# 最佳实践

## IDE

HTML/JS这项脚本语言由于无需编译，语法检查很弱，若没有一个开发环境支撑，你很可能因为少一个分号或括号导致各种奇怪的BUG出现。推荐使用类似INTELLJ/WEBSTORM等工具，智能提示许多基本的语法检查。

## 先实现业务，在考虑复用，再考虑UI

一开始不要纠结在表单的复用，展示UI。

用基本的HTML标签构建表单，提交HTTP。例如：

我们要实现一个最基本的功能是关于两个表单的删减：

## 复用，组件式开发思路

为了避免重复制造车轮，考虑用相同熟悉HTML开发以后再考虑的工程。

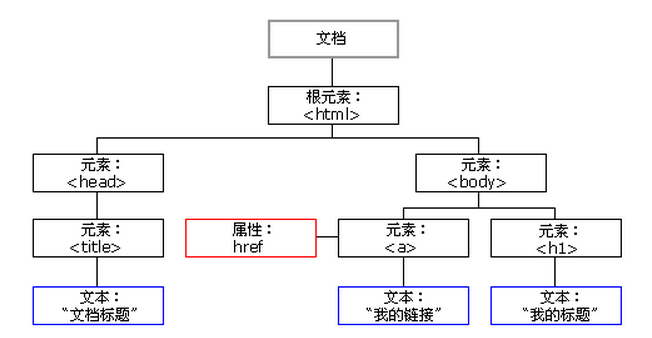
RESTFUL，GULP

你看下做得好的网站，他们的脚本样式文件都是压缩合并过的，根据自己的需要还可以做md5之类的操作。以前没有前端构建工具的时候，估计还得用像shell来做或者手动来做，但我没经历过那个阶段所以也是猜测啦。grunt出来后，可以说就是为了减少前端手动做这种工作的流程化工具，个人觉得并不是工具上带来了什么新鲜的东西，而是托node的福，前端可以不用去学新的语言就可以去用去写适合自己的前端构建流程，我grunt task一下，打包压缩什么一大堆的事情就帮我做好了，想想都觉得痛快！后来又有人嫌grunt过于冗余繁杂，就有了gulp。

# HTML

## DOM(jQuery元素选择器)

HTML DOM（文档对象模型）可以理解成一种特殊的数据格式(类似JSON、XML一样)，他可以被许多语言（如JS）解析成对象读取。HTML DOM规范了HTML下各级标签的关系:



由父及子依次为： document-> 元素-> 属性-> 文本

|  |  |  |
| --- | --- | --- |
| **选择器** | **实例** | **选取** |
| [\*](http://www.w3school.com.cn/jquery/selector_all.asp) | $("\*") | 所有元素 |
| [#*id*](http://www.w3school.com.cn/jquery/selector_id.asp) | $("#lastname") | id="lastname" 的元素 |
| [.*class*](http://www.w3school.com.cn/jquery/selector_class.asp) | $(".intro") | 所有 class="intro" 的元素 |
| [*element*](http://www.w3school.com.cn/jquery/selector_element.asp) | $("p") | 所有 <p> 元素 |
| .*class*.*class* | $(".intro.demo") | 所有 class="intro" 且 class="demo" 的元素 |

## 交互元素

内容/菜单/状态

<details>

<summary>

## 表单<form>

表单是一个包含表单元素的区域，表单元素是用来收集用户收入的元素。

<form>定义了一个表单元素的区域，

action: 表单提交的地址, 肯定是要提交给某个服务器的,action的地址就是要提交的服务器的地址；

method:以何种方式提交,常见的为get和post. 区别在于一个是明文(在url中显示),一个是暗文；

<form action="url" method="submit\_method">

<input type="radio" name="sex" value="male" checked>male

<input type="radio" name="sex" value="female" >female

</form>

### 输入类型<input>

是用户输入信息类元素，具体type有:

* text/password文本/密码域
* checkbox 复选框
* radio单选框
* file
* submit

HTML5以后支持：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Input type** | **IE** | **Firefox** | **Opera** | **Chrome** | **Safari** |
| **email【正则验证】** | No | 4.0 | 9.0 | 10.0 | No |
| **url【正则验证】** | No | 4.0 | 9.0 | 10.0 | No |
| **number【正则验证】** | No | No | 9.0 | 7.0 | No |
| range | No | No | 9.0 | 4.0 | 4.0 |
| Date pickers | No | No | 9.0 | 10.0 | No |
| search | No | 4.0 | 11.0 | 10.0 | No |
| color | No | No | 11.0 | No | No |

#### 实例

<input type="text" id="input1" size="40" ms-on-input="@check" value="FIN.CDE\_DISCOUNT\_CHANNEL\_MON" />

其中ms-on-input是avalon在执行点击表单元素时使用的；

### 表单元素<select>

下拉列表：

<**form**>  
 <**select id="selectSCPType"**>  
 <**option** >oracleLDR</**option**>  
 <**option** >tdFLD</**option**>  
 <**option** >tdTptload</**option**>  
 <**option** >tdTptexp</**option**>  
 </**select**>

</**form**>

### HTML表单元素

|  |  |  |
| --- | --- | --- |
| datalist | 限定输入的下拉列表，同时支持通过输入来快速检索，适用于5个选项以上的，如地区 |  |
| output | output 元素用于不同类型的输出，比如计算或脚本输出： |  |

参考：

1. <http://www.w3school.com.cn/html/html_forms.asp>

<http://www.w3school.com.cn/html5/tag_input.asp>

2.台湾资源<http://www.cs.nccu.edu.tw/~lien/BCC/HTML/YZIT/HTMLFORM.html>

4.FROM的TARGRT <http://blog.micxp.com/index.php/archives/214/>

#### 分块 DIV/SPAN

不同区域的元素，文本需要不同的处理,这就是div和span的区别,他们能圈取一个区域进行统一管理。

阴影开关；

div、h1 或 p 元素常常被称为块级元素。即“块框”。

span 和 strong 等元素称为“行内元素”，这是因为它们的内容显示在行中，即“行内框”。

面试题目：

FILE API:<http://www.html5rocks.com/zh/tutorials/file/dndfiles/>

# CSS

Cascading Style Sheets层叠样式表

CSS定义的东西，更以更美观地显示 HTML 元素，如边框、字体、形状、距离等等。

你可以在许多地方定义CSS，多处可以定义也意味着定义冲突的存在，因此需要有一个定义优先级，优先级由低到高的顺序如下：

* CSS样式(外部样式)
* head样式（内部样式，位于 <head> 标签内部）
* 内联样式（在 HTML 元素内部）

使用方法如下：

* CSS样式声明：

<link rel=”stylesheet” type=”text/css” href=”ext/resources/ext-theme-neptune/ext-theme-neptune-all.css”>

* head样式：

class优先级：

定义样式可能会**冲突**，元素**根据**<style>最后声明，与<div class>顺序**无关**。

<style>

.class1{color:blue;}

.class2{color:red;}

</style>

<div class="class2 class1">AA</div>

<div class="class1 class2">AA</div>

<style>

.class1{color:blue}

.class2{color:}

</style>

<div class="class2 class1">AA</div>

<div class="class1 class2">AA</div>

由于上述定义的class2非法，则会匹配class1的蓝色

* 内联样式：

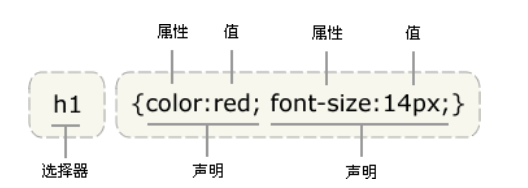
 <div style="position:absolute;margin-left:15%;margin-top:20px;width:1000px;height:400px;background-color:yellow;">

## 单位

|  |  |
| --- | --- |
| 单位 | 描述 |
| % | 百分比 |
| in | 英寸 |
| cm | 厘米 |
| mm | 毫米 |
| em | 1em 等于当前的字体尺寸。 2em 等于当前字体尺寸的两倍。 例如，如果某元素以 12pt 显示，那么 2em 是24pt。 在 CSS 中，em 是非常有用的单位，因为它可以自动适应用户所使用的字体。 |
| ex | 一个 ex 是一个字体的 x-height。 (x-height 通常是字体尺寸的一半。) |
| pt | 磅 (1 pt 等于 1/72 英寸) |
| pc | 12 点活字 (1 pc 等于 12 点) |
| px | 像素 (计算机屏幕上的一个点) |

## 选择器

在外部样式表中用于元素选取，类似jQuery下



基本选择器

| 序号 | 选择器 | 名称 | 说明 | css版本 |
| --- | --- | --- | --- | --- |
| 1 | \* | 通用选择器 | 选择所有元素，包括html和body | 2 |
| 2 | div,span | 标签选择器 | 根据标签选择元素 | 1 |
| 3 | # | id选择器 | 选择指定id元素 | 1 |
| 4 | . | 类选择器 | 选择指定class | 1 |
| 5 | [attr] | 属性选择器 | 根据元素属性去选择 | 2-3 |

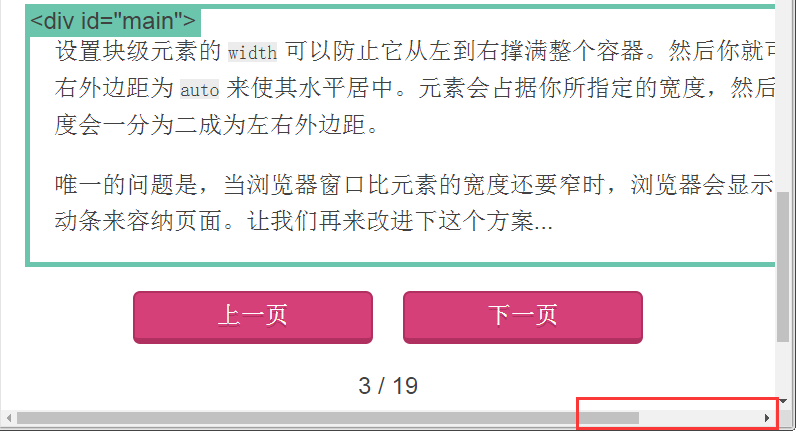
|  |  |  |  |
| --- | --- | --- | --- |
| **选择器** | **例子** | **例子描述** | **CSS** |
| [.*class*](http://www.w3school.com.cn/cssref/selector_class.asp) | .intro | 选择 class="intro" 的所有元素。 | 1 |
| [#*id*](http://www.w3school.com.cn/cssref/selector_id.asp) | #firstname | 选择 id="firstname" 的所有元素。 | 1 |
| [\*](http://www.w3school.com.cn/cssref/selector_all.asp) | \* | 选择所有元素。 | 2 |
| [*element*](http://www.w3school.com.cn/cssref/selector_element.asp) | p | 选择所有 <p> 元素。 | 1 |
| [*element*,*element*](http://www.w3school.com.cn/cssref/selector_element_comma.asp) | div,p | 选择所有 <div> 元素和所有 <p> 元素。 | 1 |
| [*element* *element*](http://www.w3school.com.cn/cssref/selector_element_element.asp) | div p | 选择 <div> 元素内部的所有 <p> 元素。 | 1 |
| [*element*>*element*](http://www.w3school.com.cn/cssref/selector_element_gt.asp) | div>p | 选择父元素为 <div> 元素的所有 <p> 元素。 | 2 |
| [*element*+*element*](http://www.w3school.com.cn/cssref/selector_element_plus.asp) | div+p | 选择紧接在 <div> 元素之后的所有 <p> 元素。 | 2 |
| [[*attribute*]](http://www.w3school.com.cn/cssref/selector_attribute.asp) | [target] | 选择带有 target 属性所有元素。 | 2 |
| [[*attribute*=*value*]](http://www.w3school.com.cn/cssref/selector_attribute_value.asp) | [target=\_blank] | 选择 target="\_blank" 的所有元素。 | 2 |
| [[*attribute*~=*value*]](http://www.w3school.com.cn/cssref/selector_attribute_value_contain.asp) | [title~=flower] | 选择 title 属性包含单词 "flower" 的所有元素。 | 2 |
| [[*attribute*|=*value*]](http://www.w3school.com.cn/cssref/selector_attribute_value_start.asp) | [lang|=en] | 选择 lang 属性值以 "en" 开头的所有元素。 | 2 |
| [:link](http://www.w3school.com.cn/cssref/selector_link.asp) | a:link | 选择所有未被访问的链接。 | 1 |
| [:visited](http://www.w3school.com.cn/cssref/selector_visited.asp) | a:visited | 选择所有已被访问的链接。 | 1 |
| [:active](http://www.w3school.com.cn/cssref/selector_active.asp) | a:active | 选择活动链接。 | 1 |
| [:hover](http://www.w3school.com.cn/cssref/selector_hover.asp) | a:hover | 选择鼠标指针位于其上的链接。 | 1 |
| [:focus](http://www.w3school.com.cn/cssref/selector_focus.asp) | input:focus | 选择获得焦点的 input 元素。 | 2 |
| [:first-letter](http://www.w3school.com.cn/cssref/selector_first-letter.asp) | p:first-letter | 选择每个 <p> 元素的首字母。 | 1 |
| [:first-line](http://www.w3school.com.cn/cssref/selector_first-line.asp) | p:first-line | 选择每个 <p> 元素的首行。 | 1 |
| [:first-child](http://www.w3school.com.cn/cssref/selector_first-child.asp) | p:first-child | 选择属于父元素的第一个子元素的每个 <p> 元素。 | 2 |
| [:before](http://www.w3school.com.cn/cssref/selector_before.asp) | p:before | 在每个 <p> 元素的内容之前插入内容。 | 2 |
| [:after](http://www.w3school.com.cn/cssref/selector_after.asp) | p:after | 在每个 <p> 元素的内容之后插入内容。 | 2 |
| [:lang(*language*)](http://www.w3school.com.cn/cssref/selector_lang.asp) | p:lang(it) | 选择带有以 "it" 开头的 lang 属性值的每个 <p> 元素。 | 2 |
| [*element1*~*element2*](http://www.w3school.com.cn/cssref/selector_gen_sibling.asp) | p~ul | 选择前面有 <p> 元素的每个 <ul> 元素。 | 3 |
| [[*attribute*^=*value*]](http://www.w3school.com.cn/cssref/selector_attr_begin.asp) | a[src^="https"] | 选择其 src 属性值以 "https" 开头的每个 <a> 元素。 | 3 |
| [[*attribute*$=*value*]](http://www.w3school.com.cn/cssref/selector_attr_end.asp) | a[src$=".pdf"] | 选择其 src 属性以 ".pdf" 结尾的所有 <a> 元素。 | 3 |
| [[*attribute*\*=*value*]](http://www.w3school.com.cn/cssref/selector_attr_contain.asp) | a[src\*="abc"] | 选择其 src 属性中包含 "abc" 子串的每个 <a> 元素。 | 3 |
| [:first-of-type](http://www.w3school.com.cn/cssref/selector_first-of-type.asp) | p:first-of-type | 选择属于其父元素的首个 <p> 元素的每个 <p> 元素。 | 3 |
| [:last-of-type](http://www.w3school.com.cn/cssref/selector_last-of-type.asp) | p:last-of-type | 选择属于其父元素的最后 <p> 元素的每个 <p> 元素。 | 3 |
| [:only-of-type](http://www.w3school.com.cn/cssref/selector_only-of-type.asp) | p:only-of-type | 选择属于其父元素唯一的 <p> 元素的每个 <p> 元素。 | 3 |
| [:only-child](http://www.w3school.com.cn/cssref/selector_only-child.asp) | p:only-child | 选择属于其父元素的唯一子元素的每个 <p> 元素。 | 3 |
| [:nth-child(*n*)](http://www.w3school.com.cn/cssref/selector_nth-child.asp) | p:nth-child(2) | 选择属于其父元素的第二个子元素的每个 <p> 元素。 | 3 |
| [:nth-last-child(*n*)](http://www.w3school.com.cn/cssref/selector_nth-last-child.asp) | p:nth-last-child(2) | 同上，从最后一个子元素开始计数。 | 3 |
| [:nth-of-type(*n*)](http://www.w3school.com.cn/cssref/selector_nth-of-type.asp) | p:nth-of-type(2) | 选择属于其父元素第二个 <p> 元素的每个 <p> 元素。 | 3 |
| [:nth-last-of-type(*n*)](http://www.w3school.com.cn/cssref/selector_nth-last-of-type.asp) | p:nth-last-of-type(2) | 同上，但是从最后一个子元素开始计数。 | 3 |
| [:last-child](http://www.w3school.com.cn/cssref/selector_last-child.asp) | p:last-child | 选择属于其父元素最后一个子元素每个 <p> 元素。 | 3 |
| [:root](http://www.w3school.com.cn/cssref/selector_root.asp) | :root | 选择文档的根元素。 | 3 |
| [:empty](http://www.w3school.com.cn/cssref/selector_empty.asp) | p:empty | 选择没有子元素的每个 <p> 元素（包括文本节点）。 | 3 |
| [:target](http://www.w3school.com.cn/cssref/selector_target.asp) | #news:target | 选择当前活动的 #news 元素。 | 3 |
| [:enabled](http://www.w3school.com.cn/cssref/selector_enabled.asp) | input:enabled | 选择每个启用的 <input> 元素。 | 3 |
| [:disabled](http://www.w3school.com.cn/cssref/selector_disabled.asp) | input:disabled | 选择每个禁用的 <input> 元素 | 3 |
| [:checked](http://www.w3school.com.cn/cssref/selector_checked.asp) | input:checked | 选择每个被选中的 <input> 元素。 | 3 |
| [:not(*selector*)](http://www.w3school.com.cn/cssref/selector_not.asp) | :not(p) | 选择非 <p> 元素的每个元素。 | 3 |
| [::selection](http://www.w3school.com.cn/cssref/selector_selection.asp) | ::selection | 选择被用户选取的元素部分。 | 3 |

## 块 & 行内

块/行内元素 通过属性display:block & display:inline 声明；

\*还有另外两种inline-block 和 flex

div元素默认属性就是一个块级元素，块级元素默认width:100%，width可以防止他从左到右充满，如果我设置的是一个像素绝对值500px，那么会有可能出现，缩小的时候出现滚动条->



我不希望有滚动条，我希望有一种手段可以使其宽度不超过一屏：

两份文本内容长度：100px/800px;

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 缺省  (width:100%) | width: 500px;  max-width: 600px; | width: 500px;  max-width: 400px; | **max-width: 600px;** | width: 500px;  max-width: 20%; |
| 初始 | 100% | 500px | 400px | 600px | (500,屏幕\*20%)中小的 |
| 缩进放大 | 100% | 500px | 400px | 100%(364px) | 20% |
| 缩进缩小 | 100% | 500px | 400px | 600px | 500 |

缩进缩小意味着一屏可以放更大尺寸的内容，而此时max进行控制可以

min(屏幕最大宽度尺寸,500px)，即用

max-width=500px

#且初始化为500px

<http://zh.learnlayout.com/max-width.html>

%是基于包含它的块级对象的百分比最大高度。

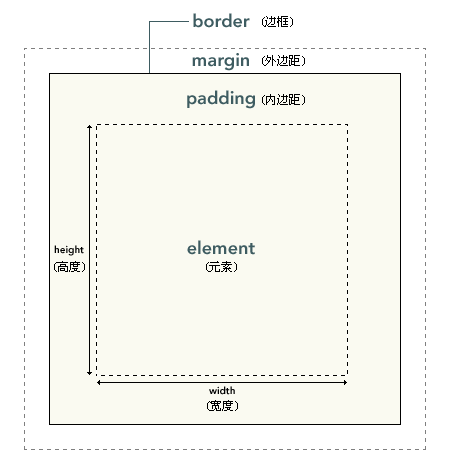
对于高度方面，默认希望填充，在方向上会min(屏幕最大高度,500px)

max-height=500px，且设置了heigth=500px

对于行元素img而言；默认按元素长度而不是100%填充；

max-width：100%：如果img宽度大于div宽度，img就显示div100%宽度，否则就显示img的宽度；  
width:100%：不管img宽度多少，都显示div宽度

### 边框



h1

{margin:10px 5px 15px 20px;}//分别对应上右下左边框

div#navsecond

{

float: left;

margin: 0 0 0 6px;

padding: 0;

border: 0;

width: 180px;

background : transparent;

}

说明：边框是指距离**页面**的距离，并非窗口，当我们调整窗体时的时候，他们会随之变化

路径：

.cool{background-image:url(1/2.gif);}

#### Margin与Padding的区别

         两者都是代替表格最重要的作用->分割块的好方法，区别在于Margin是不同的Tag间互相隔离的距离而Padding是同一元素中不同内容分割使用，这在表格中最看得清楚。尤其是当对一个区域加载了背景样式之后，Padding下的内容背景色会发生改变，而Margin则不会有所改变。

## CSS定位

float: left;margin-top: 5px;

<div id="rel\_div" style="width:400px; z-index:100px; position:absolute; right:20px; float:right; background:#eea; ">

float:right;  
层位于右侧  
  
  
position:absolute; right:20px; top:20px;  
这样是相对固定的。  
  
  
z-index:100px;  
层级于上层。  
  
  
JS jquery 调用，随鼠标移动，层跟着移动。  
$("#rel\_div").css("top", $(document).scrollTop()+20 );//将滚动条高度赋给悬浮框。

## 列式布局

CSS 为定位和浮动提供了一些属性，利用这些属性，可以建立列式布局，将布局的一部分与另一部分重叠，还可以完成多年来通常需要使用多个表格才能完成的任务。

<p style="color: red; margin-left: 20px">

This is a paragraph

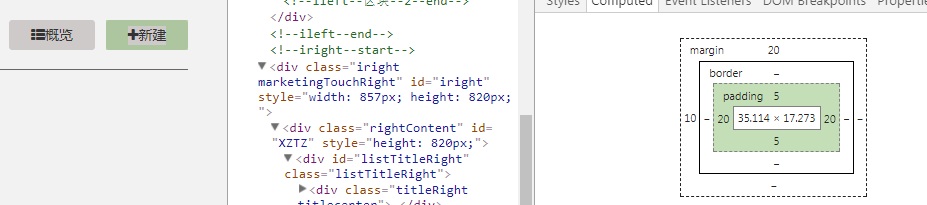
</p>

## 常用样式

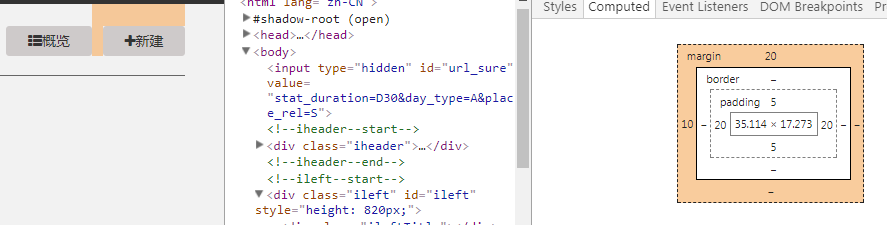


字体可以设置：

**padding**:5**px** 20**px;**



**margin-left**:10**px**;**margin-top**:20**px**;//不可以简写在margin里



上色

{color: '#FFF',

font-weight:'bold',

font-size:'14px'},

{

width: '95%',

marginBottom: '10px'

}

{

color : '#FFF',

font : {

size : '20px',

weight : 'bold'

}

},

例1：设置豆绿色背景

<style type="text/css">

body{background:#CCE8CF;}

</style>

## Resource

Amaze UI

CSS编辑器：

<http://crunchapp.net/>

CSS:background背景样式介绍

<http://www.divcss5.com/rumen/r125.shtml>

# boostrap

教程：**http://www.imooc.com/learn/182**

We start with a portal.

<http://bootstrap.ninghao.net/getting-started.html#html-template>

## resource

Bootstrap

<http://v3.bootcss.com/getting-started/>

Helloworld: <http://www.the5fire.com/bootstrap-introduce.html>

HTML课程<http://ninghao.net/course/1279>

<http://www.w3cschool.cc/bootstrap/bootstrap-intro.html>

表单生成：<http://higrid.net/docs/form/>

[拖放布局的 Twitter Bootstrap 网站生成器](http://maizilin.com/zaixian-Twitter-156)

布局设计：[http://www.bootcss.com/p/layoutit/#](http://www.bootcss.com/p/layoutit/)

## SAMPLE

### 问卷调查(使用麦客——crm)

1、使用分页（使用最基本的jQuery）；

2、页面定时向后台取缓存，关闭时将缓存入库；

3、增加安全：使用HTTPS协议，入库前对敏感字段做加密处理；

4、购买独立数据库服务；

### 选中特效，浮动，背景固定

<!DOCTYPE html>

<html>

<head>

<title>生命小站</title>

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<link href="bootstrap/css/bootstrap.min.css" rel="stylesheet"

media="screen" />

<style type="text/css">

body {

margin: 30px;

}

#bg\_body {

**background: url(img/color.jpg);**

**background-repeat: no-repeat;**

**background-size: cover;**

**background-attachment: fixed;**

}

\* {

font-family: "Microsoft YaHei" ! important;

}

</style>

<script src="bootstrap/js/jquery.js"></script>

<script src="bootstrap/js/unslider.js"></script>

<script src="bootstrap/js/bootstrap.min.js"></script>

<script src="bootstrap/js/stick.js"></script>

<script type="text/javascript">

jQuery(function($) { *//initiating jQuery*

$(document).ready(function() {

*//enabling stickUp on the '.navbar-wrapper' class*

$('.navbar-wrapper').stickUp({

*//enabling marginTop with the 'auto' setting*

marginTop : 'auto',

parts : {

0 : 'Principle 1',

1 : 'Principle 2',

2 : 'Principle 3',

3 : 'Principle 4'

},

itemClass: 'menuItem',

itemHover : 'active'

});

});

});

</script>

</head>

<body id="bg\_body">

<div class="container-fluid">

<div class="row-fluid">

<div class="span2">

<ul id="nav"

class="nav nav-pills nav-stacked sidebar-nav navbar-wrapper"

style="top:100px;width:150px; height:30px">

<li><a href="#Principle 1">Principle 1</a></li>

<li><a href="#Principle 2">Principle 2</a></li>

<li><a href="#Principle 3">Principle 3</a></li>

<li><a href="#Principle 4">Principle 4</a></li>

</ul>

</div>

<div class="span5">

木马：

<http://2sitebbs.com/thread-707-1-1.html>

<http://ninghao.net/video/489>

## MODAL案例

|  |  |  |
| --- | --- | --- |
| 详细展示框 | 登录窗口 | Mobiscroll表单选择框 |
|  |  |  |
|  |  | D:\toshiba\dev\2.h5\2017.cmcc作业\Mobiscroll999920161124\Mobiscroll |
|  |  | $(function () {  // create a datepicker with default settings  $("#scroller").mobiscroll().date();  });  <http://www.cnblogs.com/Look_Sun/p/4482103.html> |

### 日期选择框

$('#demo').mobiscroll().range({

theme: 'mobiscroll',

lang: 'zh',

display: 'bottom',

controls: ['calendar', 'time'],

defaultValue: [ **new** Date(2013, 6, 12), **new** Date(2013, 6, 18, 23, 59) ],

startInput: '#startDate',

endInput: '#endDate'

});

//下面注释部分是上面的参数可以替换改变它的样式

//希望一起研究插件的朋友加我个人QQ:1010305129也可以，本人也建个群 291464597 欢迎进群交流。哈哈。这个不能算广告。

$("#scroller").mobiscroll(opt).date();

// Shorthand for: $("#scroller").mobiscroll({ preset: 'date' });

//具体参数定义如下

{

//preset: 'date', //日期类型--datatime --time,

//theme: 'ios', //皮肤其他参数【android-ics light】【android-ics】【ios】【jqm】【sense-ui】【sense-ui】【sense-ui】

//【wp light】【wp】

//mode: "scroller",//操作方式【scroller】【clickpick】【mixed】

//display: 'bubble', //显示方【modal】【inline】【bubble】【top】【bottom】

//dateFormat: 'yyyy-mm-dd', // 日期格式

//setText: '确定', //确认按钮名称

//cancelText: '清空',//取消按钮名籍我

//dateOrder: 'yymmdd', //面板中日期排列格

//dayText: '日',

//monthText: '月',

//yearText: '年', //面板中年月日文字

//startYear: (new Date()).getFullYear(), //开始年份

//endYear: (new Date()).getFullYear() + 9, //结束年份

//showNow: true,

//nowText: "明天", //

//showOnFocus: false,

//height: 45,

//width: 90,

//rows: 3,

//minDate: new Date() 从当前年，当前月，当前日开始}

<http://blog.csdn.net/libin_1/article/details/50400941>

## 相关资源

EXTJS:modal,

jQuery:nyroModal

http://www.runoob.com/bootstrap/bootstrap-modal-plugin.html

http://www.cnblogs.com/leejersey/p/3601511.html

## 做一个两列MINUS小工具

滤重特效：

<https://segmentfault.com/q/1010000000264761>

TOOLTIP;

# JS框架

jsGen：使用Node.js MONGODB

开源：<http://www.oschina.net/p/jsgen>

案例及交流：<http://www.angularjs.cn/A001>

开发指南：<http://www.angularjs.cn/T008>

SenCha <http://www.sencha.com/products/space/features/>

Octopress基于JEKYII，是一个基于jekyll的博客框架，它整合了Pinboard、Delicious、Disqus、google plus one等插件，还有一个高质量的HTML5模版。虽然octopress将自己标榜为“a blogging framework for hackers”，但普通用户也可以很容易用octopress创建一个漂亮的jekyll博客。

Django+Mongo <http://www.xukailun.me/>

# PHP语法

### FILE API

$text = file\_get\_contents("log\_title.txt");

echo fgets(fopen("log\_title.txt","r"));



### 字符串操作

<http://www.w3school.com.cn/php/php_ref_string.asp>

语法：

|  |  |
| --- | --- |
| 函数 | 说明 |
| explode( string separator, string string [, int limit] ) | 本函数为 implode() 的反函数，使用一个字符串分割另一个字符串，返回一个数组。  separator 分割标志  string 需要分割的字符串  limit 可选，表示返回的数组包含最多 limit 个元素，而最后那个元素将包含 string 的剩余部分，支持负数。  <?php  $text = file\_get\_contents("log\_title.txt");  echo $text;  echo "<br />";  print\_r(explode("\r\n", $text));  ?> |
| .  .= | 连接两个字符串 |
| charAt(0) | 定位某处 |
| slice(1) | 切割字符串.hello,则返还ello |
| [match()](http://www.w3school.com.cn/js/jsref_match.asp) | 找到一个或多个正则表达式的匹配。 |
| [replace()](http://www.w3school.com.cn/js/jsref_replace.asp) | 替换与正则表达式匹配的子串。  注意：如果输入的是串，则只替换第一个；  若要输入正则表达式，则参考：/\$ar/g  转义字符 |
| [search()](http://www.w3school.com.cn/js/jsref_search.asp) | 检索与正则表达式相匹配的值。 |

SAMPLE：<http://www.php100.com/cover/php/540.html>

函数前加@这个是隐蔽（禁止显示）某个语句发生的错误。   
  
如： PHP code mysql\_connect("localhost123","root","不正确密码");

这条语句没有“@”发生错误后，解释器输出Warning。