

**PHP开发**

目录

[目录 2](#_Toc492713843)

[第1章 服务器 4](#_Toc492713844)

[1. 服务器类型 4](#_Toc492713845)

[2. 服务器软件 4](#_Toc492713846)

[3. Web服务器 5](#_Toc492713847)

[第2章 客户端 5](#_Toc492713848)

[第3章 网络基础 5](#_Toc492713849)

[3.1 IP地址 5](#_Toc492713850)

[3.2 域名 6](#_Toc492713851)

[3.3 DNS服务 6](#_Toc492713852)

[3.4 端口 6](#_Toc492713853)

[第4章 C/S和B/S 6](#_Toc492713854)

[4.1 C/S结构 6](#_Toc492713855)

[4.2 B/S结构 7](#_Toc492713856)

[第5章 搭建Web服务 7](#_Toc492713857)

[5.1 安装Apache 8](#_Toc492713858)

[5.2 管理Web服务 8](#_Toc492713859)

[5.3 配置根目录 8](#_Toc492713860)

[5.4 网站部署 9](#_Toc492713861)

[5.5 配置虚拟主机 10](#_Toc492713862)

[第6章 PHP基础 11](#_Toc492713863)

[6.1 入门 11](#_Toc492713864)

[6.2 变量 11](#_Toc492713865)

[6.3 数据类型 12](#_Toc492713866)

[6.4 内容输出 12](#_Toc492713867)

[6.5 运算符 12](#_Toc492713868)

[6.6 函数 12](#_Toc492713869)

[6.7 分支、循环语句 13](#_Toc492713870)

[6.8 表单处理 13](#_Toc492713871)

[6.9 常用PHP函数 13](#_Toc492713872)

[6.10 应用实例 14](#_Toc492713873)

# 服务器

通俗的讲，能够提供某种服务的机器（计算机）称为服务器。

## 服务器类型

按照不同的划分标准，服务可划分为以下类型：

1、按服务类型可分为：文件服务器、数据库服务器、邮件服务器、Web 服务器等；

2、按操作系统可分为：Linux服务器、Windows服务器等；

3、按应用软件可分为 Apache服务器、Nginx 服务器、IIS服务器、Tomcat服务器、Node服务器等。

## 服务器软件

使计算机具备提供某种服务能力的应用软件，称为服务器软件，通过安装相应的服务软件，然后进行配置后就可以使计算具备了提供某种服务的能力。

常见的服务器软件有：

1、文件服务器：Server-U、FileZilla、VsFTP等；

2、数据库服务器：Oracle、MySQL、PostgreSQL、MSSQL等；

3、邮件服务器：Postfix、Sendmail等；

4、Web 服务器：Apache、Nginx、IIS、Tomcat、NodeJS等；

## Web服务器

即网站服务器，主要提供文档(文本、图片、视频、音频)浏览服务，一般安装Apache、Nginx服务器软件。

HTTP服务器可以结合某一编程语言处理业务逻辑，由此进行的开发，通常称之为服务端开发。

常见的运行在服务端的编程语言包括 PHP、Jsp、Asp、Python、Ruby、Perl等。

# 客户端

具有向服务器索取服务能力的终端，如比如 手机、电脑等，通过安装不同的客户端软件，可以获取不同的服务，比如通过QQ获得即时通讯服务、通过迅雷获得下载服务等。

常见的客户端软件：浏览器、QQ、迅雷、Foxmail等。

以浏览器为主（宿主环境），结合 HTML、CSS、Javascript等技术，而进行的一系列开发，通常称之为前端开发。

# 网络基础

## IP地址

所谓IP地址就是给每个连接在互联网上的主机(计算机)分配的一个32位地址。(就像每部手机能正常通话需要一个号码一样)

查看本机IP地址ipconfig(windows)、ifconfig(linux)

## 域名

由于IP地址基于数字，不方便记忆，于是便用域名来代替IP地址，域名是一个IP地址的“面具”。

查看域名对应的IP地址 ping、tracert

## DNS服务

DNS记录了 IP 地址和域名的映射（对应）关系；

查找优先级 本机hosts文件、DNS服务器

## 端口

端口号是计算机与外界通讯交流的出口，每个端口对应不同的服务。

现实生活中，银行不同的窗口办理不同的业务。

查看端口占用情况 netstat -an

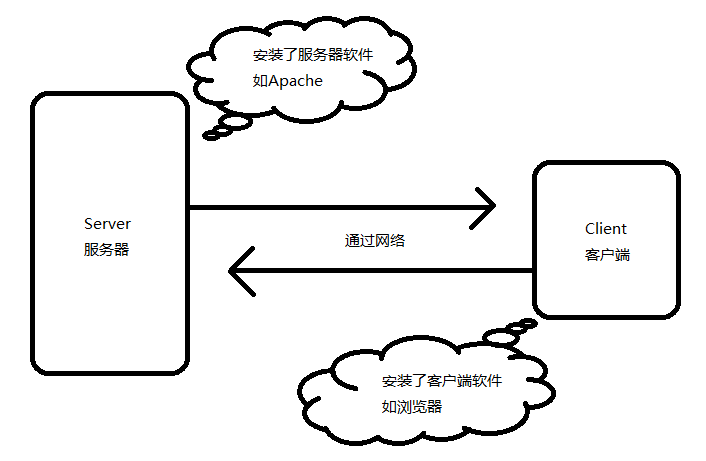
常见端口号 80、8080、3306、21、22

# C/S和B/S

## C/S结构

即Client、Server

C/S工作流程图



在C/S结构的情况下，不同的服务需要安装不同的客户端软件，比如QQ、迅雷、Foxmail这种情况下安装的软件会越来越多，同时也有许多弊端，比如A出差，需要在B电脑上查收邮件，但是B电脑并未安装Foxmail等类似的客户端软件，这样不得不先去下载Foxmail，非常不方便。

## B/S结构

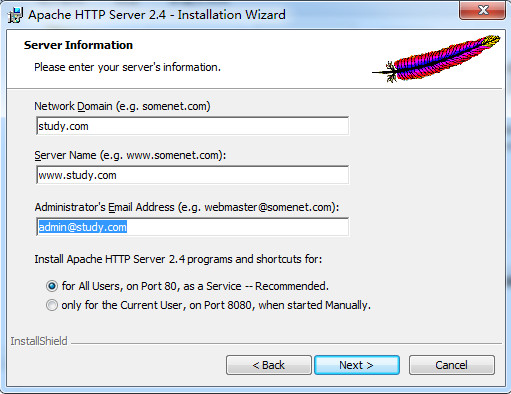
B/S（即Broswer、Server）解决了C/S所带来的不便，将所有的服务都可以通过浏览器来完成（因为基本所有浏览器都安装了浏览器），但B/S也有一些不利，比如操作稳定性、流畅度等方面相对较弱。

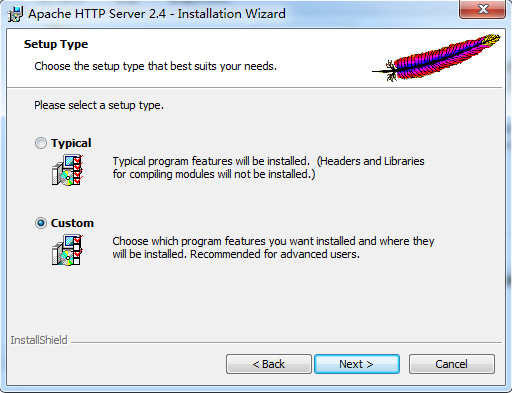
# 搭建Web服务

Windows (Linux) + Apache + Mysql + PHP，首字母组合。

## 安装Apache

安装apache，和普通软件安装无差别，除指定安装路径外，其它默认安装。

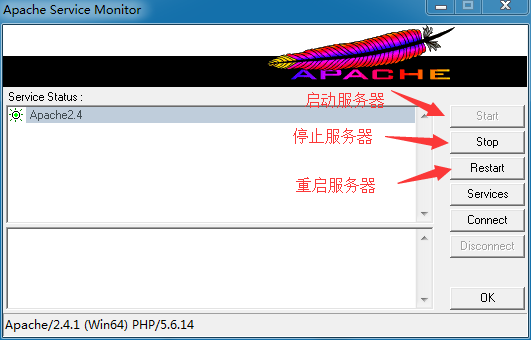




## 管理Web服务

任务图标绿色为正常启动状态

通过图形控制台可以启动、重启、停止所有服务



**注意事项：**

1、确保端口没有被其它程序占用 netstat -ano

2、“#”表示注释

3、修改配置要格外小心，防止止不小心修改其它内容

## 配置根目录

网站根目录是Web服务器上存放网站程序的空间，可通过修改配置文件自定义，如E:/www

具体步骤如下

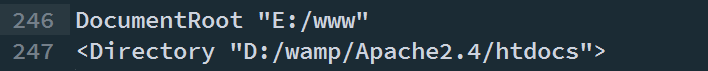
1、打开配置文件

如：D:\wamp\Apache2.4\conf（不同的的安装目录，配置文件位置不同）

2、设定新根目录，查找并修改

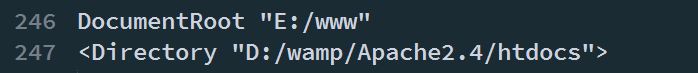


例如：

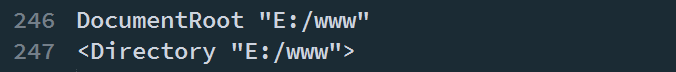


这样就指定了 "E:/www/"为存放网站的根目录。

3、配置根目录权限，查找<Dirctory ...>

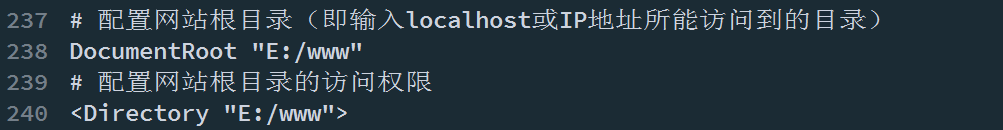


修改成



4、修改完后，并不能立即生效，需要重启Apache

注：可以指定任意目录为根目录



## 网站部署

将我们制作好的网页拷贝到配置好的根目录下，浏览器访问127.0.0.1或localhost即可。

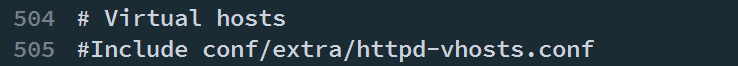
## 虚拟主机

在一台Web服务器上，我们可以通过配置虚拟主机，然后分别设定根目录，实现对多个网站的管理。

Apache 在启动时不会主动加载输配置，需要在主配置中声明。

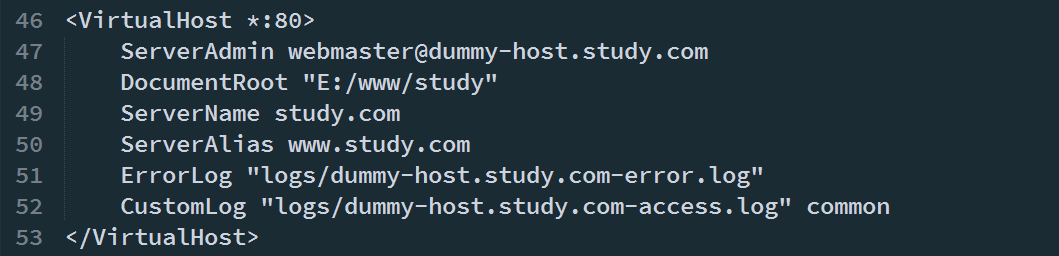
具体步骤如下：

1、开启虚拟主机辅配置，在httpd.conf 中找到

****

去掉前面的#号注释，开启虚拟主机配置

2、配置虚拟主机，打开conf/extra/httpd-vhosts.conf



分别修改以下三项

DocumentRoot "E:/www/example"

ServerName example.com

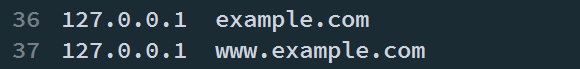
ServerAlias www.example.com

其它项无需指定。

3、修改DNS（hosts）文件

打开C:\Windows\System32\drivers\etc\hosts

目录是固定的



注：修改hosts文件权限

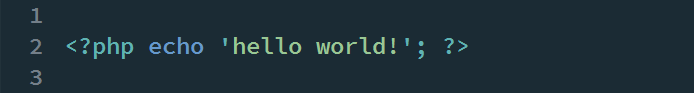
4、重启Apache

5、浏览器访问www.example.com

# PHP基础

## 入门

文件以.php后缀结尾，所有程序包含在<?php /\*\* 这里是代码 \*\*/ ?>



**注：避免使用中文目录和中文文件名**

见代码示例hello.php

## 变量

1、变量以$开头 字母/数字/下划线 不能以数字开头

2、大小写敏感（区分大小写）

见代码示例6-2.php

## 数据类型

字符型、整型、浮点型、布尔型、数组、对象、NULL、资源

单引号&双引号区别

索引数组、关联数组（了解即可）

见代码示例6-3.php

## 内容输出

echo：输出简单数据类型，如字符串、数值

print\_r()：输出复杂数据类型，如数组

var\_dump()：输出详细信息，如对象、数组（了解）

见代码示例6-4.php

## 运算符

基本与Javascript语法一致

. 号表示字符串拼接符，Javascript中为+号

见代码示例6-5.php

## 函数

与Javascript基本一致

函数名对大小写不敏感

见代码示例6-6.php

## 流程控制

与Javascript基本一致

数组遍历foreach()

见代码示例6-7.php

## 表单

HTML表单属性

|  |  |  |
| --- | --- | --- |
| 属性 | 示例 | 含义 |
| action | <form action="xx.php"> | 后端处理程序地址 |
| method | <form method="post/get"> | 请求方式 |

表单name属性的是用来提供给服务端接收所传递数据而设置的

表单action属性设置接收数据的处理程序

表单method属性设置发送数据的方式

当上传文件是需要设置 enctype="multipart/form-data"，且只能post方式 $\_GET接收 get 传值

$\_POST接收 post 传值

$\_FILES接收文件上传

## 应用实例

1、用户登录

2、动态网站

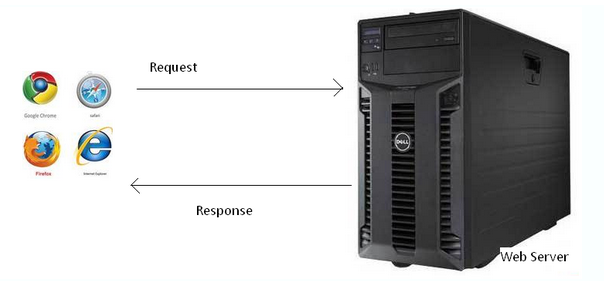
# HTTP协议

指服务器和客户端间进行通信时的约束和规范，客户端与服务端的数据交互并不是杂乱无章的，需要遵照一定的规范进行。

HTTP协议即超文本传输协议，网站的图片、CSS、JS等都是基于HTTP协议进行传输的。

HTTP协议对客户端到服务器的请求(Request)和服务器到客户机的响应(Response)进行了约束和规范。

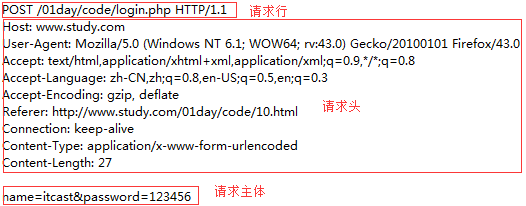
即HTTP协议主要由请求和响应构成。



常用请求方法 POST、GET、PUT、DELETE

### 请求/请求报文

请求由客户端发起，其规范格式为：请求行、请求头、请求主体。



1、请求行



由请求方式、请求URL和协议版本号构成

2、请求头

Host：localhost请求的主机

Cache-Control：max-age=0控制缓存

Accept：\*/\* 接受的文档MIME类型

User-Agent：很重要

Referer：从哪个URL跳转过来的

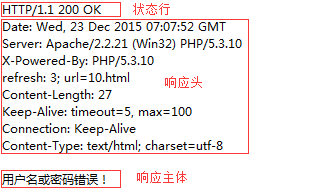
Accept-Encoding：可接受的压缩格式

3、请求主体

即传递给服务端的数据

### 响应/响应报文

响应由服务器发出，其规范格式为：状态行、响应头、响应主体。



1、状态行



由协议版本号、状态码和状态信息构成

2、响应头

Date：响应时间

Server：服务器信息

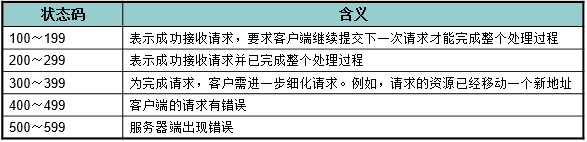
Content-Length：响应主体长度

Content-Type：响应资源的MIME类型

3、响应主体

即服务端返回给客户端的内容；

状态码



常见的有200代表成功、304文档未修改、403没有权限、404未找到、500服务器错误

### 调试工具

利用HTTP抓包工具在开发中可以帮我们进行调试

浏览器插件

Firebug、HttpWatch、chrome devTools

代理软件

Charles、Fiddler

### 应用

1、MIME类型&文件编码

2、缓存

3、重定向

4、刷新

5、下载