

**PHP开发**

目录

[目录 2](#_Toc493227129)

[第1章 服务器 5](#_Toc493227130)

[1. 服务器类型 5](#_Toc493227131)

[2. 服务器软件 5](#_Toc493227132)

[3. Web服务器 6](#_Toc493227133)

[第2章 客户端 6](#_Toc493227134)

[第3章 网络基础 6](#_Toc493227135)

[3.1 IP地址 6](#_Toc493227136)

[3.2 域名 7](#_Toc493227137)

[3.3 DNS服务 7](#_Toc493227138)

[3.4 端口 7](#_Toc493227139)

[第4章 C/S和B/S 7](#_Toc493227140)

[4.1 C/S结构 7](#_Toc493227141)

[4.2 B/S结构 8](#_Toc493227142)

[第5章 搭建Web服务 8](#_Toc493227143)

[5.1 安装Apache 9](#_Toc493227144)

[5.2 管理Web服务 10](#_Toc493227145)

[5.3 配置根目录 10](#_Toc493227146)

[5.4 网站部署 11](#_Toc493227147)

[5.5 虚拟主机 12](#_Toc493227148)

[第6章 PHP基础 13](#_Toc493227149)

[6.1 入门 13](#_Toc493227150)

[6.2 变量 13](#_Toc493227151)

[6.3 数据类型 13](#_Toc493227152)

[6.4 内容输出 14](#_Toc493227153)

[6.5 运算符 14](#_Toc493227154)

[6.6 函数 14](#_Toc493227155)

[6.7 流程控制 14](#_Toc493227156)

[6.8 表单 15](#_Toc493227157)

[6.9 应用实例 15](#_Toc493227158)

[第7章 HTTP协议 15](#_Toc493227159)

[7.1.1 请求/请求报文 16](#_Toc493227160)

[7.1.2 响应/响应报文 17](#_Toc493227161)

[7.1.3 调试工具 18](#_Toc493227162)

[7.1.4 应用 18](#_Toc493227163)

[第8章 PHP语言特性 21](#_Toc493227164)

[8.1.1 变量作用域 21](#_Toc493227165)

[8.1.2 常量 23](#_Toc493227166)

[8.1.3 文件包含 24](#_Toc493227167)

[8.1.4 字符串 26](#_Toc493227168)

[8.1.5 数组 27](#_Toc493227169)

[8.1.6 时间 28](#_Toc493227170)

[8.1.7 数学 29](#_Toc493227171)

[8.1.8 正则 29](#_Toc493227172)

[8.1.9 文件 29](#_Toc493227173)

[第9章 HTTP会话 31](#_Toc493227174)

[9.1.1 cookie 31](#_Toc493227175)

[9.1.2 session 32](#_Toc493227176)

# 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

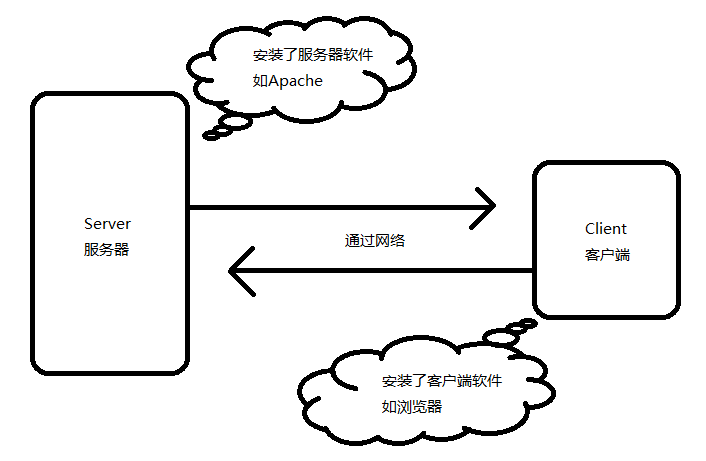
常见端口号 80、8080、3306、21、22

# 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

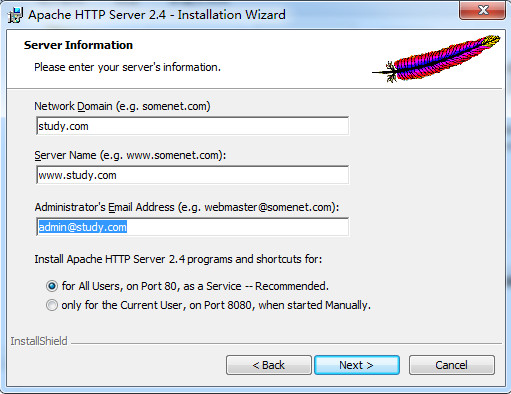
B/S（即Broswer、Server）解决了C/S所带来的不便，将所有的服务都可以通过浏览器来完成（因为基本所有浏览器都安装了浏览器），但B/S也有一些不利，比如操作稳定性、流畅度等方面相对较弱。

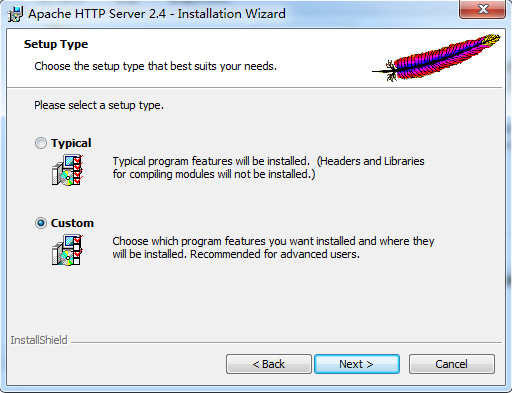
# 搭建Web服务

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。

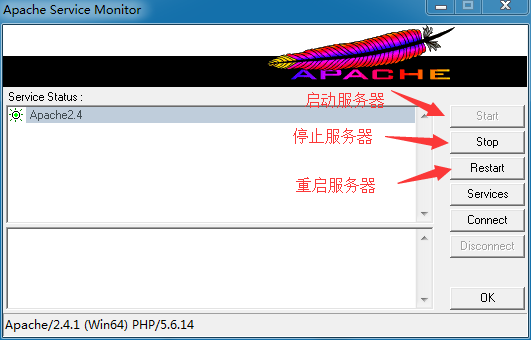




## 管理Web服务

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

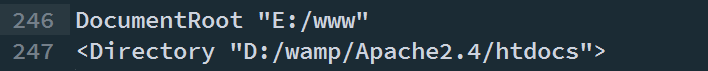
1、打开配置文件

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

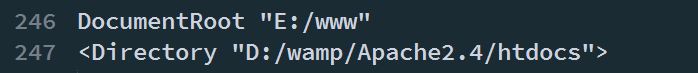


例如：

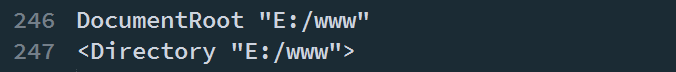


这样就指定了 "E:/www/"为存放网站的根目录。

3、配置根目录权限，查找<Dirctory ...>

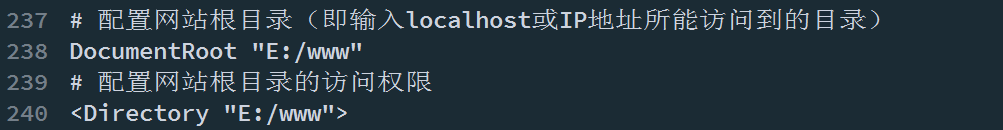


修改成



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。

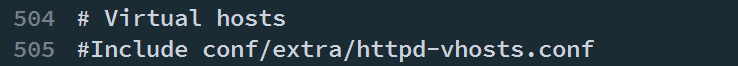
## 虚拟主机

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。

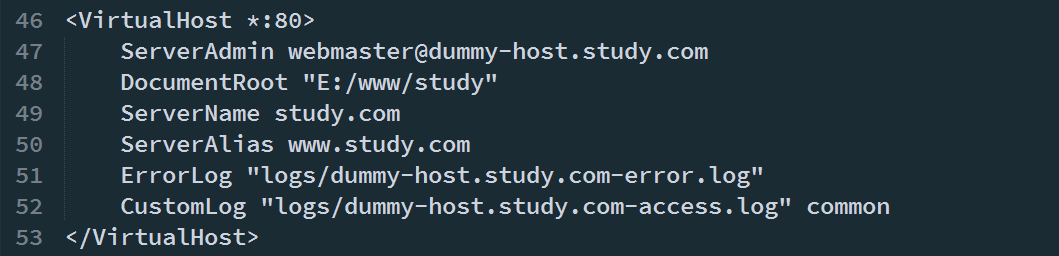
具体步骤如下：

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

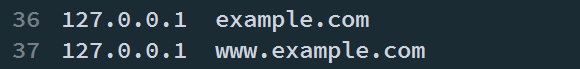
ServerAlias www.example.com

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

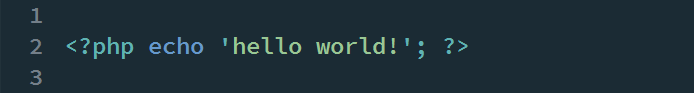
4、重启Apache

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

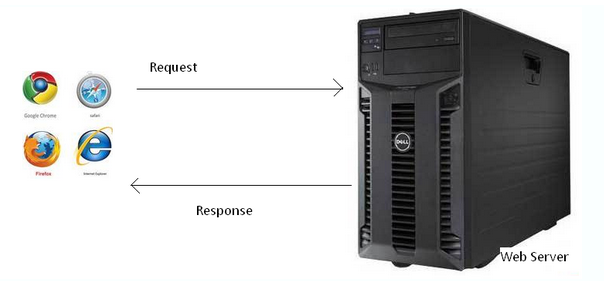
# HTTP协议

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。

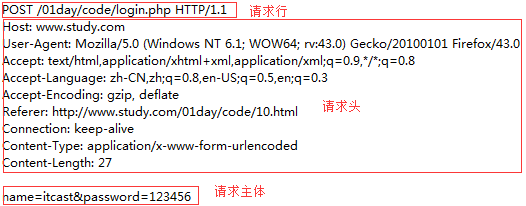
即HTTP协议主要由请求和响应构成。



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

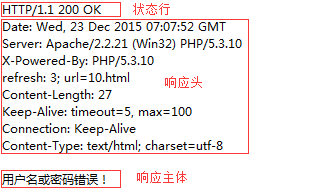
Accept-Encoding：可接受的压缩格式

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

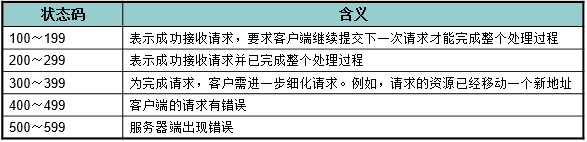
Content-Length：响应主体长度

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

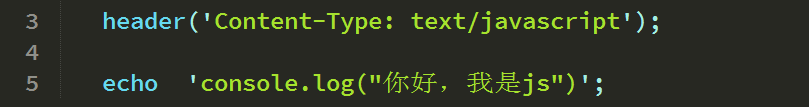
Charles、Fiddler

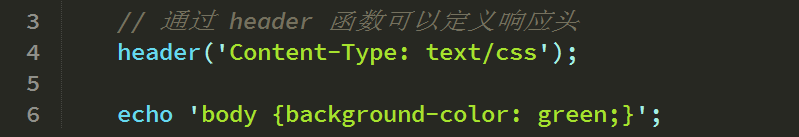
### 应用

1、MIME类型&文件编码

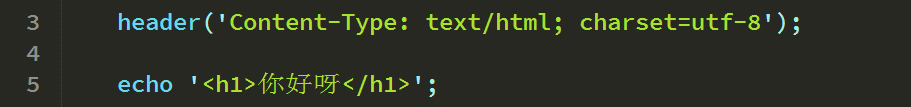
浏览器发起请求的方式是多样的，当发起请求后服务端会有对应的内容响应过来，浏览器会根据响应头Content-Type来对响应的内容进行解析。

在PHP中通过header函数可以对响应头进行设置，如下图





可以同时指定编码集



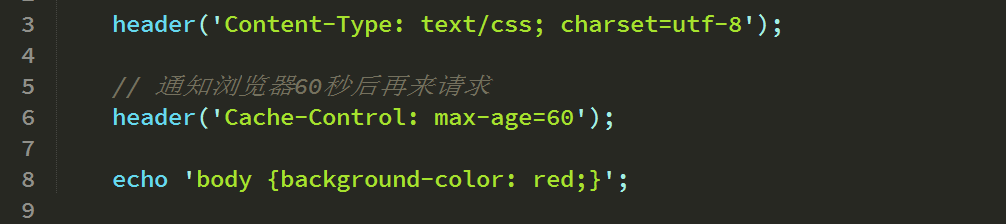
注：通过对Content-Type的学习，可以理解每次请求，只需关心其返回的内容，以及浏览器应如何解析，而不应关注请求地址的后缀（有的请求地址甚至没有后缀）。

2、缓存

为了提升访问网页的速度，服务器并不是每次都会响应内容给浏览器，一般情况下浏览器发起第一次请求后服务器响应内容给浏览器，浏览器会将响应来的内容缓存起来，当下将再请求时如果响应内容没有改变那么浏览器会直接读取缓存的内容。

通上述分析得之缓存可以提升网页加载速度，但是并不是最完美的，因为响应内容没有改变时，浏览器仍要发送一次请求，只是这次请求没有响应内容，即然后没有响应结果，那么这次请求其实是没有意义的，如果连这次请求都不发送会更大程度扣升性能。

通过PHP设置响应头可以实现这一点



3、重定向

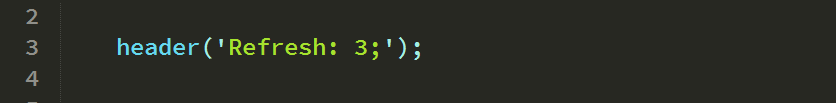
使用PHP的header设置 Location 响应头可以将请求指向另外一个地址，将这种行为称为重定向。



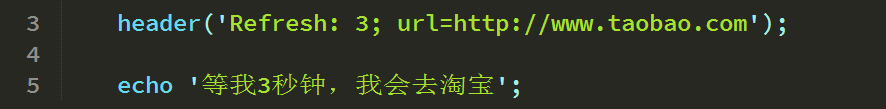
重定向到 http://www.baidu.com 注：其状态码为302

4、刷新

使用PHP的header设置Refresh响应头可以对当前页面进行刷新，也可以刷新其它页面。



上图，每隔3秒，刷新一次当前页

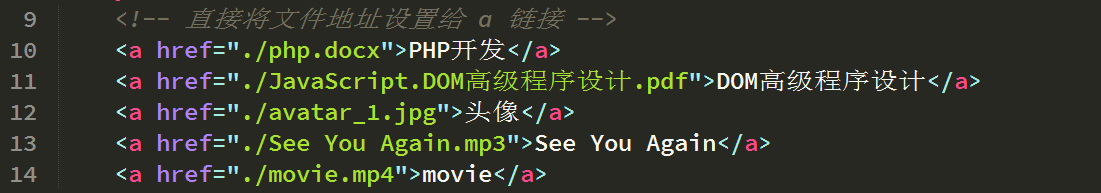


上图，每隔3秒，刷新一次淘宝页面

注：其状态码为200

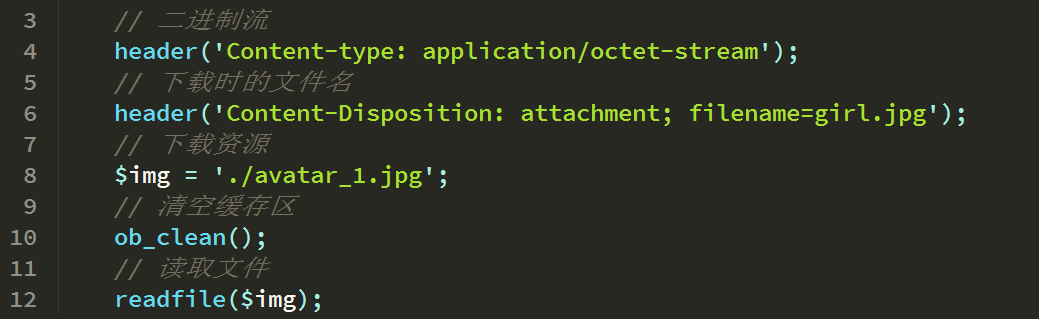
5、下载

通过网页是可以下载文件的。



将文件地址赋值给a链接，服务器会自动设置相应的Content-Type告知浏览器应如何解析，如果浏览器没有能力解析，那么将会自动下载。

如果需要专门提供下载而不是让浏览器进行解析，可以通过修改Content-Type和Content-Disposition实现。

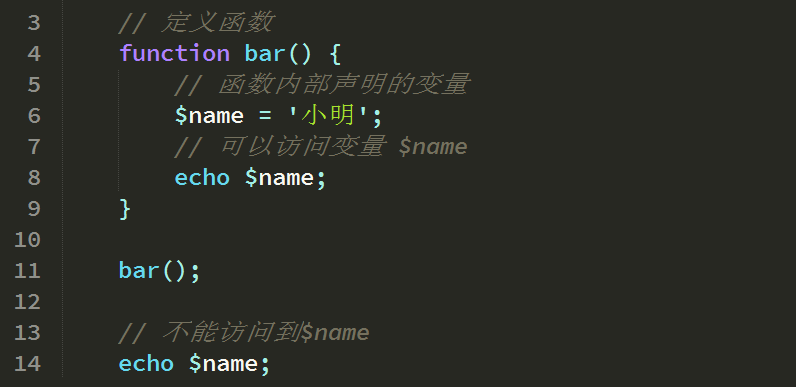


# PHP语言特性

### 变量作用域

在 PHP 中，可以在脚本的任意位置对变量进行声明。变量的作用域指的是变量能够被引用/使用的那部分脚本。

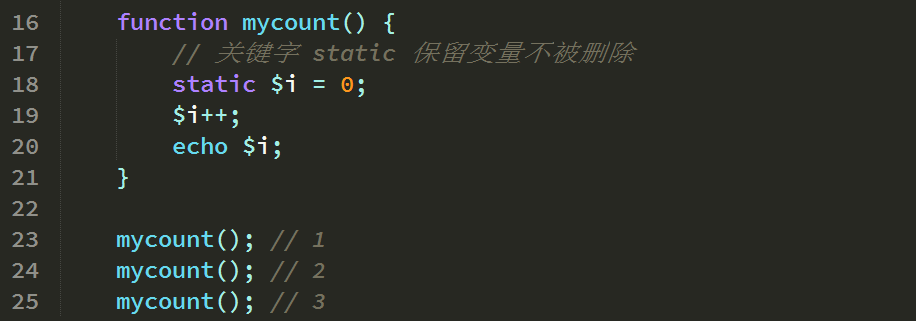
1、函数内部声明的变量拥有local（局部）作用域，只能在当前函数内部进行访问。



2、函数之外声明的变量拥有global（全局）作用域，只能在函数以外进行访问。



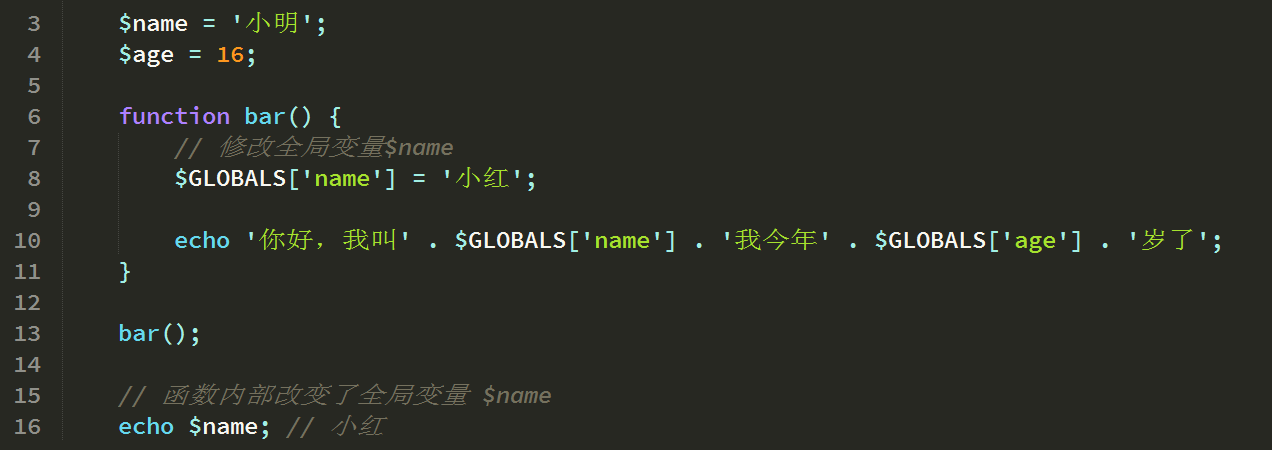
3、通常情况下，函数完成/执行后，会删除内部所有变量，通过关键字static 可以使得变量不被删除。



4、函数使用关键字gobal声明对全局变量的引用，可以访问和修改全局变量。



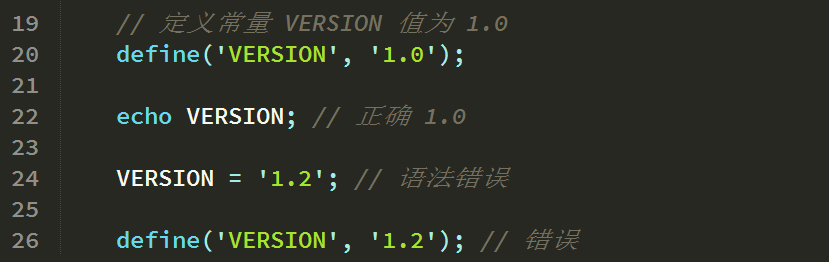
5、函数内部通过超全局变量$\_GLOBALS也可以访问到全局变量。



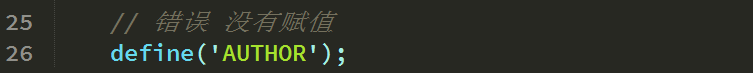
### 常量

常量与变量类似，区别在于常量一旦声明便无法被修改，定义常量不需要“$”符号。

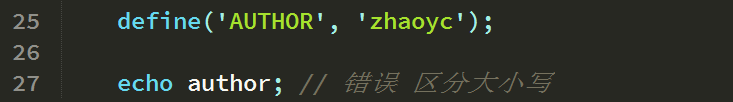
在PHP中使用define定义常量。



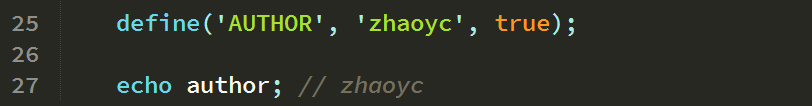
定义常必须要赋值



默认情况下常量区分大小写，可以通过第3个参数调整，默认值为false



第3个参数为true，不区分大小写

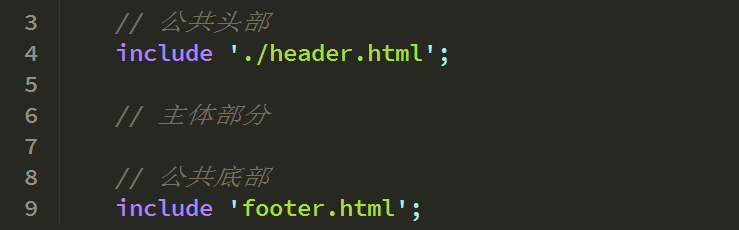


注：常量常用于定义网站配置信息

### 文件包含

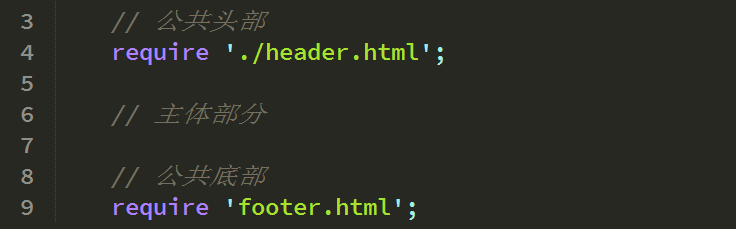
通过文件包含可以将其它文件内容插入当前执行脚本中，可以实现功能逻辑的复用。

1、include和include\_once



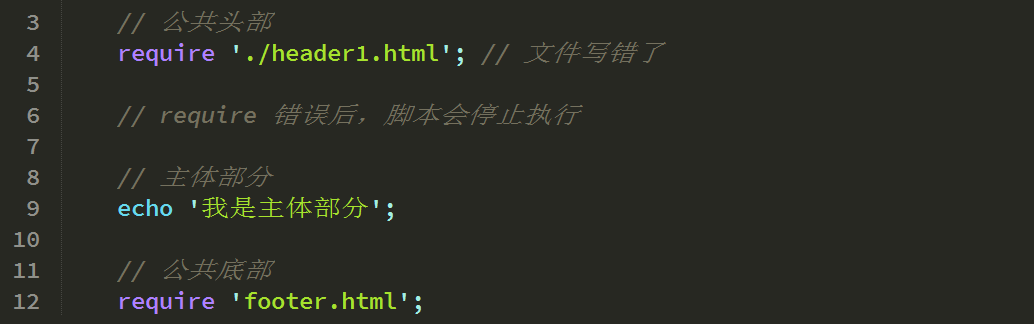
include\_once 和 include 的区别是include\_once相同文件只包含一次。

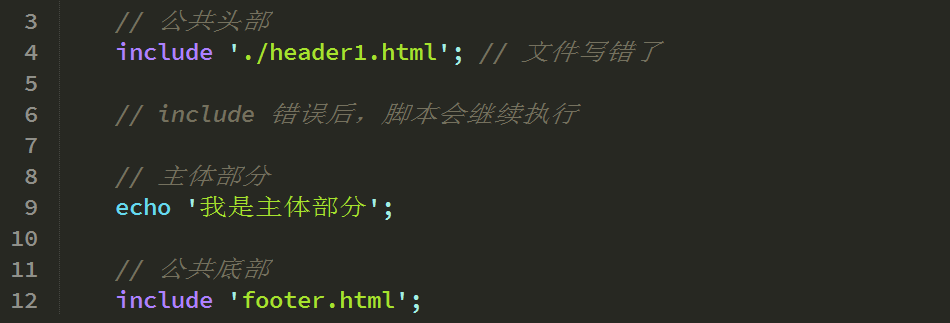
2、require和require\_once



require\_once和require的区别是require\_once相同文件只包含一次。

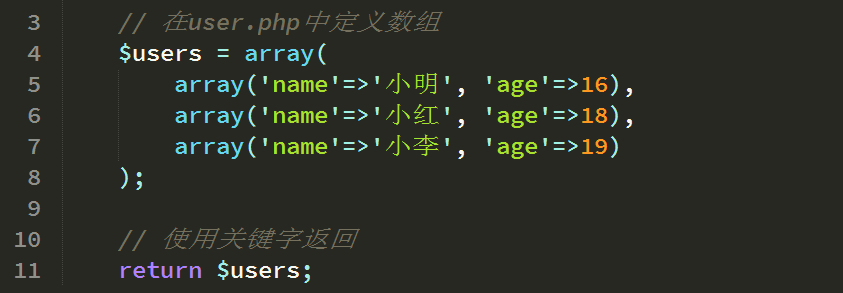
require和include都可以用来包含外部文件，其区别在于：

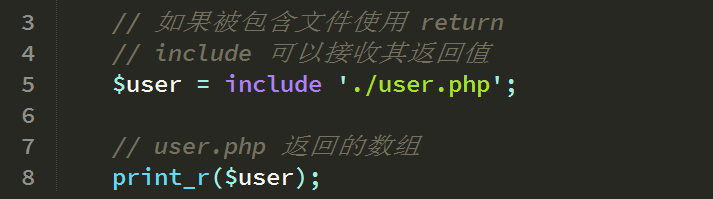




当require包含文件出现错语时，脚本会停止解析，而include只是错误提示，脚本会继续执行。

3、如果被包含文件使用 return 关键字，include可以接收其返回值。





### 字符串

strlen() 获取字符串长度

mb\_strlen()

substr() 返回字符串的子串

mb\_substr()

strtoupper() 将字符串转化为大写

strtolower() 将字符串转化为小写

ucfirst() 将字符串的首字母转换为大写

[ucwords()](http://www.w3school.com.cn/php/func_string_ucwords.asp) 把字符串中每个单词的首字符转换为大写

trim() 去除字符串首尾处的空白字符

ltrim() 删除字符串开头的空白字符

rtrim() 删除字符串末端的空白字符

strrev() 反转字符串

strpos() 查找字符串首次出现的位置，从左往右查找

mb\_strpos()

strrpos() 计算指定字符串在目标字符串中最后一次出现的位置，从右往左查找

strchr()和strstr() 两者一样，用于查找字符串的首次出现

str\_replace() 子字符串替换

str\_repeat() 重复一个字符串

### 数组

1、数组指针（了解即可）

current() 返回当前数组指针处元素的值，不移动指针

key() 返回当前数组指针处元素的下标，不移动指针

next() 返回下一个数组元素的值，并将指针下移一行

prev() 返回上一个数组元素的值，并将指针回退一步

end() 返回最后一个数组元素的值，并将指针移到数组最后一个元素

reset() 返回第一个数组元素的值，并将指针移到第一个元素上

2、数组取值

list() 从索引数组中取值并赋值给变量

3、数据排序

sort() 对数组元素的值升序排列，不保留索引关系

rsort() 对数组元素的值降序排列，不保留索引关系

asort() 对数组元素的值升序排列，保留索引关系

arsort() 对数组元素的值降序排列，保留索引关系

ksort() 数组按键名(下标)升序排列，保留索引关系

krsort() 数组按键名降序排列，保留索引关系

注：都是对原数组进行操作

4、字符串数组互换

implode() 将一个一维数组的值转化为字符串(数组转字符串)

explode() 使用一个字符串分割另一个字符串(将字符串转成数组)

5、其它

[array\_keys()](http://www.w3school.com.cn/php/func_array_keys.asp) 用指定键名的给定键值填充数组

[array\_values()](http://www.w3school.com.cn/php/func_array_values.asp) 返回数组中所有的值

[array\_key\_exists()](http://www.w3school.com.cn/php/func_array_key_exists.asp) 检查指定的键名是否存在于数组中

[array\_unique()](http://www.w3school.com.cn/php/func_array_unique.asp) 删除数组中的重复值

[array\_pop()](http://www.w3school.com.cn/php/func_array_pop.asp)删除数组的最后一个元素（出栈）

[array\_push()](http://www.w3school.com.cn/php/func_array_pop.asp)将一个或多个元素插入数组的末尾

array\_shift()删除数组中首个元素

array\_unshift()在数组开头插入一个元素

### 时间

time() 返回自从 Unix 纪元（格林威治时间 1970 年 1 月 1 日 00:00:00）到当前时间的秒数

date() 格式化一个本地时间/日期

microtime() 返回当前 Unix 时间戳和微秒数

strtotime() 将任何英文文本的日期时间描述解析为 Unix 时间戳

### 数学

max() 求最大值

min() 求最小值

rand() 产生一个随机整数

ceil() 进一法取整

floor() 舍去法取整

round() 对浮点数进行四舍五入

pow() 次方运算

abs() 求绝对值

sqrt() 开方运算

### 正则

preg\_match 根据正则表达式查找内容

preg\_replace 根据正则表达式替换内容

preg\_split 根据正则表达式拆分内容

preg\_grep 根据正则表达式在数组中查找单元

### 文件

pathinfo() 文件信息

[basename()](http://www.w3school.com.cn/php/func_filesystem_basename.asp) 查找最后一级目录

[dirname()](http://www.w3school.com.cn/php/func_filesystem_dirname.asp) 查找父级目录

[filesize()](http://www.w3school.com.cn/php/func_filesystem_filesize.asp) 读取文件大小

[file\_exists()](http://www.w3school.com.cn/php/func_filesystem_file_exists.asp) 检查文件是否存在

[is\_dir()](http://www.w3school.com.cn/php/func_filesystem_is_dir.asp) 是否为目录

[is\_file()](http://www.w3school.com.cn/php/func_filesystem_is_file.asp) 是否为文件

realpath() 绝对路径

rename() 修改文件名

[file\_get\_contents()](http://www.w3school.com.cn/php/func_filesystem_file_get_contents.asp) 读取文件

[file\_put\_contents()](http://www.w3school.com.cn/php/func_filesystem_file_put_contents.asp) 写入文件

[move\_uploaded\_file()](http://www.w3school.com.cn/php/func_filesystem_move_uploaded_file.asp) 文件上传

copy() 复制文件

unlink() 删除文件

readfile() 读取文件，返回读取内容的字节数

fopen() 打开文件

fread() 读取文件

fwrite() 写文件

fgets() 逐行读取文件

fclose() 关闭文件

mkdir() 创建目录

rmdir() 删除空目录

getcwd() 查找当前目录

opendir 打开目录

readdir 读取目录

closedir 关闭目录

scandir 查找指定目录文件及子目录

disk\_free\_space() 磁盘剩余空间

disk\_total\_space() 磁盘总大小

# HTTP会话

[超文本传输协议](https://baike.baidu.com/item/%E8%B6%85%E6%96%87%E6%9C%AC%E4%BC%A0%E8%BE%93%E5%8D%8F%E8%AE%AE)（HTTP，HyperText Transfer Protocol）是[互联网](https://baike.baidu.com/item/%E4%BA%92%E8%81%94%E7%BD%91" \t "_blank)上应用最为广泛的一种[网络协议](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C%E5%8D%8F%E8%AE%AE)，具有简单灵活、无连接和无状态等特点。

1、无连接是指每次连接只处理一个请求，服务器处理完客户端的请求并收到客户的应答后，即断开连接。采用这种方式可以节省传输时间。

2、无状态是指协议对于事务处理没有记忆能力，这在现实应用中产生了不良有影响。

3、通过cookie和session可以弥补HTTP无状态的缺陷。

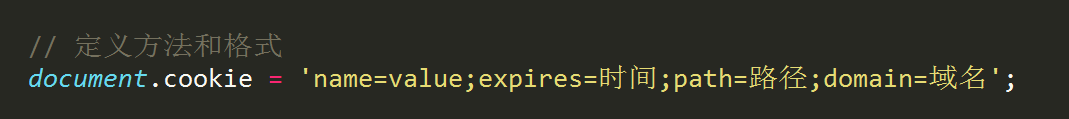
### cookie

储存在客户端（浏览器）上的数据，可以用来辨别用户身份和进行 session 跟踪，cookie会以请求头的方式发送至服务端。

在Javascript中通过document.cookie进行读取和设置

1、设置

一个cookie主要由name、value、expires、path、domain，5部分构成，其设置方法和格式如下图：



其中name和value必须进行设置，其它都有默认值：

expires表示cookie失效的时间，默认为关闭浏览器

path表示cookie生效的路径，默认为当路径

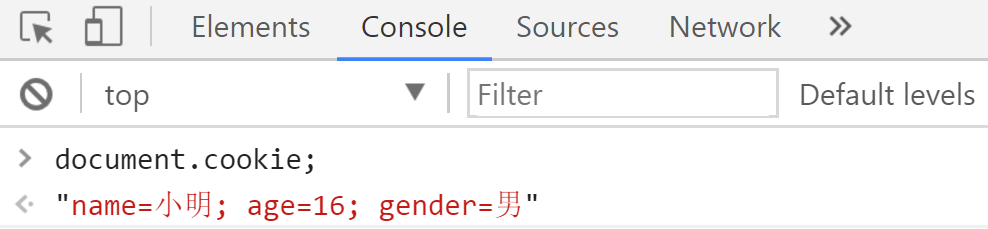
domain表示cookie生效的域名，默认为当前域名

2、读取

cookie是以固定格式字符串形式存在的，需要手工进行解析，通过以下方式进行读取。



通过Chrome调试工具可以查看存储的cookie



3、作用

a) 数据缓存，通过设置过期时间，可以长时间保存数据。

b) 数据共享，通过domain和path 可以跨页面传递数据

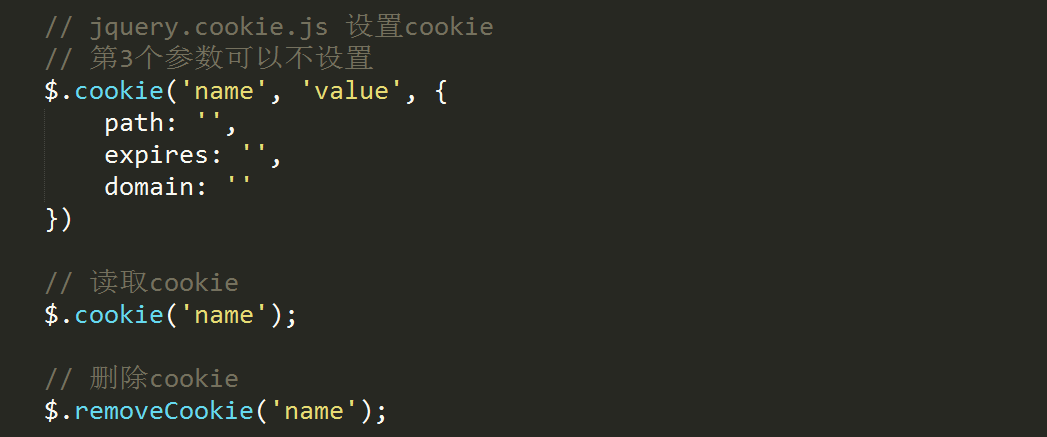
b) 服务端交互，最主要用途是浏览器会自动将cookie做为所有请求的头信息传递给服务器。

注：cookie是有大小限制的约4k

4、解析

由于cookie是一个字符串，当成数据在使用时不方便，需要利用字符串函数或正则表达进行处理。

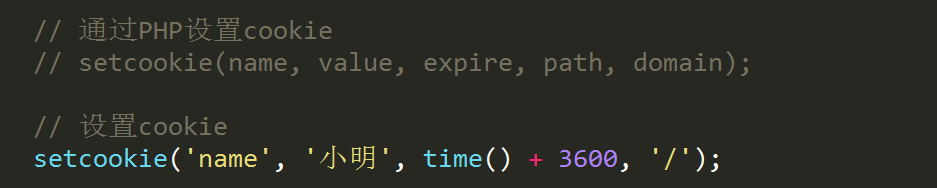
也可以使用jquery插件jquery.cookie.js



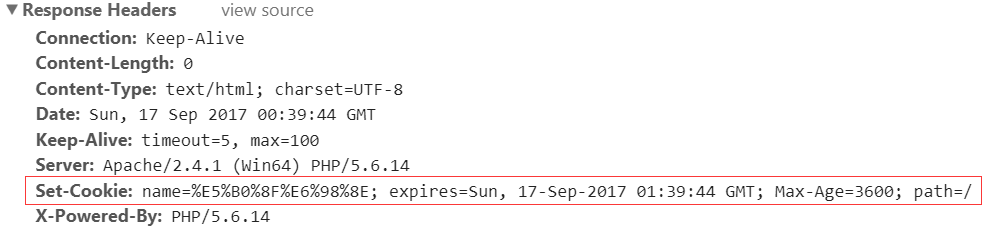
cookie被客户端以请求头的方式传递给了后端，后端是可以接收并处理cookie，在PHP中通过超全局数组 $\_COOKIE获取。

cookie是存储在客户端（浏览器）中的，是由客户端（浏览器）实现操作的，但是服务端也可间接的对cookie进行干预。

在PHP中当调用setcookie函数时，会自动设置一个响应头Set-Cookie，通知客户端（浏览器）设置一个cookie。



通过Chrome调试工具查看



### session

存储在服务端的数据，主要记录客户端的访问状态。

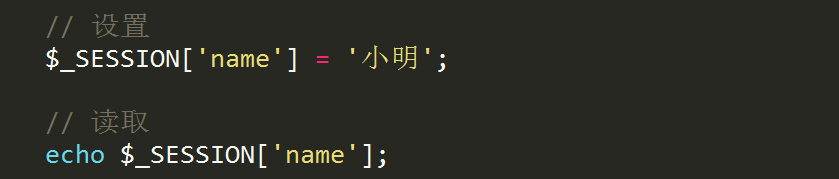
Session 的工作机制是：为每个访问者创建一个唯一的标识 (UID)，并基于这个标识来存储变量，一般标识存储在cookie 中。

以PHP为例，操作session

1、开启session



2、通过超全局数组 $\_SESSION操作session

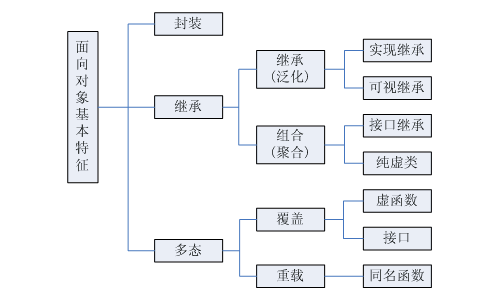


session不是独立使用的，通常需要配合cookie一起使用，当服务端设置一个cookie时会自动设置响应头Set-Cookie通知客户端（浏览器）设置一个名为（默认）PHPSESSID的cookie，客户端（浏览器）下次再发起请求时，会将这个cookie以请求头的方式回传至服务端，服务端再根据这个cookie获取相对应的session。

# PHP面向对象

面向[对象](https://baike.baidu.com/item/%E5%AF%B9%E8%B1%A1)（Object Oriented，OO）是[软件开发方法](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E5%BC%80%E5%8F%91%E6%96%B9%E6%B3%95" \t "_blank)，具有封装、继承、多态3个特征，与之对应的是面向过程。

[面向对象编程](https://baike.baidu.com/item/%E9%9D%A2%E5%90%91%E5%AF%B9%E8%B1%A1%E7%BC%96%E7%A8%8B)（Object Oriented Programming，OOP，[面向对象程序设计](https://baike.baidu.com/item/%E9%9D%A2%E5%90%91%E5%AF%B9%E8%B1%A1%E7%A8%8B%E5%BA%8F%E8%AE%BE%E8%AE%A1" \t "_blank)）是一种计算机编程架构。



1、封装

2、继承

3、多态

# 数据库软件

数据库是存储网站数据的一种方式，通过数据软件进行管理，常见的数据库软件有MySQL、Oracle、Sql Server等，MySQL由瑞典MySQL AB 公司开发，目前属于 [Oracle](http://baike.baidu.com/view/15020.htm) 旗下产品，MySQL 是最流行的[关系型数据库管理系统](http://baike.baidu.com/view/1450387.htm" \t "_blank)之一。

## 安装

MySQL也是一个跨平台的软件（支持多个操作系统），以Windows为例，MySQL是一个可执行程序，其安装方式与普通软件一样。

## 数据库

数据库（Database）是按照数据结构来组织、存储和管理数据的仓库，类似于EXCEL表格管理数据一样。





## 客户端

MySQL是一个C/S结构的软件系统，上述步骤安装了MySQL的服务器（即Server），通过客户端（Client）访问MySQL服务器。

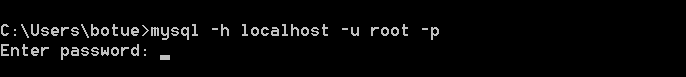
### CMD

1、登录

通过客户端访问数据库服务器，CMD是一个可以访问MySQL的客户端，使用CMD访问MySQL需要添加环境变量。

找到自已的MySQL的安装路径D:\wamp\bin\mysql\mysql5.5.20\bin，然后添加到环境变量中。

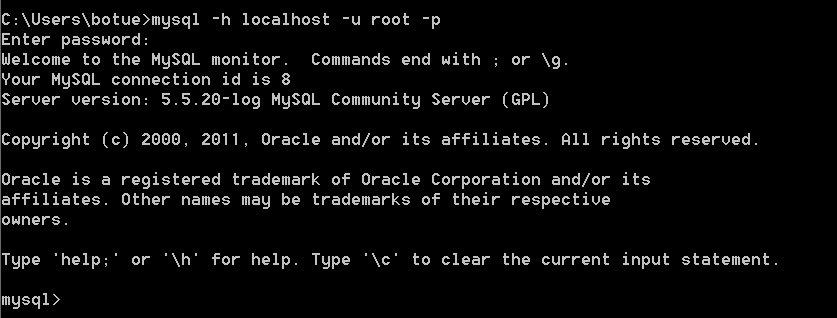
使用如下命令登录



-h 表示MySQL服务器地址

-u 表示用户名称 默认 root

-p 表单用户密码 默认 空



2、创建数据库



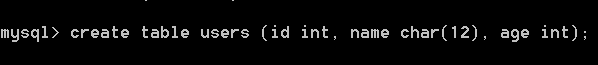
3、查看数据库



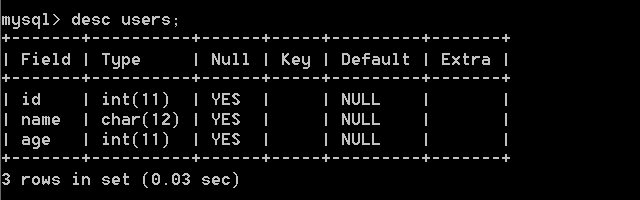
4、切换数据库



5、创建数据表



6、查看表结构



### 可视化工具

CMD做为客户端可以访问MySQL服务器，但工作效率非常低，实际开发中一般使用可视化的工具管理数据库。

可视工具种类非常多，推荐使用Navicat for MySQL，它是一个付费软件。

1、登录

登录界面操作

2、创建数据库

CREATE DATABASE 数据库名称

3、创建数据表

图形界面创建

4、插入数据

INSERT INTO 表名 (字段1, 字段2, 字段3...) VALUES(值1, 值2, 值3...);

例如INSERT INTO `users` (id, name, age, pass) VALUES(4, '赵六', 18, '8213121');

5、查询数据

SELECT 字段名1，字段名2，字段名3... FROM 表名;

例如SELECT `name`, `sex`, `email`, `pass` FROM users;

6、更新数据

UPDATE 表名 SET 字段名1='值1', 字段名2='值2', ....

例如 update users set pass='123865', name='小牛', age=20 where id=3

7、删除数据

DELETE FROM 表名 WHERE 条件

例如delete from users where age=20

8、筛选数据(带条件的查询)

SELECT 字段名1, 字段名2, 字段名3.. WHERE 条件 AND 条件 OR 条件

例如select id, name, age from users where age<18 and sex='boy' or age>18 and sex='girl'

9、排序数据（查找数据并将数据排序）

SELECT 字段名1, 字段名2.. FROM 表名 ORDER BY 排序条件 顺序

ASC 正序 DESC 倒序

例如select id, name, age from users order by age desc

10、分组数据（查找数据并分组进行统计操作）

SELECT 统计操作 FROM 表名 GROUP BY 条件

count统计 sum求合 avg 平均值

select avg(age) as total from users group by sex 分组求平均年龄

select sum(age) as total from users group by sex 分组求年龄和

select count(\*) as total from users group by sex 计算分组合每组个数

11、导出导入数据

数据库是个做迁移操作的，其中包含了导入和导出操作。

12、数据类型

每个字段对应的列的数据是要有特定类型的，比如 数值、字符、时间等。

[参见文档](http://www.runoob.com/mysql/mysql-data-types.html)，此内容了解即可。

13、主键

表中每一行都应该有可以唯一标识自己的一个字段，此字段称为主键（PRIMARY KEY），此字段值不允许重复，一般设为自增长（AUTO\_INCREMENT）