

编程笔记1 - 扫盲 20190811-12

以下笔记了解了大概什么是前端后端，什么是请求/响应/HTTP协议，什么是静态页面及动态页面，什么是Web服务器

集成开发环境，什么是后端语言的应用逻辑。对于自己项目实现，打算用HTML+CSS+Javascript实现。JS可用前端jQuery框架及后端Node.js框架，集成环境用MAMP (apache服务器+mySQL数据库)。

20190811note06 前后端语言 - 视频及教程入门 -----

My结论: HTML标记网页内容, CSS页面修饰布局, JS管页面动态与事件交互, PHP管逻辑。

快速入门视频教程（简单有用）

12分钟了解HTML-1 : <https://www.youtube.com/watch?v=bWPMSSsVdPk>

基本文字及大小 / 插入图片

12分钟了解HTML-2 : <https://www.youtube.com/watch?v=KJ13IX20FqU>

文件申明<!DOCTYPE html> / 插入链接 / 图片属性 / 容器概念div ID

HTML + CSS: Html是标记性语言而已。css是给html加装饰效果的（字体 颜色等）。

HTML / XML: 超文本标记语言与可拓展标记语言。前者用于网页编辑，后者用于数据格式化储存。

Javascript: 是一种客户端语言client-side language。这才是编程语言！HTML+CSS用于静态页面， Js用于动态交互页面。PHP: 是一种服务器端语言server-side language。

12分钟了解CSS : <https://www.youtube.com/watch?v=0afZj1G0BIE>

简单装饰HTML中的文本

15分钟了解PHP : <https://www.youtube.com/watch?v=ZdP0KM49IVk>

变量 / 级联concatenation / 操作 / IF从句 / 处理HTML来的数据 / 数组 arrays / 循环Loops

```
<form action="process.php" method="post">
  Enter your name:<input name="name" type="text">
  <input type="submit">
</form>
```

12分钟了解Javascript : https://www.youtube.com/watch?v=Ukg_U3CnJWI

30分钟例子使用jQuery <https://www.youtube.com/watch?v=zPHerhks2Vg>

12分钟了解Docker : <https://www.youtube.com/watch?v=YFI2mCHdv24>

Docker 开源的应用容器引擎 <https://baike.baidu.com/item/Docker> 和

<https://juejin.im/post/5b260ec26fb9a00e8e4b031a>

后端教程

Java Servlet教程 : <http://https://www.runoob.com/servlet/servlet-intro.html>

自动刷新页面 / 点击计数器 / 数据库访问等

Servlet = Service + Applet，表示小服务程序。运行在web服务器端，处理HTTP请求并响应。

相关概念：JSP(Java Server Pages), CGI(传统/效率较低), Applet(衔接在网页中，被调用/具有图形界面),

前端教程&知识

jQuery教程及语法 <https://www.runoob.com/jquery/jquery-syntax.html>

HTML 教程及语法 <http://www.runoob.com/html/html-tutorial.html>

Javascript 教程W3School <https://www.w3school.com.cn/js/index.asp>

脚本语言知识总结 / Javascript / ECMAScript <https://www.cnblogs.com/wang-meng/p/5702004.html>

Java是面向对象，写Java程序，写类和对象。JavaScript是基于对象，写Js，不用创建类，使用Js内部已经定义好的对象。

Javascript浏览器对象BOM: **DOM Window / DOM History / DOM Location**

其他看的手册：

<https://www.w3schools.com>

20190811note05 使用本地服务器 / 开发集成环境 -----

LAMP, **WAMP**, **MAMP** – They are really just referring to an Apache-MySQL-PHP web server stack on Windows, Linux, and Mac.

lamp是linux（系统）+apache（开发环境）+mysql（数据库工具）+[php5](#)（一种语言）

mamp是mac OS +apache（开发环境）+mysql（数据库工具）+[php5](#)（一种语言）

wamp是windows（系统）+apache（开发环境）+mysql（数据库工具）+[php5](#)

百度经验：PHP 使用MAMP快速搭建PHP环境

<https://jingyan.baidu.com/article/948f5924ea3959d80ff5f9f3.html>

MAMP PRO4 FOR MAC 2019

<https://inaladdin.com/216.html/>

Mac OS X 中配置 MAMP 服务器

<http://easior.is-programmer.com/posts/48924.html>

https://netbeans.org/kb/docs/php/configure-php-environment-mac-os_zh_CN.html

本教程向您演示如何在 MAMP (Macintosh、Apache、MySQL、PHP) 包上设置 PHP, 该包中含有 Apache Web 服务器、MySQL 数据库服务器和 PHP 引擎。MAMP 用作 Mac 的 PHP 开发环境, 可与 NetBeans IDE 很好地配合使用。Xdebug 是 PHP 的扩展, 可帮助提供对 PHP 脚本调试有用的信息。NetBeans 调试器在 Xdebug 扩展之上运行, 在您的开发环境中提供了有效的调试工具。

知乎文章：全栈 - 10 数据库 用MAMP和WAMP搭建Web环境

<https://zhuanlan.zhihu.com/p/25655297>

Web环境 = Web服务器 + 数据库 + 后端语言 (系统偏好设置教程)

知乎文章：全栈 - 11 数据库sql入门

<https://zhuanlan.zhihu.com/p/25724365>

本地电脑设置 (作用不大?) https://miaowm5.github.io/RMMV-F1/page/01_11_08.html

WAMP与LAMP比较 <https://blog.51cto.com/caojh/40605>

20190811note04 什么是服务器？

1.1 编译型语言 (直接机器语言, 如C++)

1.2 解释型语言 (需要Interpreter解释, 如Python / Js / VB)

即: **脚本语言**: Scripting Language, 为缩短传统的编写-编译-链接-运行 (edit-compile-link-run) 过程而创建的语言。

1.2.1 标记性语言 (Markup Language, 标记文档结构。超文本HTML, 可拓展XML, 分类中Procedural markup的, 如Tex)

非常棒的解释: Web Server / Application Server

WEB服务器with/without应用程序服务器

详细比较请看以下链接

<https://blog.csdn.net/u010098331/article/details/51879450>

例如, 设想一个在线商店(网站)提供实时定价(real-time pricing)和有效性(availability)信息。这个站点(site)很可能会提供一个表单(form)让你来选择产品。当你提交查询(query)后, 网站会进行查找

(lookup)并把结果内嵌在HTML页面中返回。网站可以有很多种方式来实现这种功能。我要介绍一个不使用应用程序服务器的情景和一个使用应用程序服务器的情景。观察一下这两中情景的不同会有助于你了解应用程序服务器的功能。

Web服务器只是简单的通过响应(response)HTML页面来处理HTTP请求(request)。

Request - handle - Response

1, 响应 response : 响应静态页面, 实现页面跳转redirect ; 动态响应(dynamic response)则把产生委托(delegate)给其他脚本如CGI, JSP, Javascript等服务器端技术(Server-side)。

在Web服务器通过回应HTML页面来处理HTTP请求(request), 而应用程序服务器则是通过处理定价和有效性(availability)请求(request)来提供应用程序逻辑的。 在大多数情形下, 应用程序服务器是通过组件 (component) 的应用程序接口(API)把商业逻辑暴露(expose)(给客户端应用程序)的, 例如基于J2EE(Java 2 Platform, Enterprise Edition)应用程序服务器的EJB(Enterprise JavaBean)组件模型。此外, 应用程序服务器可以管理自己的资源, 例如看大门的工作(gate-keeping duties)包括安全(security), 事务处理(transaction processing), 资源池(resource pooling), 和消息(messaging)。就象Web服务器一样, 应用程序服务器配置了多种可扩展(scalability)和容错(fault tolerance)技术。

Web服务器 / 应用服务器 / Web容器 / 反向代理服务器

<https://www.cnblogs.com/vipyoumay/p/7455431.html>



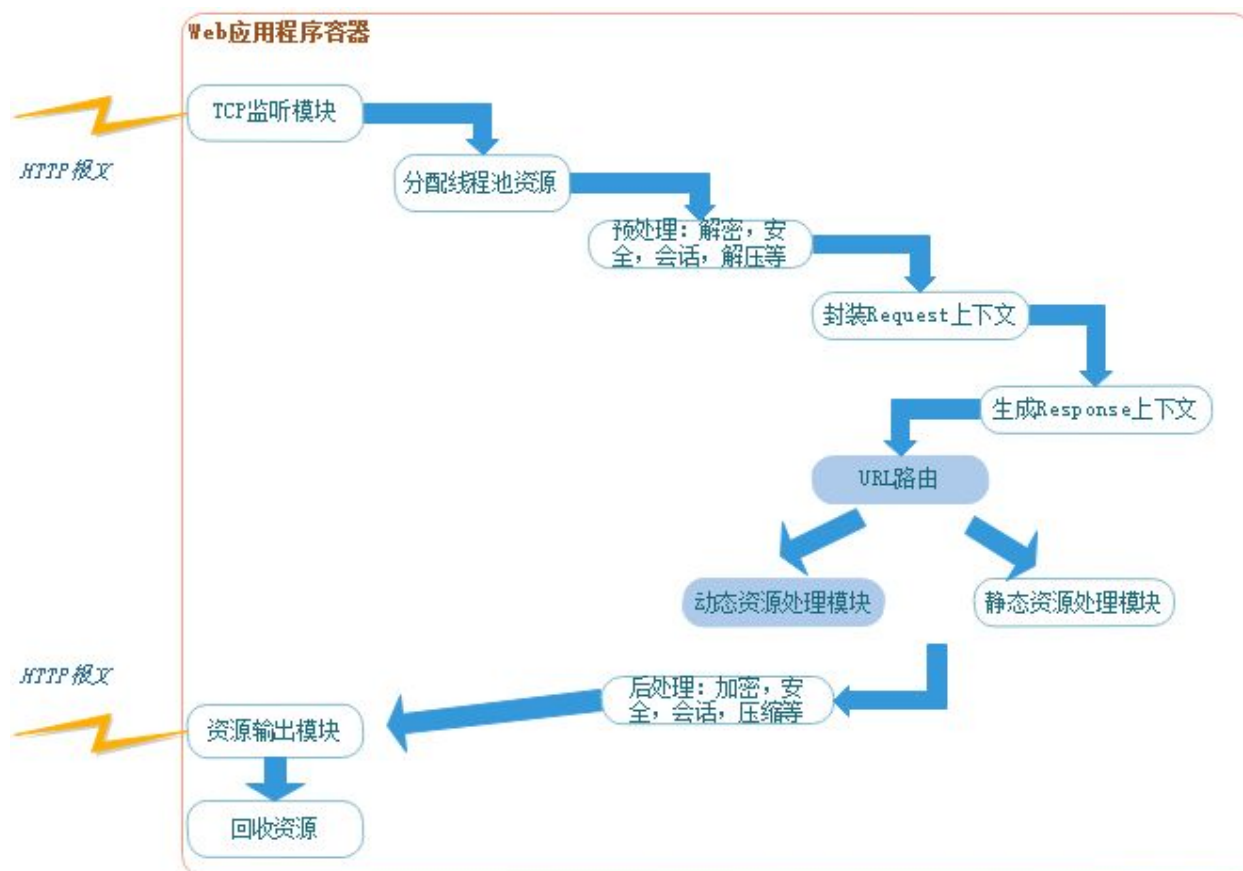
Web服务器的主要功能是存储, 处理和传递网页给客户。客户端和服务端之间的通信使用超文本传输协议 (HTTP) 进行。交付的页面最常见的是HTML文档, 除了文本内容之外, 还可能包含图像, 样式表和脚本。

Web服务器的出现标志着WWW时代的带来, 人们开始不满足与在互联网上获取静态资源, 于是出现了CGI脚本来动态获取资源。再后来网络发展方向也是朝着增强Web服务器动态获取资源的能力前进。以下是代表性的动态技术 :

技术名词	特点
CGI (Common Gateway Interface, 公用网关接口)	以独立进程运行, 可以用多种语言开发, 比如C,C++,VB,Perl, 灵活但效率低, 维护复杂
PHP	服务器端嵌入HTML脚本, 开源, 功能强大, 扩展性较差
JSP	服务器端嵌入HTML脚本, 跨平台, 部署前需编译, 主要缺点是编写JSP比较复杂, 需熟悉JAVA及相关技术
ASP	服务器端嵌入HTML脚本, 开发简单, 功能强大, 只能在windows下运行

创建一个Web程序开发框架（含运行环境，比如解释执行JSP,Web API），这个框架解决鲁棒性，可靠性问题，提供快速开发接口。换言之，开发人员只需要专注于实现业务本身，如有更高的需求还可以对框架进行定制和扩展。这个框架的另外一个名字是Web应用程序容器。

反向代理是代理服务器的一种。它根据客户端的请求，从后端的服务器（如Web服务器）上获取资源，然后再将这些资源返回给客户端。



Apache是纯粹的**web服务器**，经常与Tomcat配对使用。它对HTML页面具有强大的解释能力，但是不能解释嵌入页面内的服务器端脚本代码。

Tomcat是一个独立的Servlet（[Servlet菜鸟教程](#)）和JSP容器，还提供了JNDI和JMX API的实现机制。业务逻辑层代码和界面交互层代码可以分离了。因此，有人把Tomcat叫做轻量级**应用服务器**或带有服务器功能的Web服务器。

[Java Servlet](#) 是运行在 Web 服务器或应用服务器上的程序，它是作为来自 Web 浏览器或其他 HTTP 客户端的请求和 HTTP 服务器上的数据库或应用程序之间的中间层。使用 Servlet，您可以收集来自网页表单的用户输入，呈现来自数据库或者其他源的记录，还可以动态创建网页。实用工具例程：如自动地解析和解码 HTML 表单数据、读取和设置 [HTTP](#)头、处理[Cookie](#)、跟踪会话状态等。

后来 SUN 推出了类似于 ASP 的镶嵌型的 JSP，把 JSP TAG 镶嵌到 HTML 语句中，这样，就大大简化和方便了网页的设计和修改。新型的网络语言如 ASP，PHP，JSP 都是镶嵌型的语言。这是 JSP 和 Servlet 区别的运作原理层面。

综上所述，Servlet 是一个早期的不完善的产品。从网络三层结构的角度看 JSP 和 Servlet 的区别，一个网络项目最少分三层：data layer(数据层)，business layer(业务层)，presentation layer(表现层)。当然也可以更复杂。Servlet 用来写 business layer 是很强大的，但是对于写 presentation layer 就很不方便。并且两层混杂。所以，推出JSP+BEAN，用 JSP 写 presentation layer，用 BEAN 写 business layer。SUN 自己的意思也是将来用 JSP 替代 Servlet。这是技术更新方面 JSP 和 Servlet 的区别。强调的是：学了JSP，不会用 Java BEAN 并进行整合，等于没学。大家多花点力气在 JSP+BEAN 上。

20190811note03 实时Web应用 / 后端逻辑 -----

4.

5.Web Socket https://www.ibm.com/developerworks/cn/web/1112_huangxa_websocket/

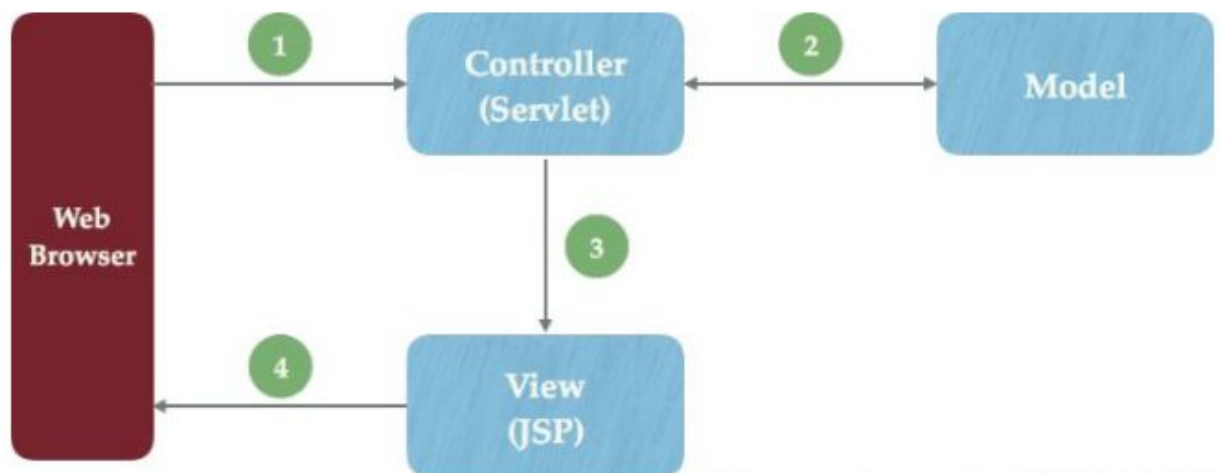
- 传统实时Web的实现方式：长轮询机制，Comet流技术。（用 Ajax 方式来模拟实时的效果）

- 现在HTML5 WebSocket

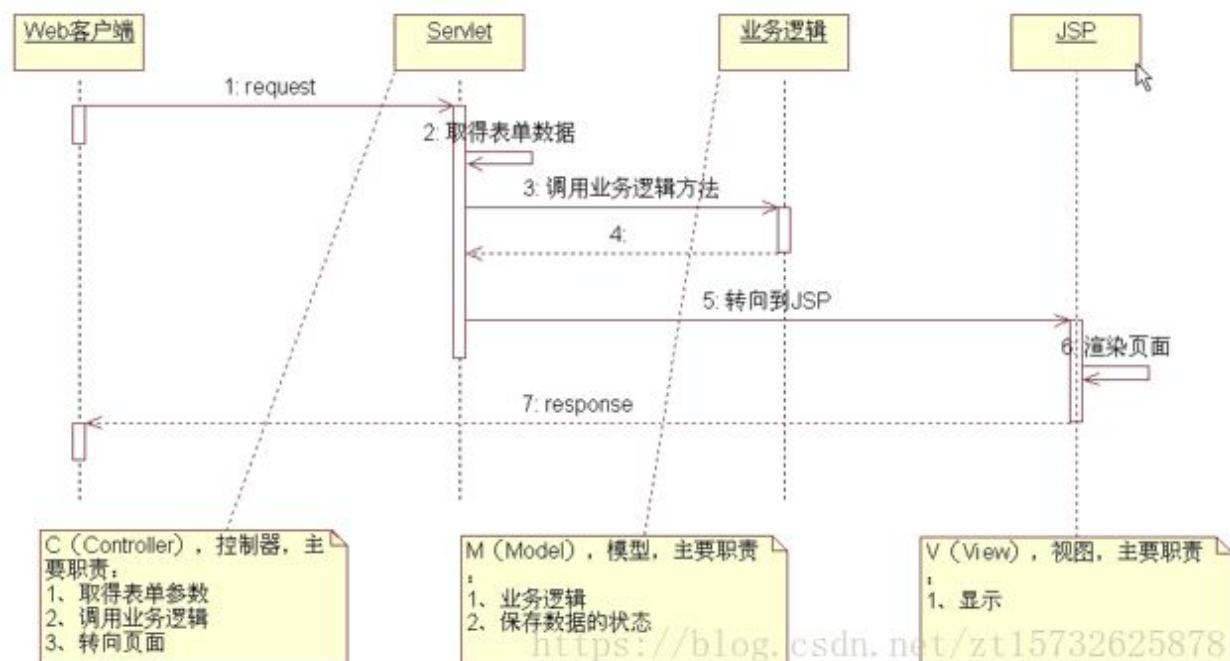
6. PHP后端该如何组织项目结构 <https://blog.csdn.net/hel12he/article/details/76855745>

7. JSP与Servlet区别， Model-Controller-View (MCV) 模型

<https://blog.csdn.net/zt15732625878/article/details/79951933>



<https://blog.csdn.net/zt15732625878>



<https://blog.csdn.net/zt15732625878>

20190811note02 后台与前端交互 / Js 实例 -----

1. Python后台与JS交互全面总结 <https://my.oschina.net/RabbitXiao/blog/968233>

2. 前端和后端的数据交互(jquery ajax+python flask+mysql)

https://blog.csdn.net/github_33873969/article/details/78723592

3. 前后端数据交互方法 <https://github.com/nimojs/blog/issues/13>

- 【get方式】使用jquery的getjson与后台交互

- 【万能方式】使用jquery的ajax与后台交互，设置不同的参数，可以get也可以post

4. Web前端与后端交互 <https://blog.csdn.net/qq1175421841/article/details/49895425>

5. H5 / JS发游戏实现案例：可作前端参考，但缺少与后端数据交互的部分。

<https://segmentfault.com/a/1190000008853259>

重要！没有之一！20190811note01 前后端及服务器基本认知模型 -----

知乎回答：了解网站开发结构

<https://www.zhihu.com/question/22689579/answer/48912898>



Part 1: 前端及通讯

- 前端：HTML, CSS, Javascript (JS), 脚本语言 [jQuery](#)框架
- 浏览器：翻译载体
- 前后端交互-数据传递与存储：XML / Json
- 页面自动刷新：[AJAX\(Asynchronous Javascript And XML\)](#)
- 文档对象模型（Document Object Model, DOM）:Window对象，history对象，link对象
- 通讯协议**：【前端-后台】交流/发送请求（HTTP:超文本传输协议）
- 域名与域名服务器Domain Name Server (DNS)：IP地址定位网站及服务器
- URL：Uniform Resource Locator - 域名+页面代号

Part 2: 网站后台

- **反向代理服务器 (web server)**：连接前后端，现成工具如[Nginx/apache](#)
- **通讯接口/网关接口 cgi / fastcgi**：反向代理服务器 与 前后端交流协议
- 平台/引擎/运行环境**: 常用的/被封装的模块与库: [node.js](#), / [.net](#) 可直接安装调用
- 框架**：服务器后台框架，如：JAVA的SSH/ [jQuery](#)；Python的[django / Web.py](#)等
（平台如零件，框架如盒子。平台与何种语言无关；框架则针对某种语言而开发的。框架+个人代码=网站个性化实现。）
- 服务器后台程序**：开发语言如PHP / Python
（红字部分的选择和连接，需要支持统一的通讯协议，如CGI；或增加翻译插件。如Python支持WSGI，可用FLUP插件与web server的CGI格式互换/通讯
- 开发工具 (Editor)**: IDE(Integrated Development Environment,集成开发环境)：如 VS / Eclipse
- 数据库与结构化查询语言：MySQL & SQL
（也就是无论使用c/java还是python,只要使用"select"这一SQL语言保留字,就可以从指定的数据库中筛选数据 --- 太厉害了！！）
- 服务器设备**：本质依然是一台计算机，不过是安装了相关程序/服务器端框架/接入网络能被访问/网络并发连接数支持/稳定安全性.....安装了web server程序的就是web server服务器，安装了Oracle的就是数据库服务器，安装了应用软件的就是应用服务器（0.0Soga）

了解后暂时不会使用的：

网站三剑客 Dreamweaver + Firework + Flash

<https://baike.baidu.com/item/%E7%BD%91%E9%A1%B5%E4%B8%89%E5%89%91%E5%AE%A2>

Dreamweaver：它与Flash、Fireworks并称网页三剑客。Dreamweaver是集网页制作和管理网站于一身的所见即所得网页编辑器，它是第一套针对专业网页设计师特别开发的视觉化网页开发工具，利用它可以轻而易举地制作出充满动感的网页。

DW百度百科

<https://baike.baidu.com/item/Adobe%20Dreamweaver/10015800?fromtitle=DREAMWEAVER&fromid=216984&fr=aladdin>

DW适合初学者-百度介绍 <https://jingyan.baidu.com/article/8ebacdf060f0fc49f75cd564.html>

其他名词解释

DOM文档对象模型 <https://baike.baidu.com/item/DOM/50288?fr=aladdin>

传输控制协议（TCP，Transmission Control Protocol）是一种面向连接的、可靠的、基于字节流的传输层通信协议

CDN - Content Delivery Network 内容分发网络

Active Server Pages（ASP）

FOSS(免费或者开源软件/Free or Open Source Software)

ECMAScript: 往往被称为JavaScript或JScript，所以它可以理解为是JavaScript的一个标准,但实际后两者是ECMA-262标准的实现和扩展。

JSON(JavaScript Object Notation, JS 对象简谱) 是一种轻量级的数据交换格式。它基于 **ECMAScript** (欧洲计算机协会制定的js规范)的一个子集，采用完全独立于编程语言的文本格式来存储和表示数据。简洁和清晰的层次结构使得 JSON 成为理想的数据交换语言。易于人阅读和编写，同时也易于机器解析和生成，并有效地提升网络传输效率。

CSS(Cascading Style Sheets): a stylesheet language for describing the presentation of a document written in HTML or XML.

CSS: 层叠样式列表，可基于HTML进行排版、颜色、字体等变化，并与脚本语言相衔接。

<https://baike.baidu.com/item/CSS/5457?fr=aladdin>