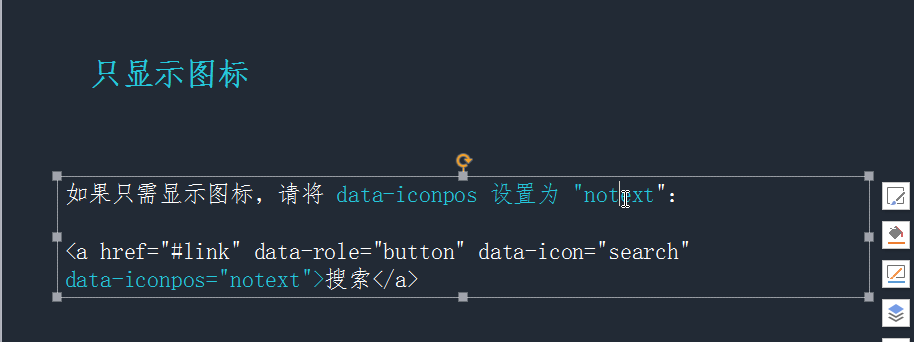
data-icon="forward" 向前

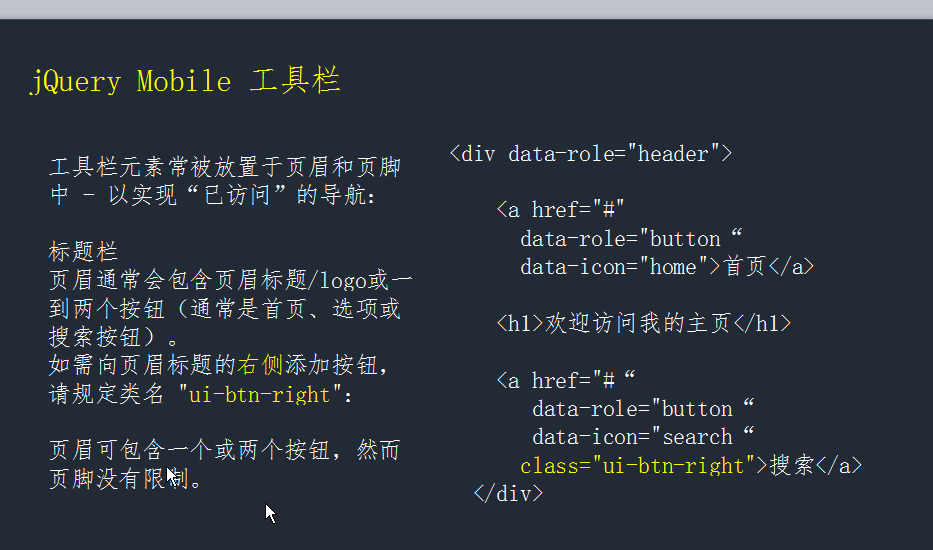
data-icon="refresh" 刷新

data-icon="star" 星

data-icon="alert" 提醒



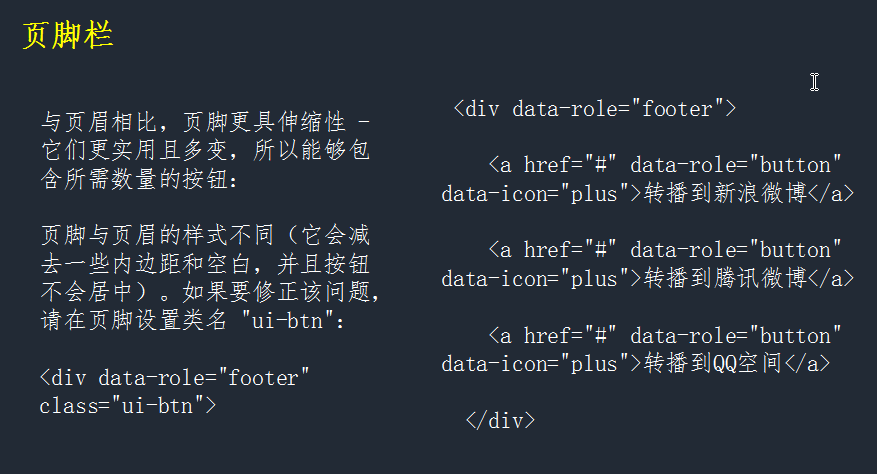




## 工具栏

### 向标题右侧添加按钮：

规定**类名:** ui-btn-right



## 定位页眉页脚

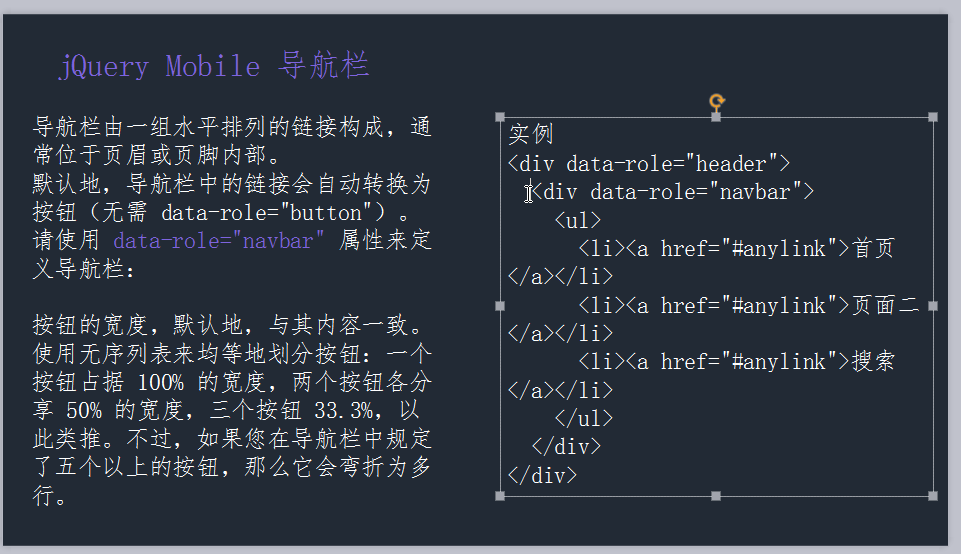
inline默认，均位于行内

data-position=”fixed” 使页眉页脚留在顶部和底部

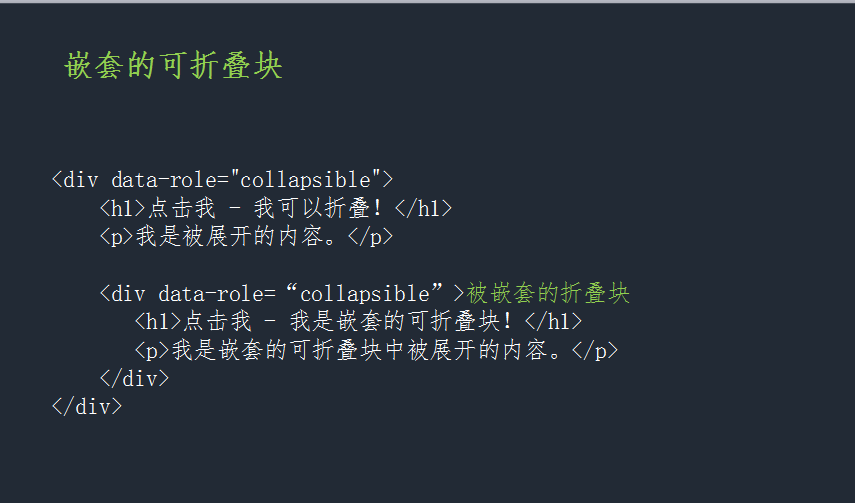
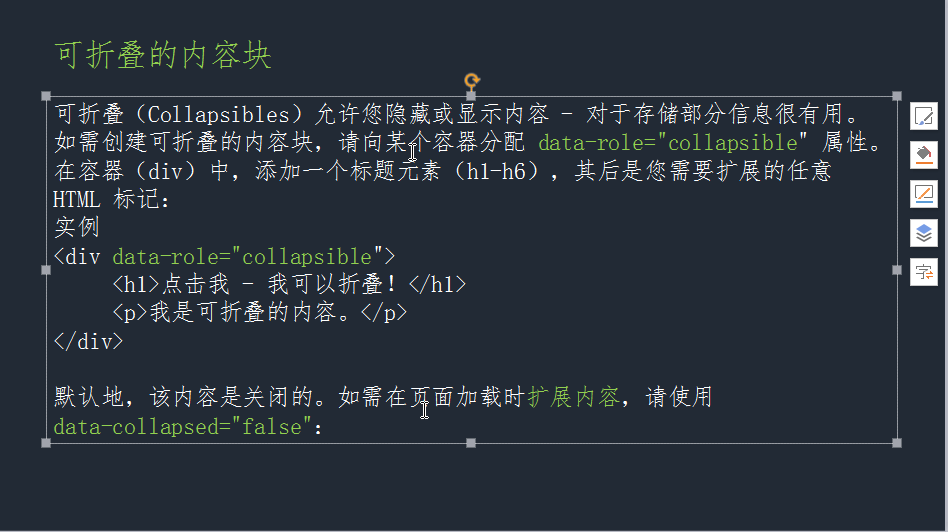
data-fullscreen=”” 有点特殊，值为 true|false

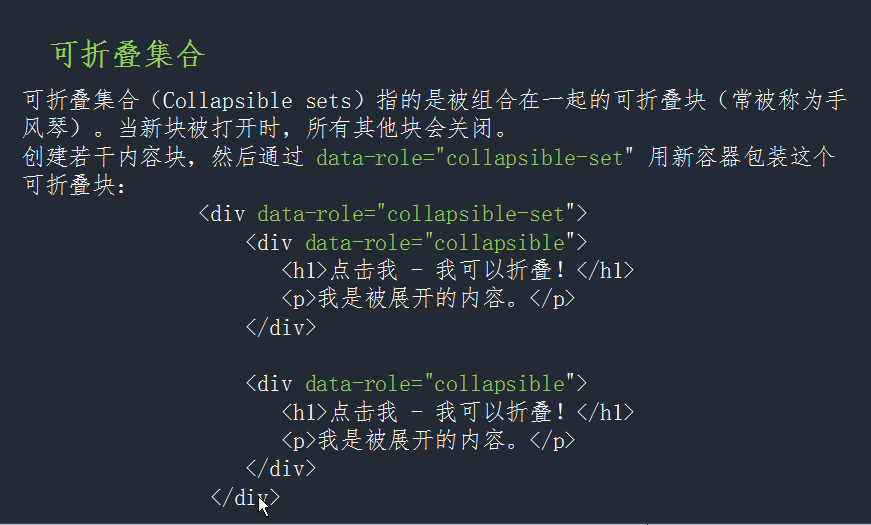
写在data-position=”fixed”后面



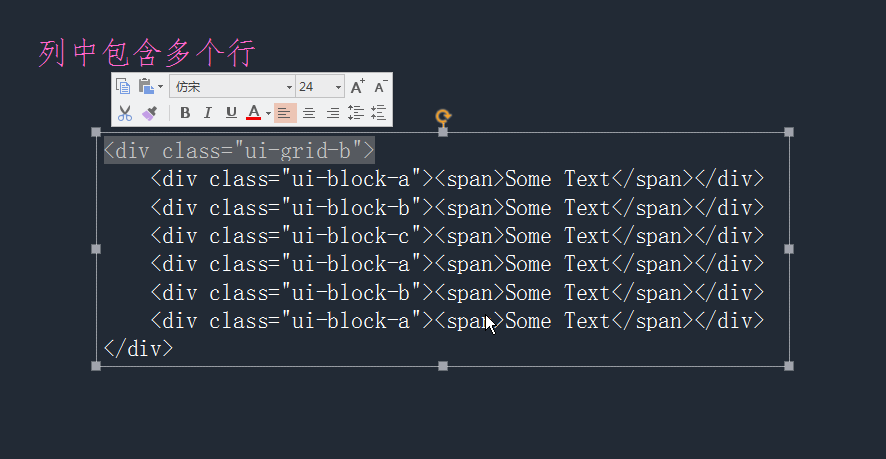




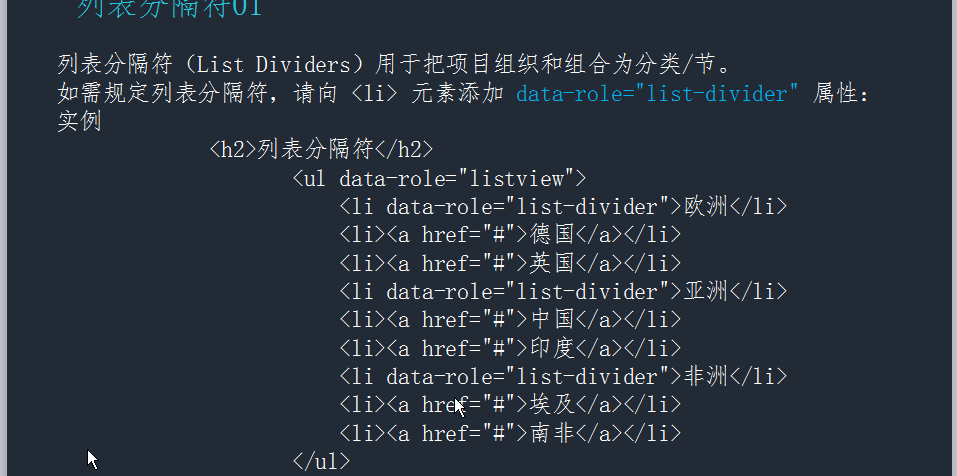
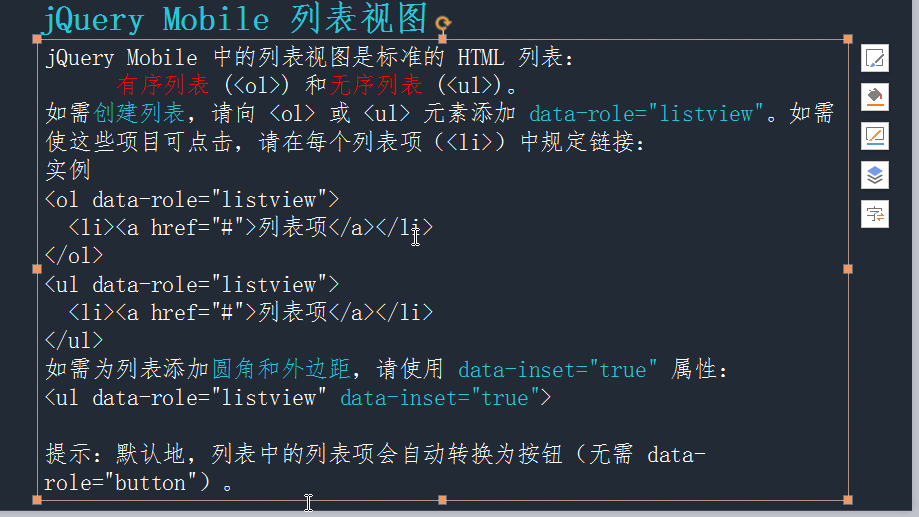


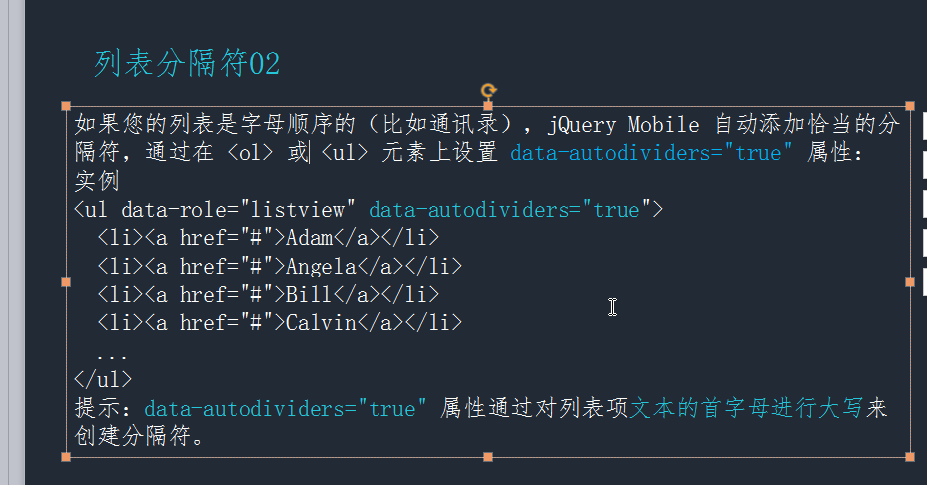


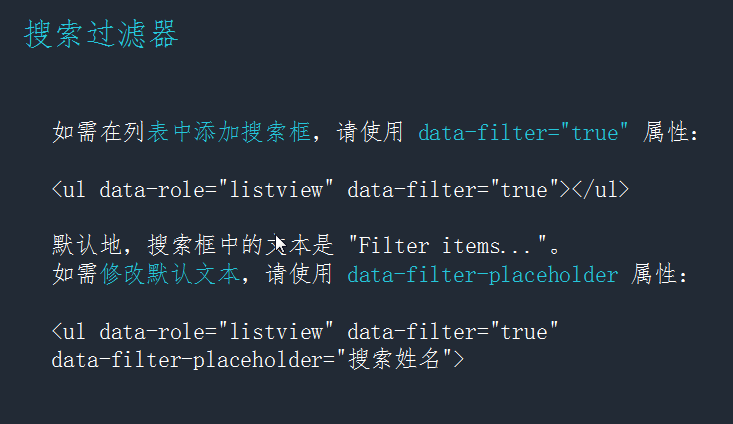


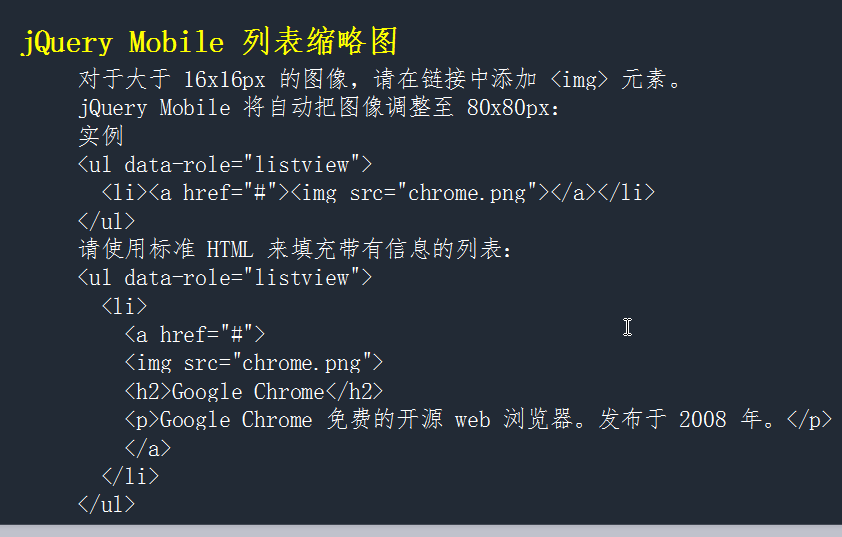


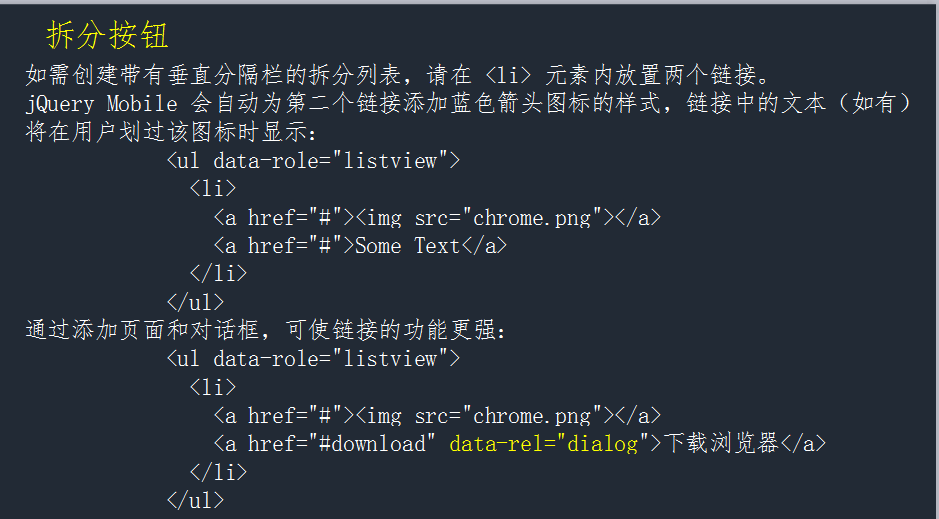


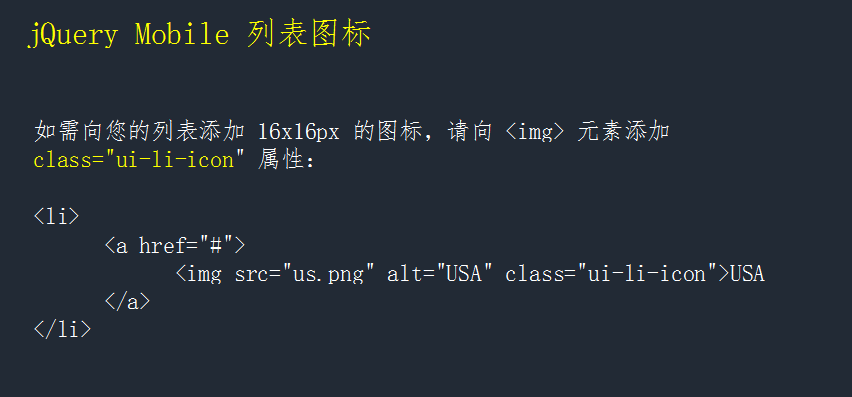






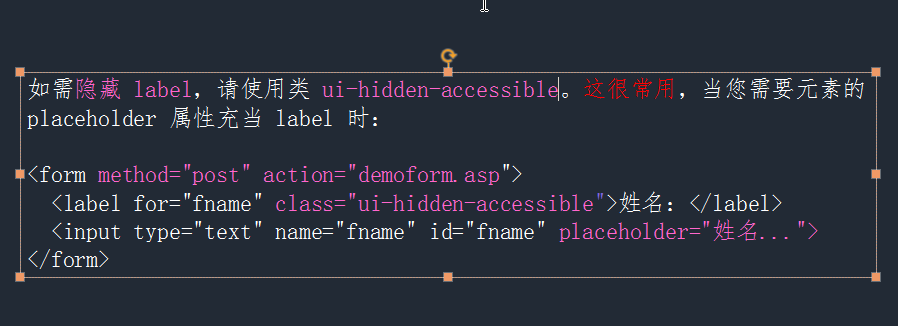


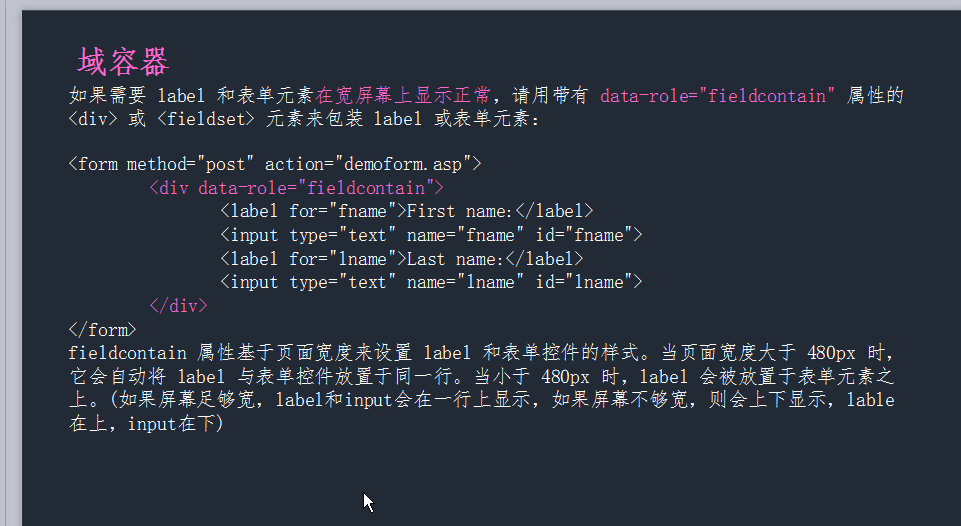










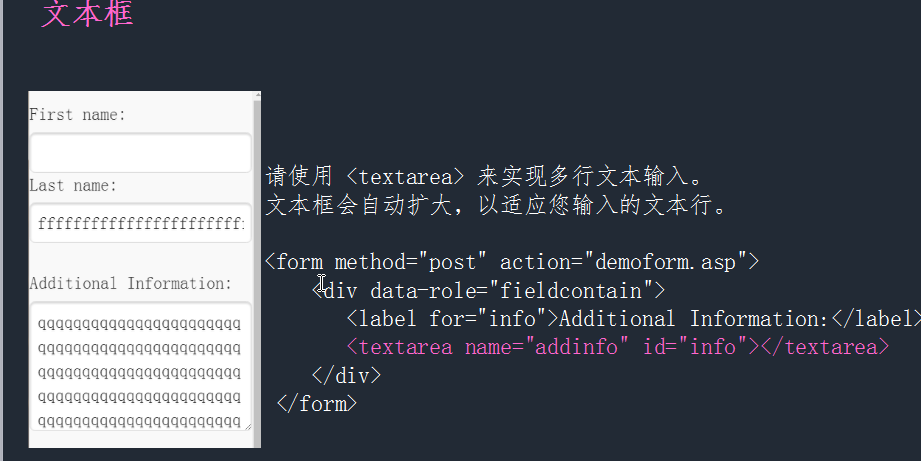


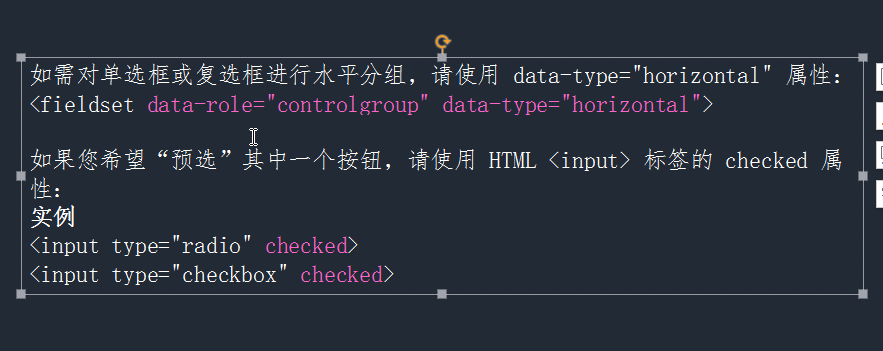
## 表单





文本框：











## 表单，滑块：

<div data-role="main">

音量：

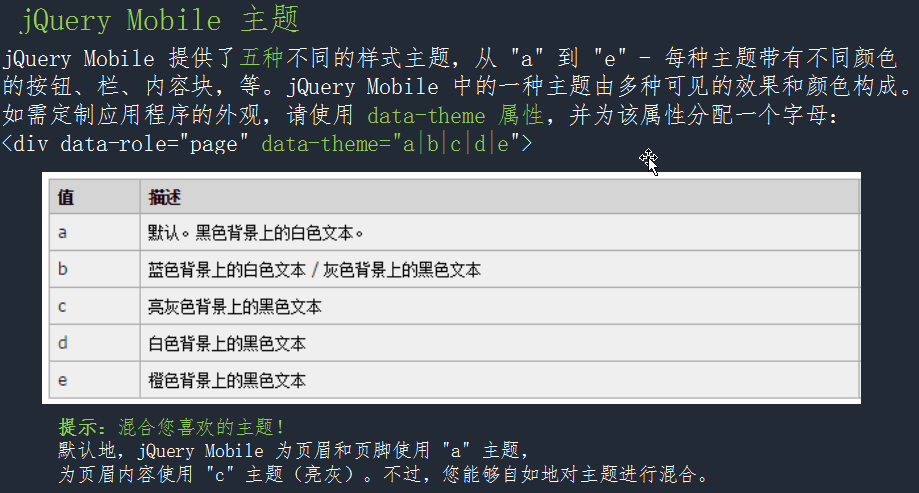
<label for=""></label>

<input type="range" name="vol" id="vol" value="10" min="0" max="100">

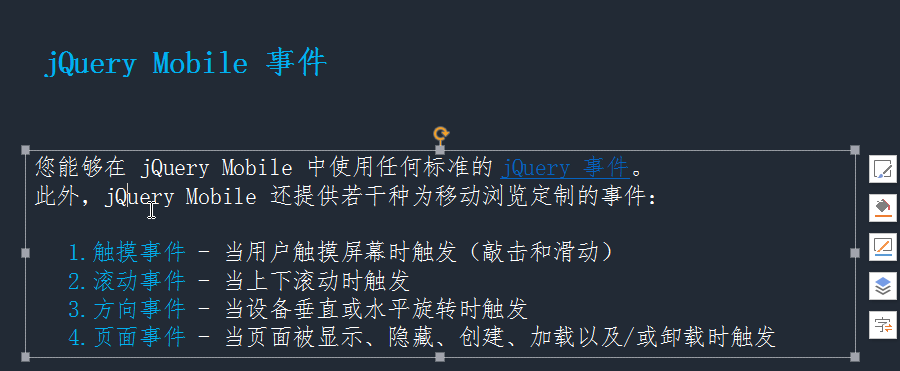
</div>

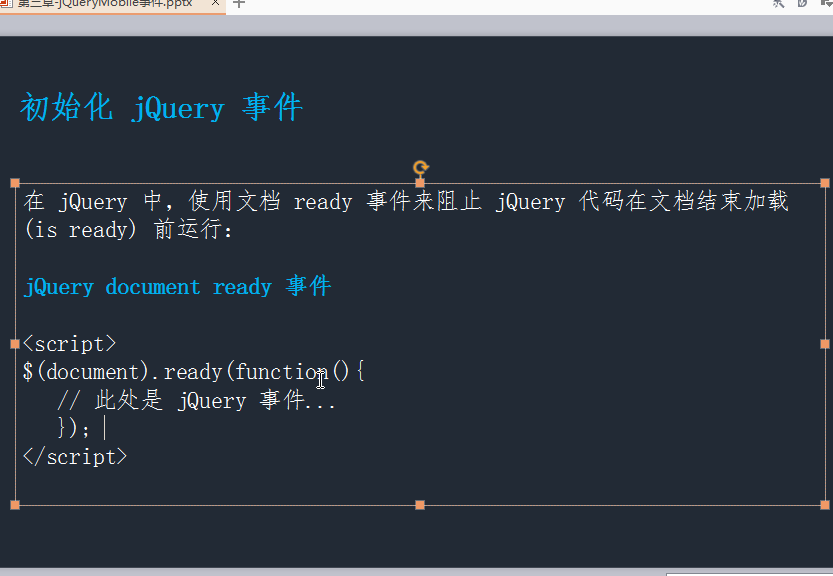
突出显示轨道(在后面加)：data-highlight="true"

切换开关：



## 事件



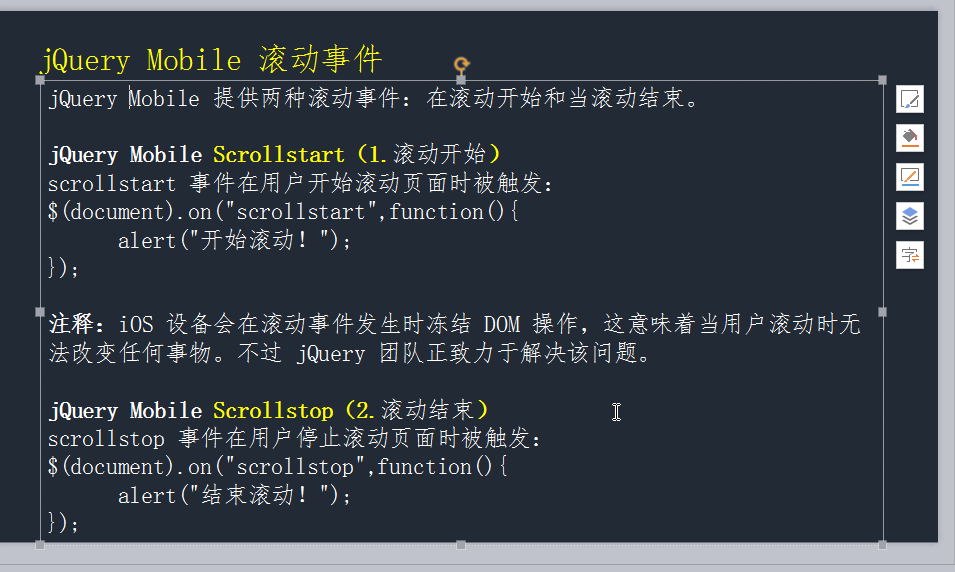


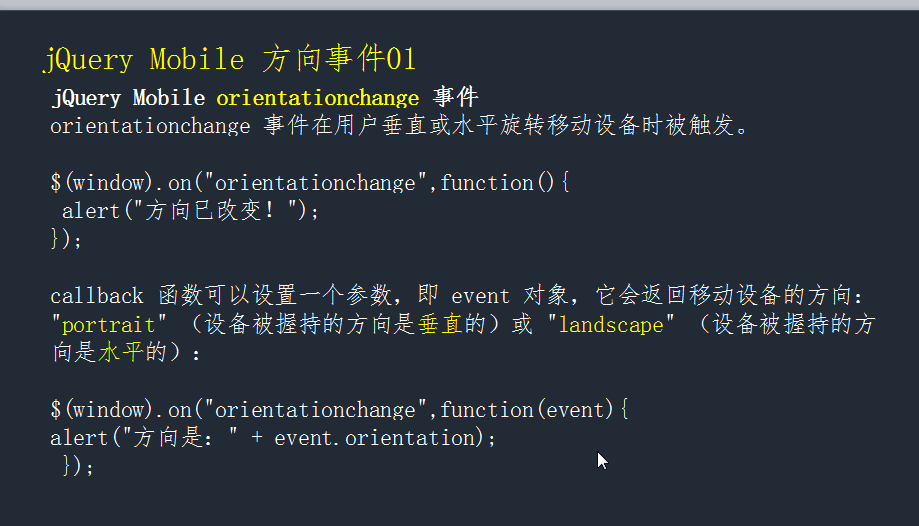


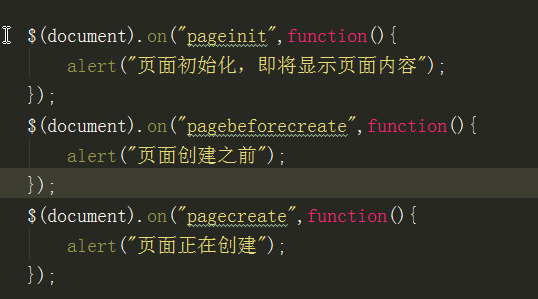
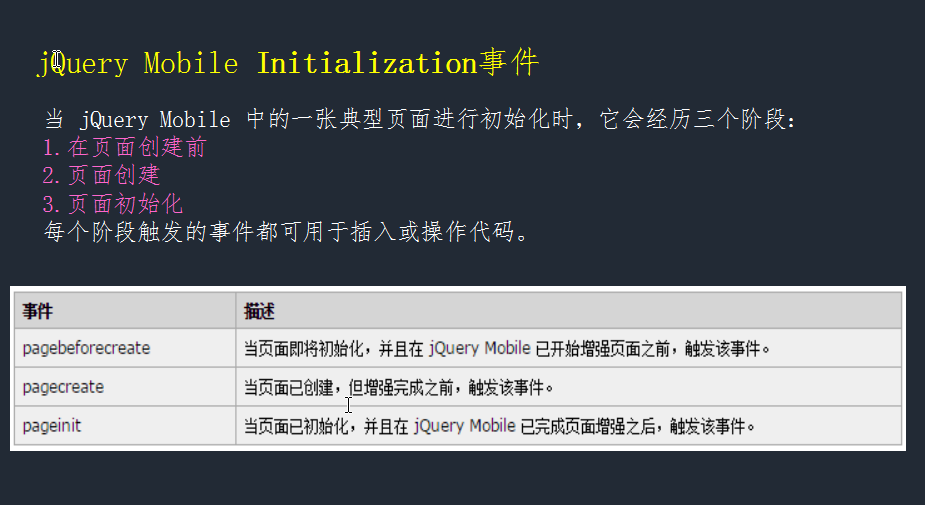
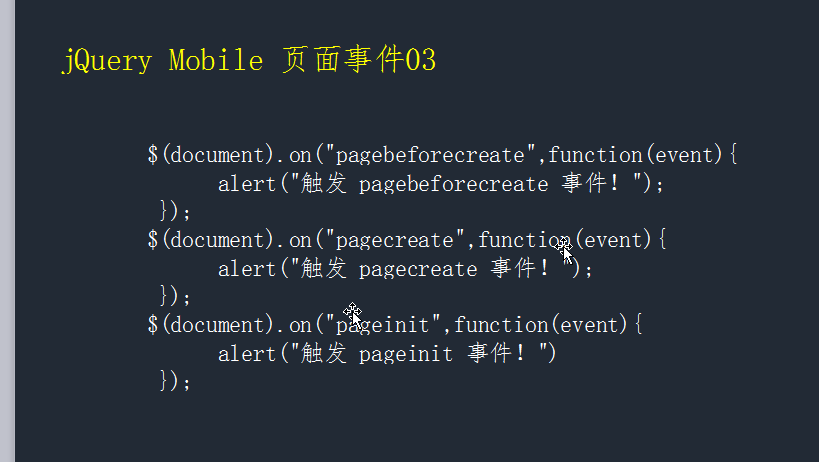


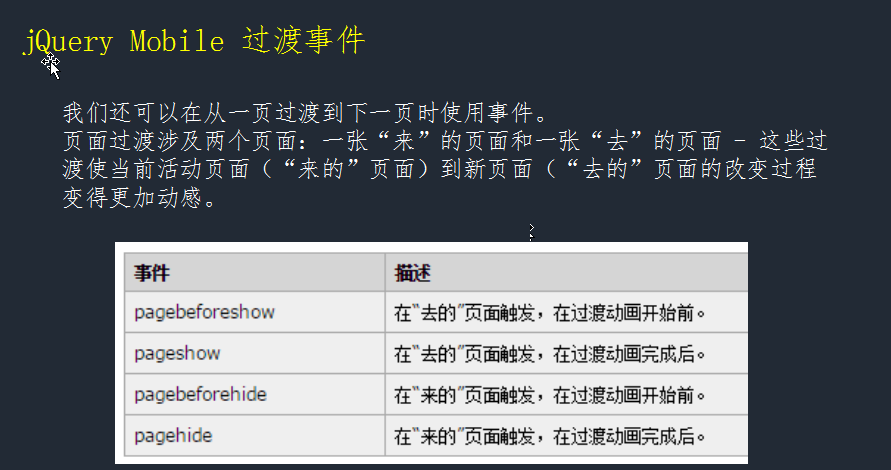


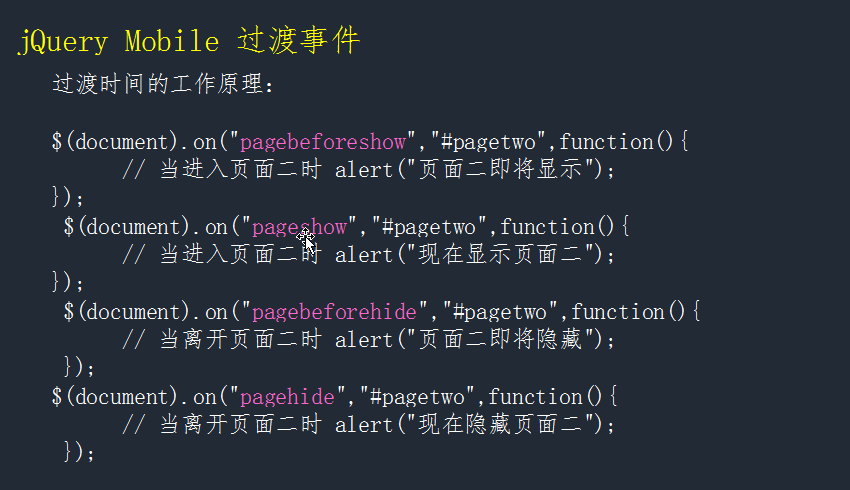
滚动



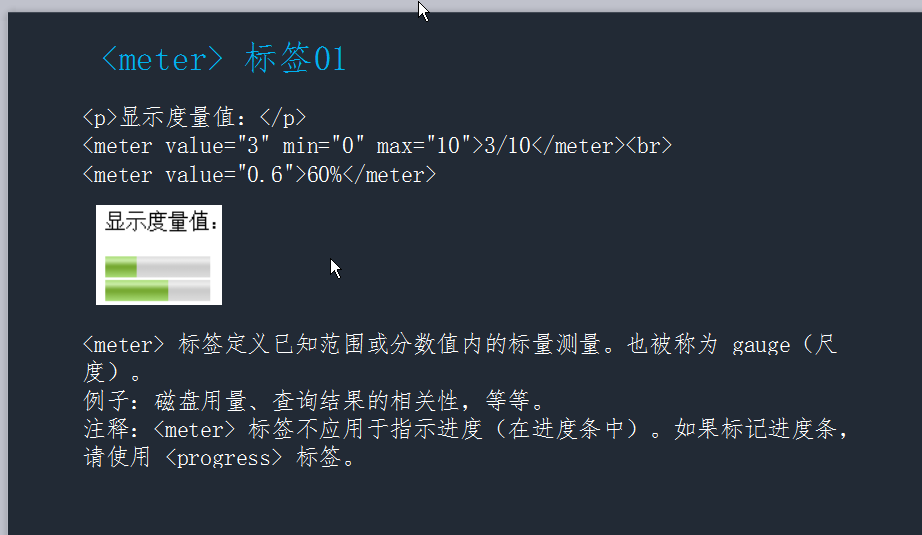


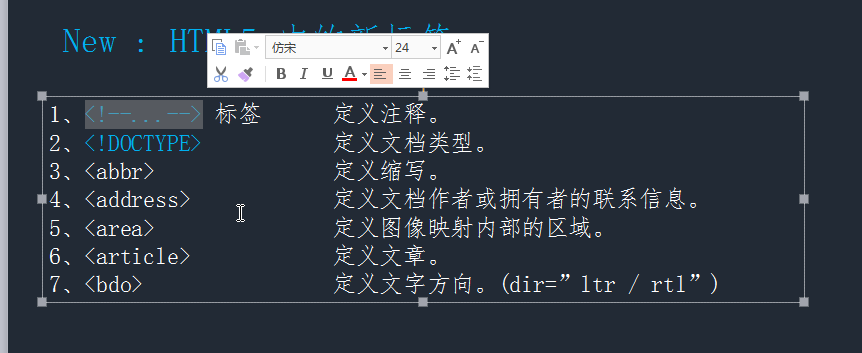


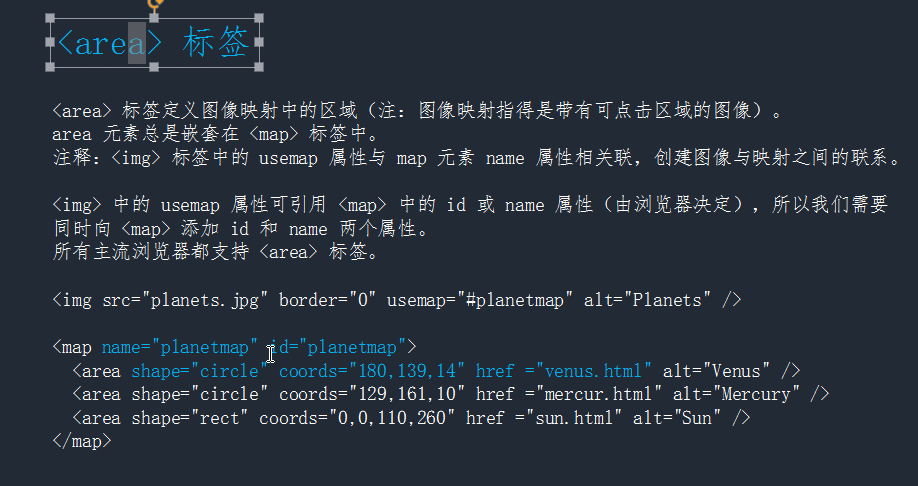




# html5





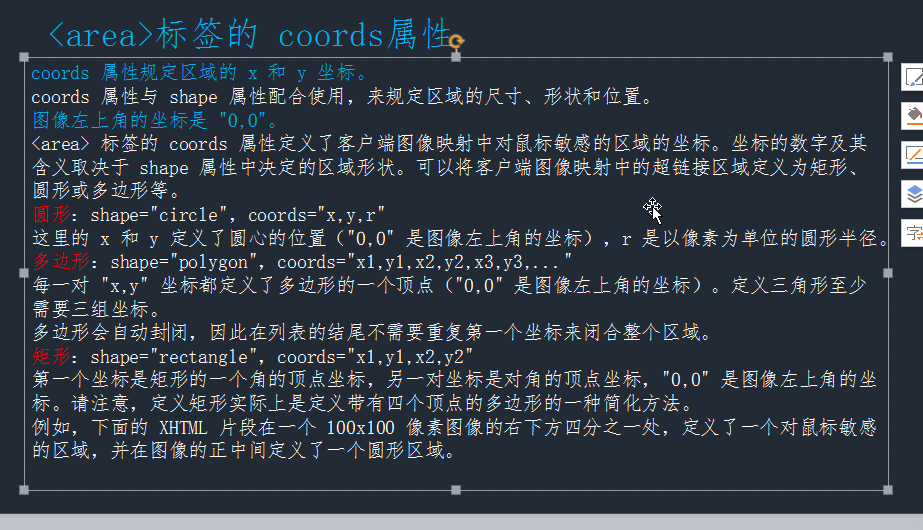


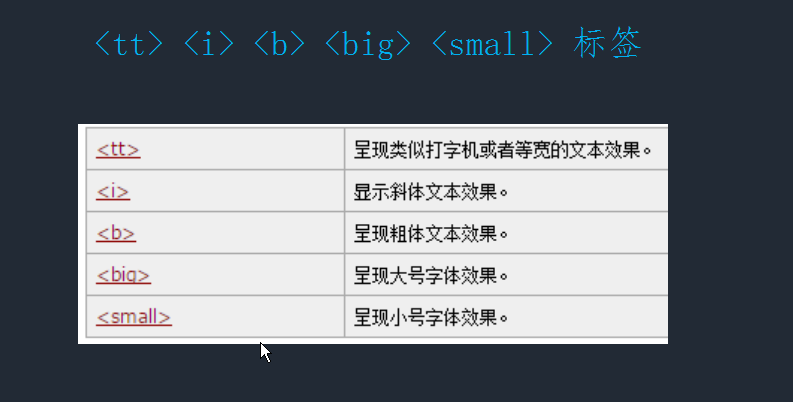
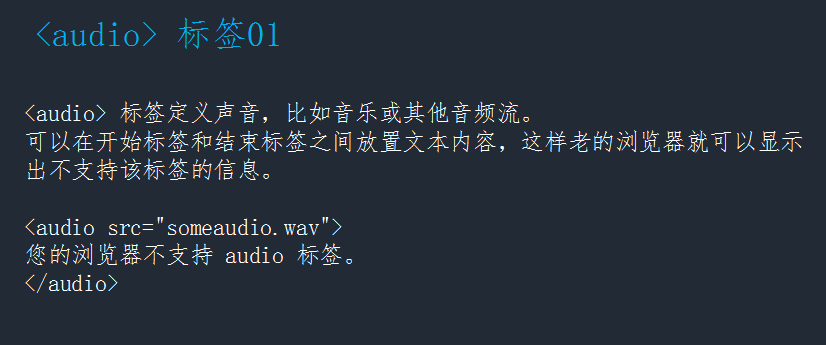
shape=”circle” 圆形区域/可以为多边形/矩形(rect)

coords=”180,139,14” 横坐标，纵坐标和圆的半径

href=”” 点击后跳转的路径

### coords属性：





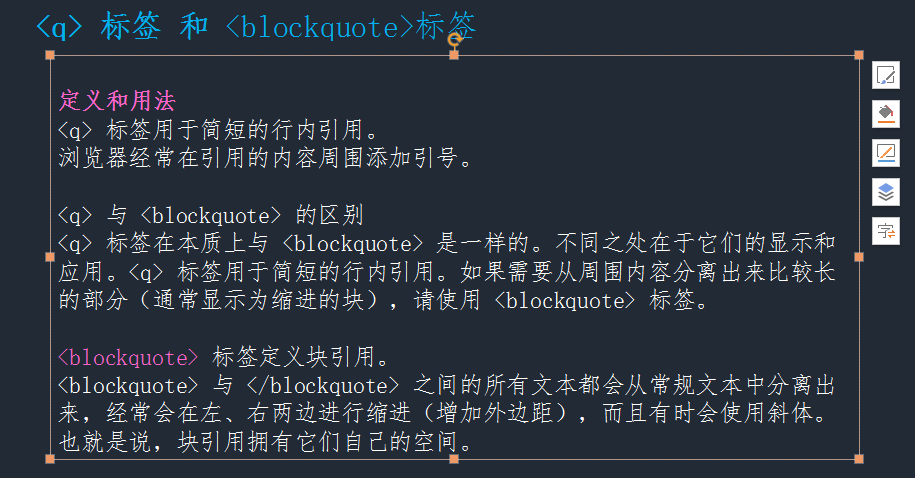
small 小号

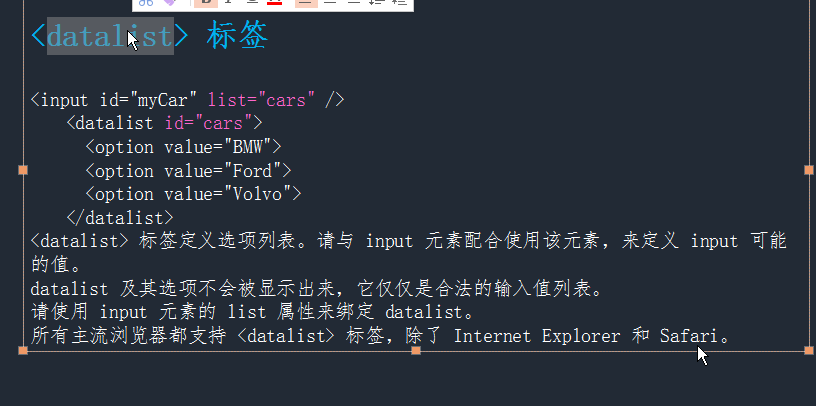
normal 正常

big 大号

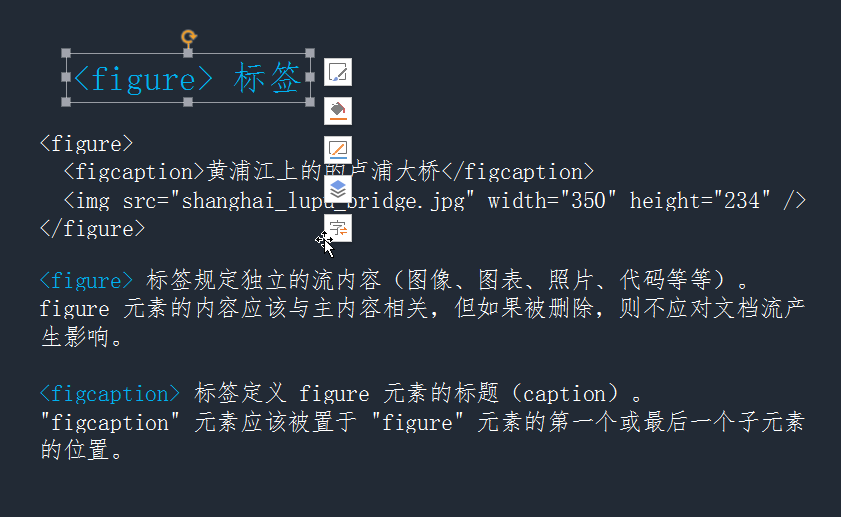


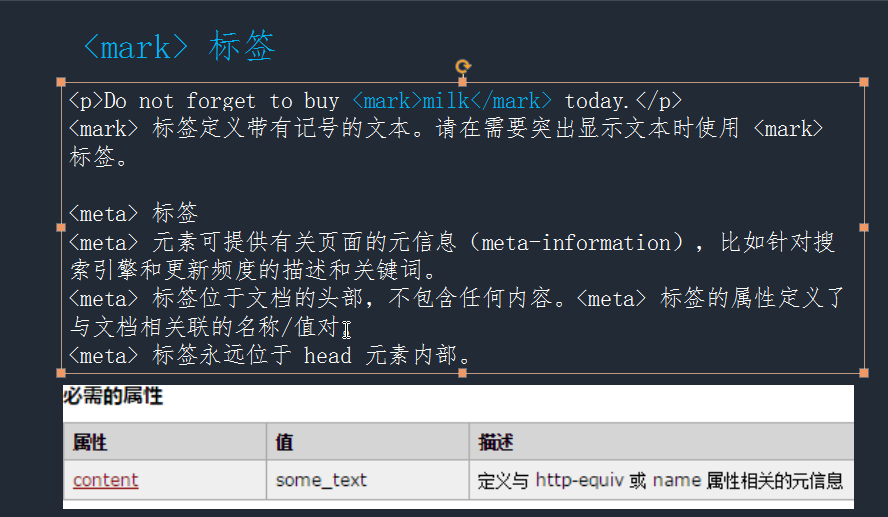


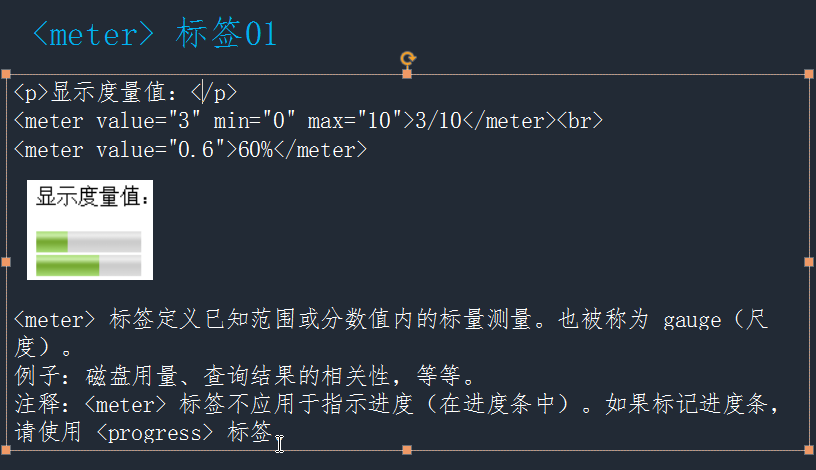


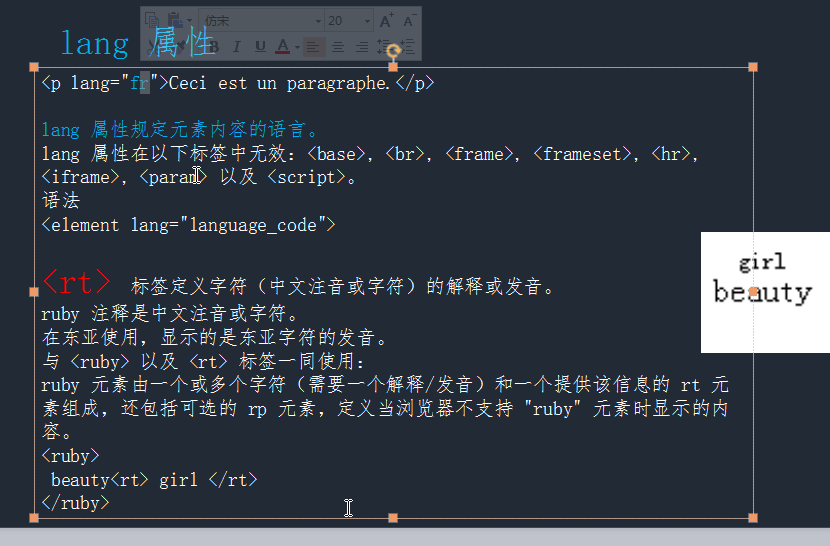








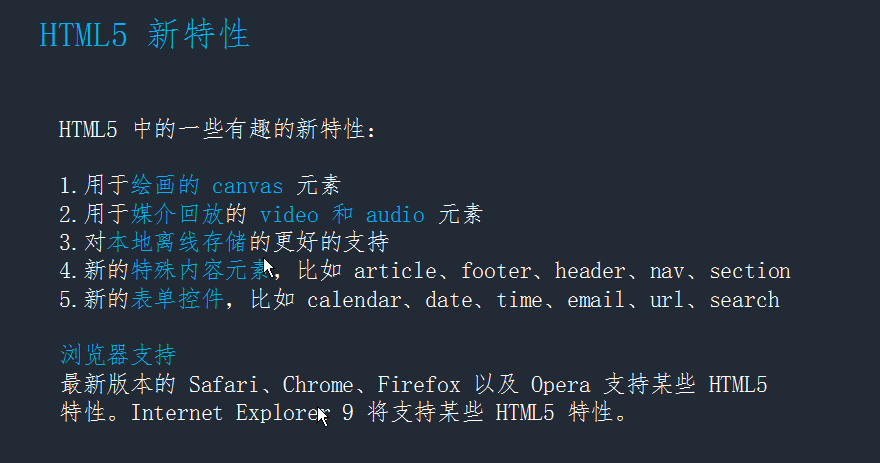
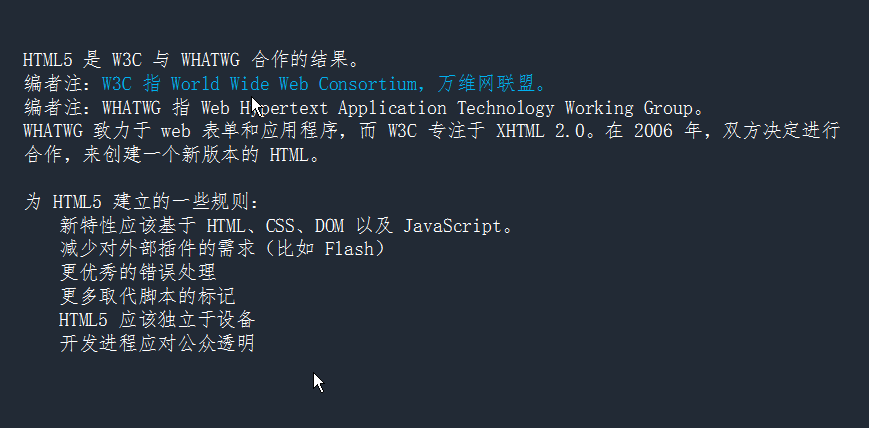




# 字体自适应:

vw

vh

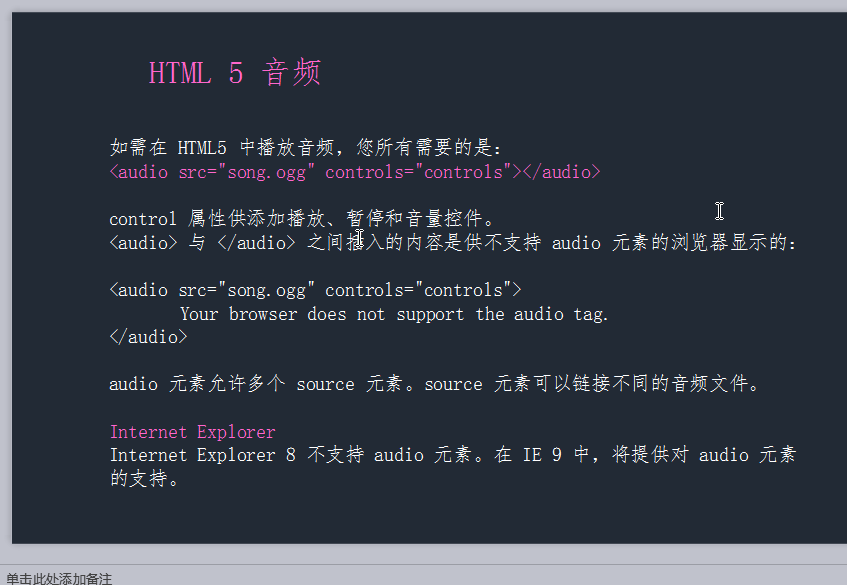


video标签：

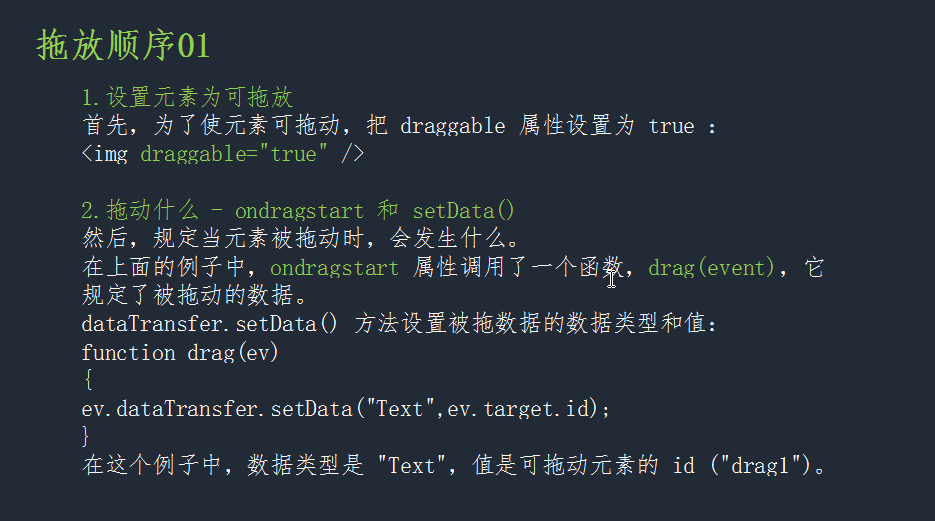
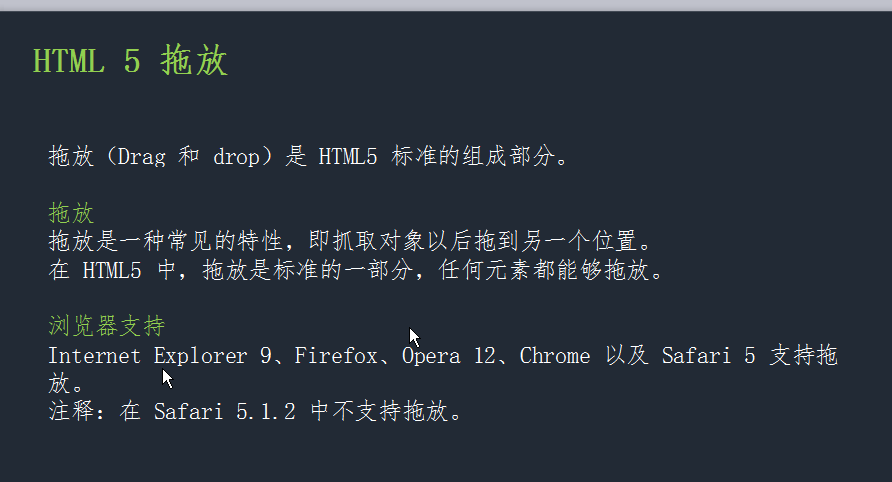
autoplay 属性：自动播放

source标签(可做为video的属性)：引入不同格式的同一文件





拖放：



1 设置为可拖放

draggable="true"

2 拖动什么

function drag(e){

e.dataTransfer.setData("string", e.target.id)

}//数据类型是string，可改变；可拖动的元素是 id

3 放到何处

调用 preventDefault() 来避免浏览器对数据的默认处理（drop 事件的默认行为是以链接形式打开）

ondragover=”allowDrop(event)” --- function allowDrop(event) = {event.preventDefault(); }

(重点是方法，函数名只是一个调用)

即：该方法阻止 对元素的默认处理方式，规定在何处放置被拖动的数据。

4 进行放置

插入节点



.appendChild

