HTML CSS JS

前端系列培训课程之一

About me

舒克

淘宝旅行前端工程师 @葫芦居士 www.gourdboy.com

这不是速成班

http://wiki.ued.taobao.net/doku.php?id=ued.bj:f2e:f2eks





















只会用工具的前端工程师时代已经过去

HTML: 结构 & 内容

CSS:样式

JS: 结构 & 内容 & 样式 & 动画 & 交互

HTML: 结构 & 内容 & 交互

CSS:样式

JS: 结构 & 内容 & 样式 & 动画 & 交互

HTML: 结构 & 内容 & 交互

CSS: 样式&动画

JS: 结构 & 内容 & 样式 & 动画 & 交互

HTML

Hypertext Markup Language 超文本标记语言

HTML

- 文档类型声明 doctype
- 元素 element
- 属性 attribute

文档类型声明

html4.0 & xhtml1.1

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

Strict, Transitional, Frameset

html4.0 ~ html5.0

<!DOCTYPE html>

向上向下兼容

head

● title 页面标题,必需且唯一

meta

```
<meta charset=" gbk" />
<meta name="description" content="Seo相关的页面介绍文字"/>
<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
<meta name="viewport" content="width=device-width,initial-scale=1,maximum-scale=1">
<meta name="apple-mobile-web-app-capable" content="yes">
```

link

```
k rel="shortcut icon" type="image/x-icon" href="http://www.taobao.com/favicon.ico" />
k rel="stylesheet" href="http://a.tbcdn.cn/??p/global/1.0/global-min.css" />
```

元素

块级元素 (block)

div, p, h1~h6, ul, li, dt, dd, dl, form, section, article, nav, aside...

特点:在文档流中默认占据整行的位置

行内元素 (inline)

span, em, strong, label, a, img, input, button, select, var, del ...

特点:在文档流中默认一行可以展示多个行内元素

• 嵌套规则

块级元素可以嵌套所有行内元素,除a以外的行内元素不能嵌套块级元素

错误的写法: <div> 文本内容1 </div>

合理使用语意化的标签

增强页面的可访问性 搜索引擎优化(SEO)

Headling

h1~h6的合理应用对于SEO至关重要















detail页面的hl应该为文章或者宝贝名称

常用语意化元素

- header, footer
- section, article
- nav , aside , figure
- ol, ul, dl
- table , tr , th ,td

通用属性

- id
- class
- 不再使用的属性

align, bgcolor, background, color, bgsound, border...

表单元素

- form
- label
- input , textarea , select
- button

表单元素对交互影响:通用属性

label标签的应用

用法1:<input id="J_MyChk" type="checkbox" value="" /><label for="J_MyChk">点击我也可以让左边选中</label>

用法2: <label> <input type="checkbox" value="" />点击我也可以让左边选中 </label>

- tabindex
- autocomplete
- disabled
- button的type属性

错误: <button>我是个按钮</button>

正确: <button type="button">我是个按钮</button>

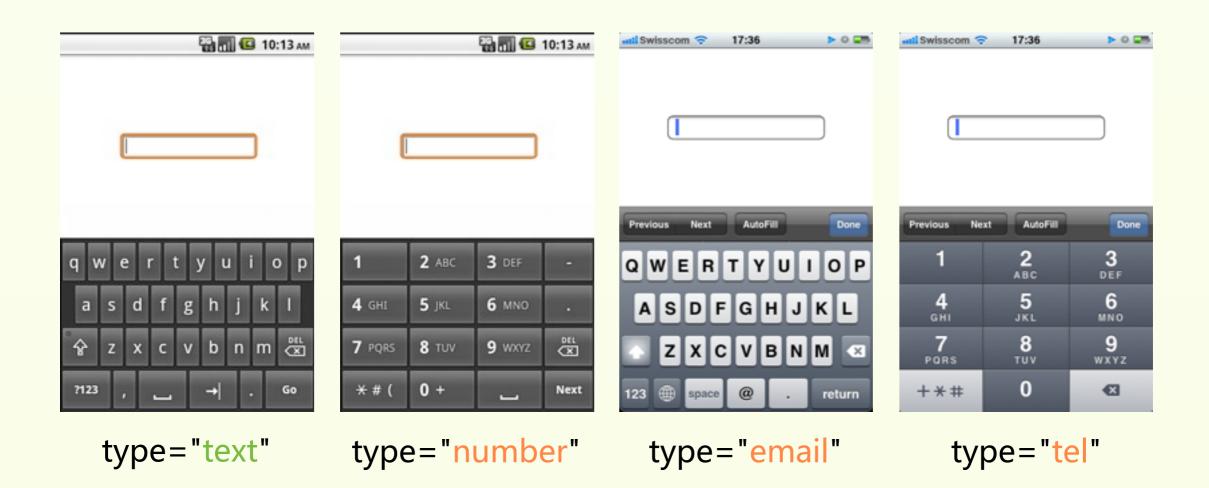
表单元素对交互影响:HTML5增强表单

input元素type属性的合理应用

以前:text, password

现在: number, date, tel, url, search, range, color ...

不同type值的input在手机端的展示



增强性 兼容性: http://www.quirksmode.org/html5/inputs.html

max,min,step

```
<input type="number" min="-10" max="1000" step="5" />
```

required

必填项

autofocus

默认获得输入焦点

placeholder

占位符

pattern

对输入控件进行正则校验

required + pattern

不再依赖JS对表单进行输入格式和必填项校验

firefox + chrome全面支持

推荐在小二工具应用中使用

增强的属性



x-webkit-speech

IE条件注释

```
<!--[if It IE 7]><html class="no-js ie ie6 Ite9 Ite8 Ite7"> <![endif]-->
<!--[if IE 7]><html class="no-js ie ie7 Ite9 Ite8 Ite7"> <![endif]-->
<!--[if IE 8]><html class="no-js ie ie8 Ite9 Ite8"> <![endif]-->
<!--[if IE 9]><html class="no-js ie ie9 Ite9"> <![endif]-->
<!--[if gt IE 9]><html class="no-js"><![endif]-->
<!--[if !IE]><!--><html><!--<![endif]-->
```

CSS Cascading Style Sheet 层叠样式表

CSS发展史

CSS1 1996.12成为推荐标准(字体,颜色、空白边)

CSS2 1998.5成为推荐标准 (浮动,定位,选择器)

CSS2.1 2002年修订(伪类,属性选择器)

CSS3 未正式发布但已普遍开始应用

针对不同设备的CSS代码

```
media属性
k media= "screen" />
样式表内的写法
@media print
{
    #header{border:1px solid #000}
}
```

可以针对不同的媒介类型书写CSS

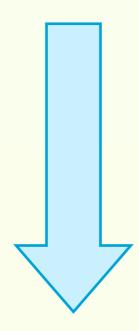
媒介类型	描述
all	用于所有的媒介设备。
aural	用于语音和音频合成器。
braille	用于盲人用点字法触觉回馈设备。
embossed	用于分页的盲人用点字法打印机。
handheld	用于小的手持的设备。
print	用于打印机。
projection	用于方案展示,比如幻灯片。
screen	用于电脑显示器。
tty	用于使用固定密度字母栅格的媒介,比如电传打字机和终端。
tv	用于电视机类型的设备。

MediaQuery 针对设备的不同状态书写CSS

```
/* 正常PC液晶屏样式 */
/* iPad 及以下,以及所有小于(不等于)960宽度的平板电脑 */
@media only screen and (max-width: 959px) {}
/* QiPad 竖版,以及所有小于(不等于)标准960宽度的平板电脑的竖版 */
@media only screen and (min-width: 768px) and (max-width: 959px) {}
/* iPhone 及以下 */
@media only screen and (max-width: 767px) {}
/* QiPhone 横版,包括某些平板电脑的竖版 */
@media only screen and (min-width: 480px) and (max-width: 767px) {}
/* Qiphone4 竖版 */
@media only screen and (max-width: 479px) {}
```

层叠的意义

- 浏览器缺省设置
- 外部样式表
- 内部样式表
- 浏览者自定义样式



CSS语法

● 选择器\属性\值

```
selector {
    property : value
}
```

选择器 Css Selector

- 元素选择器 h1
- 类选择器 .classname
- id选择器 #id
- 全局选择器 *
- 继承选择器 div p
- 选择器分组 h1,h2
- 伪类选择器:hover
- CSS 属性选择器 input[type= "button"] ie6

选择器的优先级

选择器	A(最高級style)	B(id的数目)	C(class的数目)	D(标签的数目)	优先级(组合值)
*	0	0	0	0	0
P	0	0	0	1	1
ра	0	0	0	2	2
p a. color	0	0	1	2	12
p .color .text	0	0	2	1	21
p .color div #news	0	i	1	2	112
style=""	1	0	0	0	1000
style="""	1	0	0	0	1000

表格中A、B、C、D分别代表四个等级A为最高级,D为最底级也可以理解为数学中的千位,百位,十位,个位,D就代表个位

伪类选择器

- a:link, a:hover, a:active, a:visited
- :hover 和 :focus 在表单输入控件中使用
- ::selection 对划选的内容进行样式设置
- :first-child 选择第一个元素

CSS属性

• 属性继承

Text-related properties that are inherited

文本样式相关的属性会继承

List-related properties that are inherited

列表样式相关属性会继承

The color property is inherited

颜色属性会继承

常用会继承CSS属性

border-collapse, border-spacing, caption-side, Color, Cursor, direction, empty-cells, font-family, font-size, font-style, font-variant, font-weight, font, letter-spacing, line-height, list-style-image, list-style-type, list-style, orphans, pitch-range, pitch, quotes, text-align, text-indent, text-decoration, visibility, white-space, word-spacing

CSS布局

- 表格布局
- float布局
- 负边距布局

http://www.cnblogs.com/mofish/archive/2011/01/18/1938543.html

• 圣杯布局

参考: http://www.alistapart.com/articles/holygrail

- 绝对定位布局
- inline-block布局法

YUI CssGrid 1.定宽 2.流式布局

why?

阻止边距折叠

可以包含内部元素的浮动

阻止元素被浮动覆盖

• 触发

float的值不为none。

overflow的值不为visible。

display的值为table-cell, table-caption, inline-block中的任何一个。

position的值不为relative和static。

BFC相关资料: http://www.yuiblog.com/blog/2010/05/19/css-101-block-formatting-contexts

一些容易让人困惑的 CSS属性

position:

- static 默认值
- relative 相对定位
- absolute 绝对定位
- fixed 可视区域定位/屏幕定位

overflow:

- visible 始终可见 (默认值)
- hidden 溢出隐藏 (子元素有绝对定位时慎用)
- auto 自动,溢出时出现滚动条
- scroll 始终出现滚动条

vertical-align:

- middle 垂直居中对齐
- top 顶部对齐
- bottom 底部对齐

display: & visibility:

- display:none
- visibility:hidden

渐进增强的CSS3

圆角 border-radius

ff4+ chrome5+ ie9+

width: 100px; height: 100px; border-radius:5px width: I 00px; height: I 00px; border-radius: 50px;

ie6-ie8

width: I 00px; height: I 00px; border-radius:5px

渐变 CSS Gradient

• 学习:

http://www.slideshare.net/LeaVerou/mastering-css3-gradients

工具:

http://www.colorzilla.com/gradient-editor/

过渡动画 transition

```
指定元素的某个属性发生变化时产生过渡效果
.has-trans {
  -webkit-transition: all 0.3s ease-out; /* Saf3.2+, Chrome */
  -moz-transition: all 0.3s ease-out; /* FF4+ */
  -ms-transition: all 0.3s ease-out; /* IE10 */
  -o-transition: all 0.3s ease-out; /* Opera 10.5+ */
  transition: all 0.3s ease-out;
```

前端基础培训课程 - HTML CSS JS 47

扩展阅读: http://www.w3cplus.com/content/css3-transition

动画 animation

```
@keyframes anim-flash {
     0%, 50%, 100% {opacity: 1;}
     25%, 75% {opacity: 0;}
}
.anim-flash {
     animation-name: anim-flash;
}
```

RGBA

• 增加了透明的支持

border:10px solid rgba(0,0,0,0.5);

background:rgba(255,0,0,0.3);

CSS属性值参考手册

https://developer.mozilla.org/en/CSS_Reference

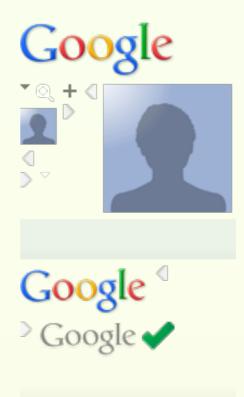
CSS sprites

优点

- 1.减少HTTP请求,降低服务器负担
- 2.文件体积更小 1+1 < 2

缺点

- 1.开发/维护成本高
- 2.扩展性差



关于图片-基础知识

- 图形 VS 照片
- 真彩色图像 VS 调色板图像
- 透明 和 alpha通道(RGBA)
- 隔行扫描

关于图片-格式

- gif
- png
- jpg

JavaScript

参看 地极PPT

Q&A

参考资料

- https://developer.mozilla.org/cn/HTML
- https://developer.mozilla.org/en/ CSS Reference
- 《高性能网站建设进阶指南》 Steve Souders