

# HTML5语法变化

DOCTYPE及字符编码

大小写都可以

具有布尔值属性

部分标签可以省略

# 语义化标签

- section

w3c规范规定section是用来划分网页的,但是不能用它来布局,表示页面中的一个内容区块,比如章节,页眉,页脚或页面的其他部分.可以和h1,h2...等其他标签结合起来一起使用,表示文档结构

# 语义化标签

- article

定义可以独立于内容其余部分的完整独立内容块,article元素就是专门为摘要设计的,比如一篇文章

# 语义化标签

- **aside**

表示article标签内容之外的,与article标签内容相关的辅助信息,aside元素应该被用于无关内容。

- 如果你有你认为应该与主内容分开的内容，那么aside元素是你应该正 确考虑使用的元素。
- 询问你自己aside元素中的内容是否可以被独立开来而不会影响文档或 者 section中主内容的含义。
- 可以用在主要内容相关的引用,侧边栏,广告,nav元素组等
- 简单来说 :aside的内容如果被删除，剩下的内容仍然很合理

# 语义化标签

- **hgroup**

对整个页面/页面中的一个内容区块的标题进行组合

- **header**

一个内容区块或整个页面的标题

- **footer**

整个页面或页面区块的尾部

- **nav**

页面中导航链接的部分

# 语义化标签

- figure

- 表示一段独立的流内容,一般表示文档主体流内容中的一个独立单元
- figure元素经常用于图片

- figcaption

- 代表一个图例的说明
- 代表了figure元素的一个标题或者说是其相关解释.
- 在使用figcaption时,它最好是figure块的第一个或者最后一个元素

# 语义化标签

- **mark**

高亮显示

- **progress**

进度条

- **time**

`<time>` `</time>` 用来表现时间或日期

- `<p>` 我们在每天早上 `<time>9:00</time>` 开始营业。

`</p>` 参数

- `<p>` 我在 `<time datetime= "2016-02-14T20:00Z" pupdate>` 这一天`</time>` 发布了一篇文章。 `</p>`

- **wbr**

`<p>`一行放不下则在这换行`<wbr/>`能放下则一行显示`</p>`

# 语义化标签

- **datalist**

选项列表与 input 元素配合使用，可以设置提示信息

```
<input type="text" id="" list="shopCars"/>
```

```
<datalist id="shopCars"> <!--为表单设置可选值-->
```

```
<option value="aa"></option>
```

```
<option value="ab"></option>
```

```
<option value="c"></option>
```

```
</datalist>
```



# 语义化标签

- details

用于描述文档或文档某个部分的细节

summary标签和details一起使用,表示标题,用户点击标题时会得到细节信息

```
<details>
```

```
<summary>111</summary>
```

```
<ul>
```

```
<li>111</li> <li>222</li> <li>333</li>
```

```
</ul>
```

```
</details>
```

# 语义化标签

- output

不同类型的输出,比如脚本的输出

```
<form oninput="x.value=
  parseInt(a.value)+parseInt(b.value)">
<input type="range" id="a" max="100" min="0"
  value="50"/> +
<input type="number" id="b" value="50"/>
<output name="x" for="a b">150</output>
</form>
```

# 新增标签的兼容问题

- HTML5语义化标签在IE6-8下，对于不支持的标签不会有任何的样式,也默认的当成行内元素来出来,所以在样式表里要对这些标签定义一下 它默认的display
- 通过引入如下js来解决ie9以下新增标签的兼容问题
- `<!--[if lt IE 9]> <script type="text/javascript" src="html5.min.js"> </script> <![endif]-->`

# Forms

- 新增input元素的种类
  - search : 搜索输入框
  - tel : 电话号码输入框
  - url : 输入url地址
  - email : 邮件输入框
  - number : 数字输入框
  - rang : 特定范围内的数值选择器(通过拖动滚动条改变一定范围内的数字)

# Forms

- 新增input元素的种类
  - color : 颜色选取器 只在 Opera 和 Blackberry 浏览器
  - datetime : 显示完整日期和时间 UTC标准时间
  - datetime-local : 显示完整日期和时间
  - time : 显示时间
  - month : 显示月
  - week : 显示周

# Forms

- 表单新特性

- placeholder
  - 输入框占位符，常用作输入提示，在光标聚焦时，占位符自动消失
- autocomplete：是否保存用户输入值
  - 默认为on,关闭提示选择off
- autofocus：自动聚焦
- required：此项必填，不能为空
- Pattern：正则验证 `pattern="\d{1,5}"`
- form 属性 只要加上 form 属性，表单元素可以放到页面的任意位置。
- formnovalidate 和 novalidate
  - 它俩都表示不需要验证表单,直接提交, : novalidate 用于 form 标签；
  - formnovalidate 用于 submit类型的提交按钮。

# Forms

- 表单验证

- validity对象，通过下面的valid可以查看验证是否通过
  - oText.addEventListener("invalid",fn1,false);
  - valid : 验证不通过时返回false
  - valueMissing : 输入值为空时
  - typeMismatch : 控件值与预期类型不匹配
  - patternMismatch : 输入值不满足pattern正则
  - customError 不符合自定义验证
    - » setCustomValidity(); 自定义验证

# 全局属性

- data-yourvalue
  - 自定义属性
    - data-name : dataset.name
    - data-name-first : dataset.nameFirst
- Hidden
  - 加上这个属性,元素则是不可见状态
- spellcheck
  - 对你输入的内容纠错
- tabindex
  - 按下tab键可以根据设置的顺序进行跳转



# 全局属性

- contenteditable
  - `<p contenteditable="true">请您留言</p>`
  - 表示这块内容是可以编辑的 去掉则不可以修改
- designMode
  - `<script>`
  - `window.document.designMode = "on "`
  - `</script>`