

前端入门参考：写给前端入门者的chat

原创

置顶 [罗忠浩](#) 于 2017-12-15 22:58:57 发布 17341 收藏 13

分类专栏：[前端开发](#) [差不多先生的前端世界](#) 文章标签：[html](#) [css](#) [前端](#)

版权声明：本文为博主原创文章，遵循[CC 4.0 BY-SA](#) 版权协议，转载请附上原文出处链接和本声明。

本文链接：<https://blog.csdn.net/insistzh/article/details/78817478>

版权



[前端开发](#) 同时被 2 个专栏收录

30 篇文章 1 订阅

订阅专栏



[差不多先生的前端世界](#)

22 篇文章 3 订阅

订阅专栏

前端是什么

众多IT岗位中的一员，以HTML+CSS+JS开发网站、微信公众号、小程序、WEBAPP的岗位，一个夹杂与UI与后台间的岗位。工作流程为从UI处得到原型图或者效果图，在项目（网站、微信公众号、小程序、WEBAPP）中还原图片效果，然后与后台进行各种数据交互。在项目中充当一号背锅侠，项目出现问题第一个收到传唤的人。一个不受UI、后台待见，甚至不受协同开发的其他前端待见的岗位。

前端，

一个让我沉迷的岗位

前端，

一个将一张张UI效果图变成一个生动的项目的岗位 一个将冰冷的后台数据转换为有趣的视觉享受的岗位

我感觉目前的前端市场整体还是处于迅速发展期，市场对于前端的需求也一直比较大，但是前端存在一些比较畸形的地方，门槛低，随便培训上两个月就可以去应聘前端工程师，而培训之前大家从事的岗位是各式各样的，可能和计算机并没有太大关系，这久导致了要想进一步的提升自己比较困难，所以市面上有很多这样的前端，培训出身，反复将一个技术在项目中使用，做到了某种技术的熟练，但是对于新技术的接受能力十分弱。现在市场对于中高级的前端工程师需求更加迫切，所以入了前端的门，就需要不断的提升自己，无论你是学院派，还是培训派，还是自学派，都应该具有强烈的学习意识。

一个项目的起源来自于甲方，可能是你的老板或者你的客户，他们会提出需求，给你们描绘出一个世界。这个抽象的世界会由UI变成一张张实景图，交给你。而填充这个世界的的数据也有由后台为你提供好。而你需要的，就是在获取到这些之后将他们变成现实。

在这之前，你获取到需求之后，你首先应该要做的事情就是-项目分析及项目前期准备。

项目前期准备

如何分析项目并进行前期准备，给大家分享一些经验：

获取到需求之后，首先确定页面，比如首页、登录/注册、XX活动、XX详情、XX介绍等等，往往一个项目可能由几十个页面组成。确认页面之后对页面功能进行分析，比如首页，首页存在轮播、导航、分类、定位、列表、推荐等等功能。组后在确定功能使用的技术，如轮播，是使用第三方插件，还是自己编写。并将以上内容整理成为一个三级列表。如下图

根据需求确定技术选型，确定代码对于浏览器的版本兼容，决定使用什么开发IDE，使用什么框架，使用哪些插件。

进行开发前期准备工作，如：第三方插件的了解学习（如：表格插件），某个功能模块的处理（如：轮播），全局调用模块的封装（如：交互）

enter image description here

同期进行的项目，在你做完这些准备工作的时候，UI及后台应该也做完了相对应的工作。

你拿到UI效果图及后台API文档之后，应该仔细核实，看其中是否存在问题。与UI核实具体效果，因为一张图片，UI表达的和你理解的可能存在误差，为了避免问题发生，在工作开展之前先需要和UI进行样式的核实。

核实完样式之后应当仔细阅读后台的API文档，看每一个接口是否是你需要的，有没有缺漏，有没有数据与你所需的不一致。如果发现问题及时与后台沟通协调。

在完成上述工作之后就准备进入正式开发阶段了，在我们的项目中，文件层次结构一定要清晰明确，便于查找管理，所编写的代码也需要严格按照代码编写规范进行，如果书写的代码IDE一直提示警告，那是不合格的。

每个人都应当结合W3标准形成一套自己的代码标准，下面给大家分享一些我自己的总结：

项目文件规范

项目中的CSS文件、HTML文件、JS文件、其他引入文件、以及静态资源（图片、音频、视频等）都应当建立单独的文件夹存放。

文件名命名

文件命名需要遵循以下几点：

语义化 如：public.css

使用作为连接符 如：indexnav.css

CSS样式规范

CSS样式须注意以下几点：

少用内联样式

写好浏览器兼容

样式名命名语义化、小写、使用'-'或者'_'作为连接符

简写

如：margin-top: 20px; margin-left: 30px; margin-right: 30px; margin-bottom: 40px; 应当直接写为：margin: 20px 30px 40px;

JS中的命名

变量名注意以下几点：

语义化

遵循驼峰命名法

避免保留关键字

常用命名单词表（附在末尾）

在项目的进行过程中，如果你做好了上述的前期工作，肯定会加快你的开发进度，并为你省去很多问题麻烦，但是每个项目还是会存在问题，当我们面临这些问题而又无法自己解决的时候，我们应当怎么解决这些问题呢？我常见的处理方式有以下集中供大家参考：

1) 大问题（如：对于如何使用NPM不清楚），面对这样的问题，首选的方案就是百度教程，类似这些在百度上都可以找到各式各样的教程，基本能解决大家的问题。有时度娘也无法为我们解决（如我进行微信小程序开发，百度提供的资源实在有限），这个时候可以尝试去看官方文档，官方文档中会有大致的讲解，但是很多文档都是英文版的，所以需要良好的英语阅读，推荐大家没事多学习英语。

2) 一般功能点技术问题，时间不急（如：怎样在VUE的列表渲染中渲染指定的下标），这类问题是对于这技术初学者可能经常出现的，首选肯定还是度娘。除此之外也可以在论坛提问盖楼，在各个技术网站中都有大量的人一直在活跃，能针对性的回答你提出的问题，and，加QQ群，一般大型的技术类QQ群都有两千人，多加几个群，肯定有能解决你问题的人。里面还可以找到各类大牛，私聊加他，有问题还可以第一时间骚扰他。

3) 很急的问题，一般遇到这样的问题需要尽快处理，这个时候就强烈推荐你进入各个QQ群寻求帮助了，众多方案中QQ群是效率最高的，但你需要有很多Q群才行，比如我的Q上就有几十个Q群，每个群2000人，一共近10W的前端技术人员，这么大的基数，基本能确保你的问题第一时间得到解决。

也给大家推荐些网站，当你遇到问题的时候可以到这些网站寻求帮助：

创造狮导航，里面有各类网站的链接，里面包括各类技术网站，学习网站，框架网站，论坛链接，工具网站，收藏一个能省去很多事儿。<http://chuangzaoshi.com>

果壳任意门，一个很有个性的导航网站，内容也很丰富，不过最近好像不能访问了（404）。<https://gate.guokr.com/>

阮一峰的个人网站，大牛的个人网站，内容极其丰富。<http://www.ruanyifeng.com/home.html>

jQuery插件库，内容最丰富的插jQuery件库，你需要的里面总是有的。<http://www.jq22.com/>

CSDN，中国最大的IT社区和服务平台。<https://www.csdn.net/>

博客园，博客园是一个面向开发者的知识分享社区，一个纯净的技术交流社区。<https://www.cnblogs.com/>

学习CSS布局，一个CSS基础教程，适合初学者。<http://zh.learnlayout.com/>

前端乱炖。<http://www.html-js.com/>

github一个面向开源及私有软件项目的托管平台。<https://github.com/>

前端网.里面涉及前端的各类问题，整个网站只有前端内容。<http://www.qdfuns.com/>

segmentfault技术交流平台。<https://segmentfault.com/>

tips：关于前端学习的网站，前端内容的网站都可以在创造狮导航中找到，就不再单独列举。

当一个项目顺利或者不顺利的结束之后，我们可能有一段休息时间，也可能还得继续下一个项目，甚至项目多个项目同时进行的情况，但是每个项目结束之后，都希望大家抽出时间来整理下该项目的心得，将项目中遇到问题，解决方案，新学习的内容，整理出来分享给大家。我每个月大概会在论坛发布3-6篇博客，将我遇见的和我收获的都分享给大家。

在工作之余，我不免会对未来的工作产生一些想法，我的未来该是怎么样的？

如果你也曾想过，那么我想你或许也需要对你的前端生涯进行一场规划了。

职业生涯规划

什么是生涯规划？

它是结合自身情况以及眼前的机遇和制约因素，为自己确立职业目标，选择职业道路，确定教育、培训和发展计划等，并为自己实现职业生涯目标而确定行动方向、行动时间和行动方案。

为什么要进行规划，两个原因？

明确的职业生涯规划会让你知道在什么阶段你该做些什么，让自己做的事具有比较明确的目的，而非混混度日。

用人单位需要你的职业生涯规划，以判断你是否符合公司的发展。

如何进行规划？

我在大学的时候与职业生涯规划的老师谈过许多，其中有一点句话我认为是最正确的，“你的职业不一定是你100%热爱的，但它一定不是你厌恶的。”不需要有陶渊明的不为五斗米折腰，但如果本身都对这份工作反感，那么它就不应该成为你职业生涯中的道路。

前端生涯规划的第一步，就是确认你对前端这份工作的热爱，如果前端工作让你心烦意乱，感觉到代码让你很不舒服，你开始憎恶代码，那么时候考虑下暂停下工作，是否要继续前端工作了。

Step2: 正确的评估自己，了解自己的状态及能力。

Step3: 制定目标，给自己确定目标，在什么时候我要达到什么样。

Step4: 拆分目标，把离现在最近的一个目标拆分，清楚自己如何做才能达到这个目标，为这个目标制定一系列的计划。

Step5: PDCA循环（制定计划、执行计划、检查计划、调整计划）根据自己制定的计划去执行，然后在执行的过程中不断检查自己是否与计划一致，如果存在问题，进行调整，或者计划已经不在符合现有阶段，那就重新制定新计划。如此反复循环。

Step6: 坚持-成功。坚持不一定会成功，但坚持一定是成功的要素之一。

最后为大家附上一些我对于前端的一些整理包括我接触或者了解过的技术以及对他们的看法以及一份娱乐性的前端等级划分：

前端各类框架/库

Q: 前端框架/库如何划分？

A: 前端框架/库有很多划分方式，我把他们分为CSS框架/库、JavaScript框架/库以及插件类框架/库。

下面从数以百计的前端框架/库中挑选部分主流框架和大家分享：

enter image description here

一、CSS框架/库

Bootstrap

Bootstrap来自于Twitter，是当前最受欢迎的前端框架之一，简洁、直观、强悍、兼顾PC与移动开发，是前端必备的一款框架，关于它的好不用多加阐述。<http://www.bootcss.com/>

fontawesome

一款风靡全球的图标字体库和CSS框架。<http://www.fontawesome.com.cn/>

MUI

最接近原生APP体验的高性能前端框架。<http://dev.dcloud.net.cn/mui/>

AUI

一款为APICloud而设计的移动端前端框架。<http://www.aucss.com/>

Amaze UI

基于社区开源项目构建的一个跨屏前端框架，以移动优先，从小屏到大屏，最终实现所有屏幕适配，适应移动互联网潮流。<http://amazeui.org/>

WeUI

框架本身适用于微信开发，是小程序与<http://github.com/Tencent/weui>

二、JavaScript框架

jQuery

jQuery 一个快速、简洁的JavaScript框架，它极大地简化了 JavaScript 编程，是一个人手必备的JavaScript框架，也许在将来会成为墓地中的一员，但现在任然是不可或缺的一个主流框架。<http://jquery.com/>

Zepto

Zepto是一个轻量级的针对现代高级浏览器的JavaScript库，它与jquery有着类似的api。如果你会用jquery，那么你也会用zepto。<http://zeptajs.com/>

angular

当前三大主流框架之一，AngularJS是为了克服HTML在构建应用上的不足而设计的，它具有模块化、自动化双向数据绑定、语义化标签、依赖注入等特性，是一款值得推荐值得学习的框架。<http://www.angularjs.net.cn/>

Vue

Vue来源于尤雨溪，是一款来自国人的框架，深受各位前端工程师的欢迎，它是一套构建用户界面的渐进式框架，采用自底向上增量开发的设计，关注视图层，简单易于上手，拥有的第三方资源，是当前国内热度最高的框架之一。<https://cn.vuejs.org/>

React

React 起源于 Facebook 的内部项目，是一个 Facebook 和 Instagram 用来创建用户界面的JavaScript 库。发展至今已经成为当下最火的前端框架之一，React 基本上用来作为 MVC 中的 V。它完全专注于 MVC 中的 V 部分，而忽略了应用中的其它部分架构。它提供了一个组件层，这让我们易于制作UI元素并将这些UI元素组合在一起。它抽象出了DOM，因此而优化了渲染，并允许你使用 node.js 渲染 React。<https://reactjs.org/>

Node

Node.js是一个Javascript运行环境(runtime)。简单的说 Node.js就是运行在服务端的 JavaScript。Node.js的出现，使得前后端使用同一种语言，统一模型的梦想得以实现。<http://nodejs.cn/>

三、纯插件类框架/库

EasyUI

easyui是一种基于jQuery的用户界面插件集合，它为创建现代化，互动，JavaScript应用程序，提供必要的功能。<http://www.jeasyui.net/>

ECharts

一个纯 Javascript 的图表库，可以流畅的运行在 PC 和移动设备上，兼容当前绝大部分浏览器（IE8/9/10/11，Chrome，Firefox，Safari等），底层依赖轻量级的 Canvas 类库 ZRender，提供直观，生动，可交互，可高度个性化定制的数据可视化图表。<http://echarts.baidu.com/index.html>

hcharts

全球知名的 Web 数据可视化图表软件，用于创建可交互性图表，性能好，功能强大，支持 IE6、Android、IOS，同时支持股票图表、地图图表。<https://www.hcharts.cn/>

tips: 除了以上还有Extjs、prototype、kissy，jQuery UI、flex、yui、Dojo等等以及基于以上几个框架衍生的优秀的框架/库和适用于移动端开发的框架/库可供大家选择，大家可以在各论坛中查询各框架的特点。

框架的选择：

Q: 在以上的那么多框架中，有哪些是我们必备的框架？

A:

JQuery, 如果说只会一个框架, 那么这个框架应该就是它了。一个轻量级框架, 拥有强大的选择器, 出色的DOM操作封装, 优秀的兼容性, 最为丰富的插件资源。很难找到不学它的理由。只要会javascript, 学习jQuery就是一件轻松的事。

当前三大主流框架: angular、Vue、React, 这三者已经占领了整个前端圈, 不说三者都会, 三者你肯定需要学习其中一种, 才能让你立足当前。

node, 基于javascript的服务端语言, 一个出色的Javascript运行环境, 它有一个很棒的包管理系统NPM。通过运行“npm install 包名”几乎可以安装任何包/库, 在开发工作中为你提供各种便利。

Bootstrap, 一套足以满足95%需求的CSS框架, 同时兼顾PC与移动端, 只需要这一个框架你就可以完成一个完整的项目。

以上几种框架都具有相当活跃的社区, 在学习和工作中出现的各种问题都能够在社区中找到答案。

每个前端工程师在工作后的一段时间都会开始了解MVVM框架, MVVM框架成为当前主流, 下面简单的为大家讲解MVVM框架:

Q: 什么是MVVM框架, 它有什么特点?

A: MVVM是Model-View-ViewModel的简写, 它来源于MVC框架, 在MVVM中, 数据是核心, 由于ViewModel与View之间的双向绑定, 操作了ViewModel中的数据, 就会同步到DOM, 我们透过DOM事件监控用户对DOM的改动, 也会同步到ViewModel, 很好做到数据的一致性。

Q: 当前主流MVVM有哪些?

A: vue+angular成为了绝大多数前端的选择, 微信开发小程序其实也是MVVM框架结构, react也能算作一款(因为React本身特殊, 不是纯粹的MVVM框架机构), 除此之外还有avalon、knockoutjs等等。

Tips: 在我的博客及后续内容中将专门针对于MVVM框架进行详细讲解, 欢迎大家前往我博客查看或继续关注后续chat。

常用工具

前端开发是最简单的, 只需要一个文本编辑器以及一个浏览器就OK了。但是工具更能为我们提供无数的便利, 提高我们的开发效率和开发质量。下面跟大家分享一些前端常用工具:

一、开发IDE

HBuilder

新手入门最好用的IDE, 基于eclipse深度封装, 全中文界面, 完善的代码自动补全, HTML+CSS+JQ+JS, 支持emmet输入, 满足新手的大部分应用。

优点: 傻瓜化安装, 全中文, 出色的智能提示;

缺点: 没有社区支持, 代码补全有点老旧, 对预处理器支持不太好;

WebStorm

全能之选, 大多数人当前的使用选择, 包括我, WebStorm可以实现针对JavaScript、TypeScript、stylesheet语言和其他最流行框架的智能代码补全、实时错误检测、以及强大的导航和重构功能。可以说WebStorm几乎拥有所有你期待的功能。使用WebStorm尽量不要使用汉化包, 容易出现错误。

Microsoft Visual Studio 2017

Microsoft Visual Studio 2017用于.net开发, 但是同样可以作为一款前端开发软件, 在我之前的公司就使用它作为开发IDE,它能够快速导航、编写并修复你的代码, 实现轻松调试和诊断, 根据你的喜好进行扩展和自定义, 并实现有效协作, ASP开发首选。

notepad++

非常小巧的一款IDE, 适合用作前端代码编写, 但是扩展功能有限, 如果设计复杂部分, 不推荐使用。

sublimeText3

非常小巧，启动速度特别快，基本替代了对文本档的使用，虽然是闭源软件，但是有着丰富的第三方支持，能够满足各种各样的扩展。并且最好用的地方在于可以通过本地node进行编译 如果把前端比作LOL，项目比作排位赛。

二、其他类工具

WebPack

WebPack可以看做是模块打包机：它做的事情是，分析你的项目结构，找到JavaScript模块以及其它的一些浏览器不能直接运行的拓展语言（Scss，TypeScript等），并将其转换和打包为合适的格式供浏览器使用。

gulp

Gulp.js 是一个自动化构建工具,开发者可以使用它在项目开发过程中自动执行常见任务。Gulp.js 是基于 Node.js 构建的,利用 Node.js 流的威力,你可以快速构建项目。

SVN

SVN是Subversion的简称，是一个开放源代码的版本控制系统,当前使用最广泛的版本控制之一，一般项目开发团队之间都使用SVN进行配合开发。

Git

Git是一个开源的分布式版本控制系统，可以有效、高速的处理从很小到非常大的项目版本管理。

NPM

NPM是随同NodeJS一起安装的包管理工具，能解决NodeJS代码部署上的很多问题，常见的使用场景有以下几种：

允许用户从NPM服务器下载别人编写的第三方包到本地使用。

允许用户从NPM服务器下载并安装别人编写的命令行程序到本地使用。

允许用户将自己编写的包或命令行程序上传到NPM服务器供别人使用。

FIS

FIS，前端工程构建工具，解决前端开发中自动化工具、性能优化、模块化框架、开发规范、代码部署、开发流程等问题

grunt

JavaScript世界的构建工具，自动化。对于需要反复重复的任务，例如压缩（minification）、编译、单元测试、linting等，自动化工具可以减轻你的劳动，简化你的工作。当你在 Gruntfile 文件正确配置好了任务，任务运行器就会自动帮你或你的小组完成大部分无聊的工作。

Firebug

Firebug是firefox下的一个扩展,能够调试所有网站语言，如Html,Css等，但FireBug最吸引人的就是javascript调试功能，使用起来非常方便，而且在各种浏览器下都能使用（IE,Firefox,Opera, Safari）。除此之外，其他功能还很强大，比如html,css,dom的查看与调试，网站整体分析等等。总之就是一整套完整而强大的 WEB开发工具。

YSlow

YSlow 是Yahoo发布的一款基于FireFox的插件，这个插件可以分析网站的页面，并告诉你为了提高网站性能，如何基于某些规则而进行优化。

JSLint

JSLint，一个用于提升你所编写的代码质量的工具，它是一个JavaScript验证工具（非开源），可以扫描JavaScript源代码来查找问题。如果JSLint发现一个问题，JSLint就会显示描述这个问题的消息，并指出错误在源代码中的大致位置。有些编码风格约定可能导致未预见的行为或错误，JSLint除了能指出这些不合理的约定，还能标志出结构方面的问题。

sass

官网介绍：世界上最成熟、最稳定、最强大的专业级CSS扩展语言！Sass完全兼容所有版本的CSS。我们对此严格把控，所以你可以无缝地使用任何可用的CSS库。

Less

Less 是一门 CSS 预处理语言，它扩展了 CSS 语言，增加了变量、Mixin、函数等特性，使 CSS 更易维护和扩展。它可以运行在 Node 或浏览器端。

PxCook markman

这两款都是用于标注的小工具，用于和UI配合。

前端等级划分

对于前端等级我有一套娱乐性的划分，供大家参考。把前端比作LOL，每个项目就是一场排位赛。

LV1：萌新-未入门

肯定就是1-29级的玩家：没法参加排位赛，无法开展项目。

这部分玩家刚开始接触前端，或者对于CSS不够熟练，或者对JS知道的还不够，所掌握的东西还没有办法独立完成一个完整的页面，让他们去看一个完整的项目，他们就会感觉，这是啥，这又是啥。

LV2：入门-青铜（初级）

这部分玩家具备一定的基础能力，能够自己完成一些简单的页面，但是对于一些比较复杂的逻辑不能理解通透，或者对于某些特定样式不能完成，遇到问题比较多时候都束手无策，比较喜欢直接询问别人如何解决，代码质量较渣，这些玩家在项目中往往会给你分配一些比较简单的页面。

LV3：进阶-白银（初级）

这部分玩家已经掌握了一定的进阶技术，能够在一个项目中承担各种任务，也能够自己解决一些比较常见的问题，有了一定的项目经验，能够总结出遇到的问题，避免各种简单的错误，书写的代码可供阅读，这类玩家在项目中能够胜任大多数的开发工作。

LV4：崛起-黄金（中级）

此类玩家已经具有相当的经验，已经成长为能独挡一面的大将，具备相当的学习能力，遇到问题都能自己独立快速解决，代码内容都是经过优化，不存在过多的代码冗余，工作之余能够选择合适的新技术来扩充自己，以迎接下一阶段的挑战。这部分玩家在项目中挑起大梁，在完成自己工作的同时帮助其他人解决问题。

LV5：成熟-铂金（高级）

前端项目中leader，拿到一个项目知道应该如何选择合适的技术框架，应当如何针对项目进行项目规划。精通组件式的开发，高效率，对前端各种技术都有自己的认知，对于后台技术也有所了解，从项目的前期设计到后期维护方案都参与，在项目中属于整个项目的前端负责人。

LV6：蜕变-钻石

掌握了前端底层的精髓，能够基于底层编写自己所需的所有东西，对前端的认识已经达到入微境界。这类人一般已经不在直接编写项目代码了，他们的工作更多的是让别人更方便的工作，各类框架就是出自于他们之手。

LV7：登顶-大师

大师级的人物，在专业内具有相当的权威，各个书店都能买到他们的作品，他们左右着技术的走向。

LV8：封神-王者

他们的名字流传于每个互联网的角落，他们在全球技术圈都具有相当的影响力，他们正在改变未来。

附（常用命名单词列表）

wrapper 页面外围控制整体布局宽度
container或#content 容器,用于最外层
layout 布局
head, #header 页头部分
foot, #footer 页脚部分
nav 主导航
subnav 二级导航
menu 菜单
submenu 子菜单
sideBar 侧栏
sideBar, #sideBar 左边栏或右边栏
main 页面主体
tag 标签
msg /message 提示信息
tips 小技巧
vote 投票
friendlink 友情连接
title 标题
summary 摘要
loginbar 登录条
searchInput 搜索输入框
hot 热门热点
search 搜索
search_output 搜索输出和搜索结果相似
searchBar 搜索条
search_results 搜索结果
copyright 版权信息
branding 商标
logo 网站LOGO标志
siteinfo 网站信息
siteinfoLegal 法律声明
siteinfoCredits 信誉
joinus 加入我们
partner 合作伙伴
service 服务
register 注册
arr/arrow 箭头
guild 指南
sitemap 网站地图
list 列表
homepage 首页
subpage 二级页面子页面
tool, #toolbar 工具条
drop 下拉
dropdown 下拉菜单
status 状态
scroll 滚动
.tab 标签页
.left .right .center 居左、中、右
.news 新闻
download 下载

.download 下载

.banner 广告条(顶部广告条)

电子贸易相关

.products 产品

.products_prices 产品价格

.products_description 产品描述

.products_review 产品评论

.editor_review 编辑评论

.news_release 最新产品

.publisher 生产商

.screenshot 缩略图

.faqs 常见问题

.keyword 关键词

.blog 博客

.forum 论坛

CSS文件命名 说明

master.css,style.css 主要的

module.css 模块

base.css 基本共用

layout.css 布局, 版面

themes.css 主题

columns.css 专栏

font.css 文字、字体

forms.css 表单

mend.css 补丁

print.css 打印

//“大清亡于闭关锁国，技术需要交流。”希望大家能多在论坛/Q群活跃，多与大家沟通交流。交流能够助你快速成长。