**WEB概念、WAMP安装、PHP语法基础**

目录

[一、 PHP简介 4](#_Toc495938187)

[1. 静态网站与动态网站 4](#_Toc495938188)

[2. PHP语言简介 5](#_Toc495938189)

[1) 发展历程 5](#_Toc495938190)

[2) PHP语言的功能特点 6](#_Toc495938191)

[二、 WEB开发基本概念及网站访问流程 7](#_Toc495938192)

[1. WEB服务器 7](#_Toc495938193)

[1) 常见的web服务器 7](#_Toc495938194)

[2) 常用的开发环境 7](#_Toc495938195)

[2. 软件架构 8](#_Toc495938196)

[1) C/S架构 8](#_Toc495938197)

[2) B/S架构 8](#_Toc495938198)

[3. IP地址 8](#_Toc495938199)

[1) IP地址概念 8](#_Toc495938200)

[2) 两种特殊的IP地址： 9](#_Toc495938201)

[4. 域名 9](#_Toc495938202)

[1) 域名与服务器名 9](#_Toc495938203)

[2) 特殊的域名 10](#_Toc495938204)

[3) 顶级域名： 10](#_Toc495938205)

[5. DNS 10](#_Toc495938206)

[1) DNS服务器 11](#_Toc495938207)

[2) 本地hosts文件 11](#_Toc495938208)

[6. 端口 12](#_Toc495938209)

[7. URL 12](#_Toc495938210)

[8. web程序的访问流程 13](#_Toc495938211)

[三、 使用PHPstudy安装WAMP环境 13](#_Toc495938212)

[1. 安装包介绍 13](#_Toc495938213)

[2. 一键安装WAMP环境 14](#_Toc495938214)

[1) 选择安装服务器程序安装路径 14](#_Toc495938215)

[2) 安装产生目录介绍 14](#_Toc495938216)

[3. 测试是否安装成功 15](#_Toc495938217)

[4. 设置服务器时区 16](#_Toc495938218)

[四、 虚拟主机 16](#_Toc495938219)

[1. 虚拟主机介绍 16](#_Toc495938220)

[2. 开启虚拟主机配置文件 17](#_Toc495938221)

[3. 配置基于域名的站点信息 18](#_Toc495938222)

[4. 解析虚拟主机 19](#_Toc495938223)

[5. 创建测试文件 19](#_Toc495938224)

[6. 设置列表显示脚本 20](#_Toc495938225)

[7. 配置www.test.com 20](#_Toc495938226)

[1) 添加站点 20](#_Toc495938227)

[2) 修改配置文件，列表形式显示脚本 20](#_Toc495938228)

[3) 解析 21](#_Toc495938229)

[五、 PHP基本语法 21](#_Toc495938230)

[1. PHP标记 21](#_Toc495938231)

[2. 语句结束标记 22](#_Toc495938232)

[3. 注释 23](#_Toc495938233)

[1) 单行注释 23](#_Toc495938234)

[2) 块注释 23](#_Toc495938235)

[4. 打造自己的PHP开发IDE 24](#_Toc495938236)

[六、 变量 24](#_Toc495938237)

[1. 变量介绍 24](#_Toc495938238)

[1) 定义 24](#_Toc495938239)

[2) 变量的组成 25](#_Toc495938240)

[3) 变量名的命名规则 25](#_Toc495938241)

[2. 变量操作 25](#_Toc495938242)

[1) 变量声明 25](#_Toc495938243)

[2) 读取变量的值 26](#_Toc495938244)

[3) 变量值的修改 26](#_Toc495938245)

[4) 删除变量 26](#_Toc495938246)

[3. 常见的输出语句 27](#_Toc495938247)

[1) echo语句 27](#_Toc495938248)

[2) 解决输出中文字符等特殊字符显示乱码问题 27](#_Toc495938249)

[3) print语句 29](#_Toc495938250)

[4) var\_dump() 29](#_Toc495938251)

[5) print\_r() 30](#_Toc495938252)

[七、 PHP数据类型 31](#_Toc495938253)

[1. PHP数据类型有哪些？ 31](#_Toc495938254)

[2. 标量（scalar,简单）数据类型 31](#_Toc495938255)

[3. 复合数据类型 32](#_Toc495938256)

[4. 特殊类型 32](#_Toc495938257)

[1) null 32](#_Toc495938258)

[2) resource 32](#_Toc495938259)

[5. 整型类型 33](#_Toc495938260)

[1) 整数的表示形式 33](#_Toc495938261)

[2) 各种进制 34](#_Toc495938262)

[3) 其他进制准换为十进制 35](#_Toc495938263)

[4) 十进制转换为其他进制X 35](#_Toc495938264)

[5) 进制转换函数 35](#_Toc495938265)

[6. 浮点型类型 36](#_Toc495938266)

[7. 布尔类型介绍 37](#_Toc495938267)

[1) isset（） 37](#_Toc495938268)

[2) empty() 38](#_Toc495938269)

**学习目标介绍：**

* 了解Web基本概念，掌握WEB访问流程
* 能够使用PHPstudy搭建WAMP开发环境
* 掌握虚拟主机的搭建
* 掌握基本的标记，语句结束标志等
* 掌握PHP的变量类型以及相关的变量管理等

# PHP简介

## 静态网站与动态网站

* 静态代码与静态网站

**Javascript**

处理网页行为

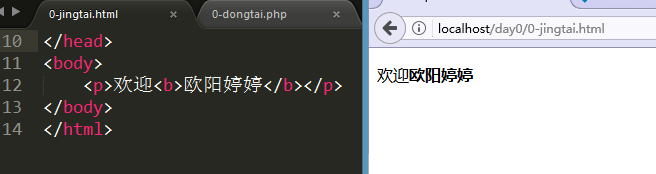
**CSS**

处理显示样式。

**HTML**

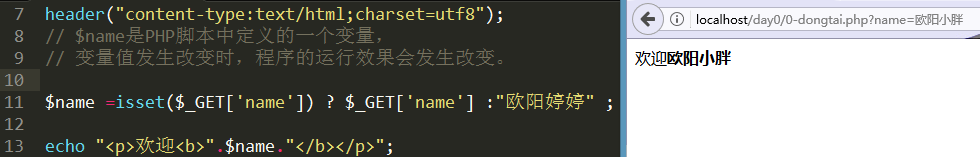
处理网页内容

静态网站：使用浏览器端语言进行编程，网站由静态代码（HTML.CSS,JS）组成。用户请求服务上已经存在的页面，不需要进行业务逻辑的处理（如查询商品信息、文章信息等）。



* 动态代码与动态网站

网页通过程序动态生成的。用户可以和服务器进行交互（可以根据用户输入的不同信息，返回不同的运行结果），可以处理复杂的业务逻辑。



**网站登录验证的过程:**

需要处理动态的数据，实现具体的业务逻辑 。

## PHP语言简介

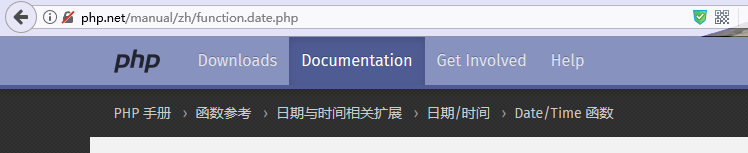
### 发展历程

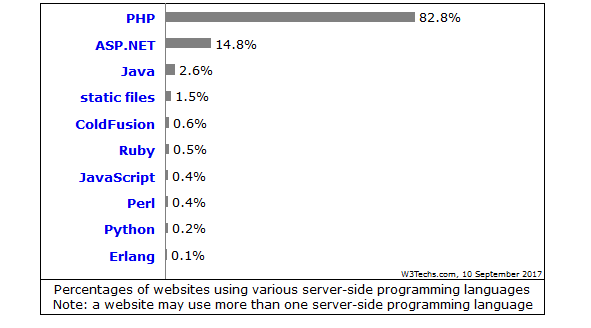
PHP/FI （form Interpreter）在 1995 年由 Rasmus Lerdorf 创建。脚本取名为“Personal Home Page”，用于显示Rasmus Lerdorf的个人履历以及统计网页流量。

PHP的全称为：Hypertext Preprocessor，超文本预处理器。

PHP2.0可以连接数据库，可以处理表单数据。

当前最新版本为7.2.0版。官网：<http://php.net/>







### PHP语言的功能特点

运行于web服务器端的WEB开发语言。用户可以通过浏览器发起对指定脚本的请求，服务器执行对应的脚本并返回执行结果。

[www.baidu.com/index.php](http://www.baidu.com/index.php)



开源（open source）软件，跨平台，常用操作系统稳定执行。Windows / Linux。

做WEB开发的经典组合 WAMP,LAMP（windows/ linux +apache+MySQL+PHP）基本都是开源软件。

* 入门简单,用户只需要关注应用，开发成本低。
* 支持的大多数主流数据库。MySQL，oracle,Redis等
* 存在大量针对PHP开发的框架如Laravel,ThinkPHP,Yii,CoderIgniter, Symfony等，加快开发速度。
* 解释型脚本。无需生成.dll或者.exe类似的文件，再执行。

# WEB开发基本概念及网站访问流程

## WEB服务器

PHP是用于网页服务器端编程的脚本语言。WEB服务器是安装了WEB服务器软件的计算机，存储网站脚本程序。

用户请求对应脚本时，服务器执行对应的脚本并返回执行结果。

### 常见的web服务器

Apache HTTP Server 服务器：是一个开源的网页服务器，跨平台，安全性高，是最流行的WEB服务器。

Nginx服务器：轻量级的高性能WEB服务器，特点：占用内存少、并发能力强。

IIS服务器：微软提供的基于Windows的服务器软件。

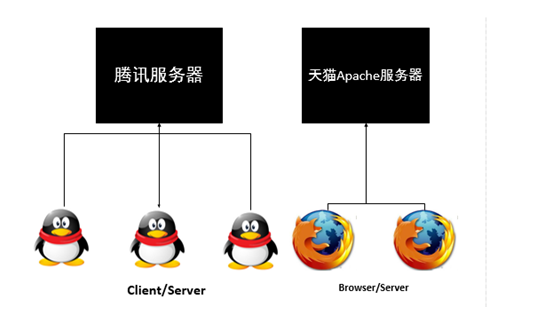
### 常用的开发环境

**WAMP**: Windows + Apache（WEB服务器，调度员） + MySQL（数据库服务器，用户名，密码，订单信息） + PHP（模块，解析PHP脚本）

LAMP: Linux + Apache + MySQL + PHP

LNMP: Linux + Nginx + MySQL + PHP

## 软件架构



### C/S架构

为Client/Server的简写。访问不同的服务器，需要各自安装不同的客户端软件。如：QQ，迅雷，百度云等

### B/S架构

B：browser，浏览器。

天猫：https://www.tmall.com/index.html

京东：https://www.jd.com/index.php

特殊的C/S模式。把所有的客户端都统一成浏览器，提供一个入口即可。用户输入请求的服务器以及脚本地址，获取对应的页面信息返回。未来软件的大趋势。



## IP地址

网络中有众多的设备，如何区分不同的设备？可以给每台设备分配一个身份信息。

### IP地址概念

IP地址相当于每一个连入互联网的设备的身份证号。在浏览器地址栏按照主机的IP地址信息进行访问，可以找到互联网上的这台主机。

IP地址分为四段。每段数字的取值范围是0-255,占据1个字节（8位）。总共4个字节。

百度的某一台主机：14.215.177.37



### 两种特殊的IP地址：

127.0.0.1作为本地测试的IP地址。本次测试时，客户端与服务器都安装在本地计算机上，数据无需任何网络传输。

192.168开头为局域网的地址。

## 域名

服务器的IP地址不便记忆。使用指定的字符串对应特定IP地址，可以简化使用。域名的英文为：Domain Name。

### 域名与服务器名

www . itcast .cn



万网域名注册参考地址：<https://wanwang.aliyun.com/domain/searchresult/>



**域名**为：itcast.cn

**服务器、主机名**为：[www.itcast.cn](http://www.itcast.cn) oa.itcast.cn bss.itcast.cn tb.itcast.cn

**顶级域名**为: .cn

其他的案例如：

qq.com wx.qq.com qzone.qq.com

### 特殊的域名

localhost，意思为本地主机。这是一个保留域名，主要用于本地测试，对应IP地址为127.0.0.1。

### 顶级域名：

.com: 商业机构

.cn: 中国国家、地区域名 .hk, .jp

.gov: 政府网站。

.org: 机构。

.edu: 教育网站。

.net: 网络服务商。 php.net

.mil: 军事。

## DNS

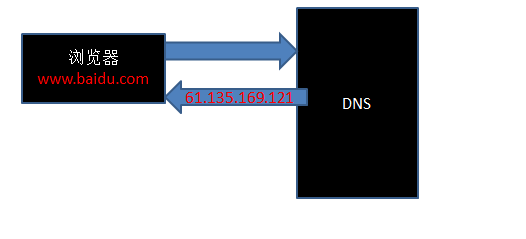
Domain Name System的简写,意思为域名解析系统。保存了服务器名与IP地址之间的映射关系。

[www.baidu.com](http://www.baidu.com) ----- 1.2.3.4

[www.itcast.cn](http://www.itcast.cn) ------2.3.4.5

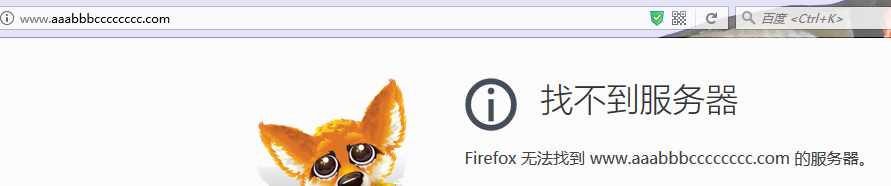
### DNS服务器

服务器的真正身份信息为IP地址。需要专门的服务器来解析IP地址与域名的映射关系。



DNS为一个分布式的数据库,存储IP地址与域名的映射关系。

当从浏览器输入域名地址，发起请求时，浏览器访问DNS服务器，获取域名对应的IP地址。然后使用该IP地址访问服务器。

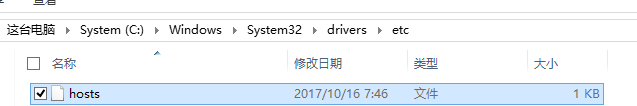


### 本地hosts文件

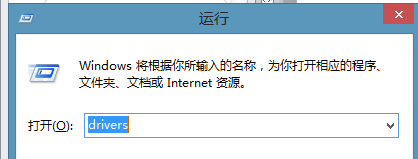
每次都请求网络中的DNS服务器，时间成本较高。windows自带的DNS文件，可以用来缓存域名与IP地址的对应关系。此外，浏览器也会缓存部分域名与IP地址的映射。

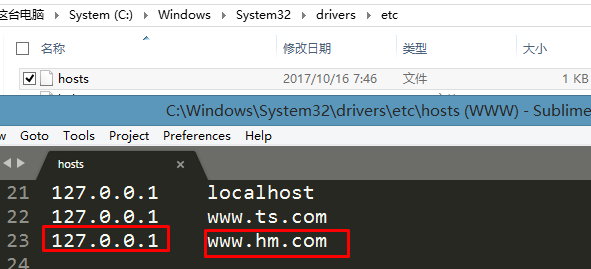
文件名：hosts

存储路径：C:\Windows\System32\drivers\etc\hosts



快捷访问： 运行窗口---drivers





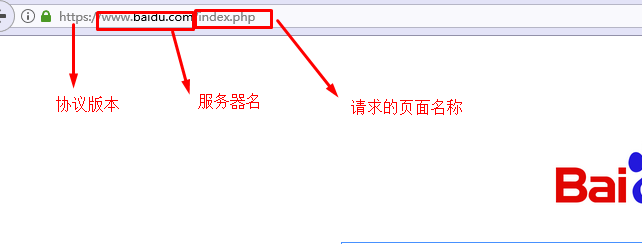
## 端口

一台计算机可能同时提供多种服务，如web服务，FTP服务（传输网站程序）

不同服务器软件监视不同的端口，来提供不同的服务（类型小区的信箱）。

web服务：80。

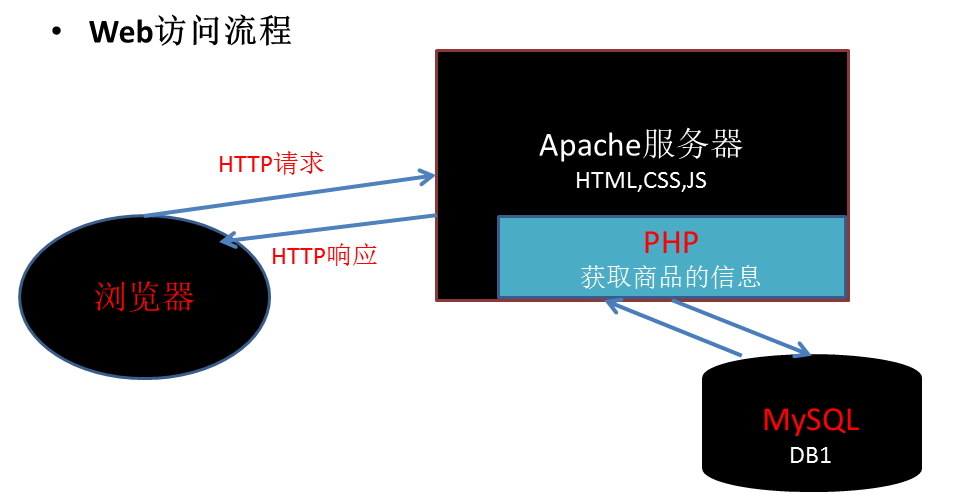
## URL



如：http://www.example.com:80/index.php

每个互联网上的资源都有访问标记符，这个标记符成为URL（uniform resource locator）统一资源定位符.其中网络协议（http/https），服务器名(www.itcast.cn)，端口号(80)，资源名(index.php)。

## web程序的访问流程





用户在浏览器地址栏输入请求URL，发起请求。通过DNS服务器解析出IP地址，找到对应的主机。

如果是以.html结尾的文件，直接返回给浏览器。如果是以.php结尾的文件，需要先执行PHP脚本。

此外，执行PHP脚本过程中，可能需要连接数据库服务器获取一些数据信息。所有代码执行完后，Apache将执行结果发给浏览器进行展示。

# 使用PHPstudy安装WAMP环境

## 安装包介绍

PHPstudy是WAMP开发的集成软件包，可以快速搭建WAMP开发环境，省去Apache服务器、MySQL数据库服务器的繁琐安装、配置过程。同时phpstudy支持在不同的PHP版本之间进行快速切换，更有利于开发调试。

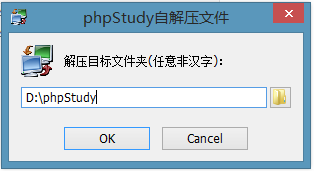
phpstudy官网：<http://www.phpstudy.net/>

类似的集成环境有WAMPServer,XAMPP等。

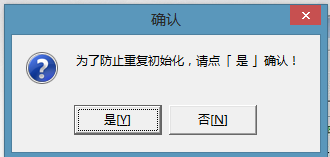
## 一键安装WAMP环境

### 选择安装服务器程序安装路径

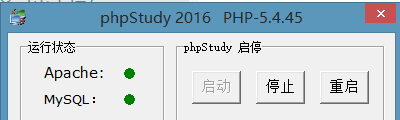
自行选择安装路径。



以下选择：是

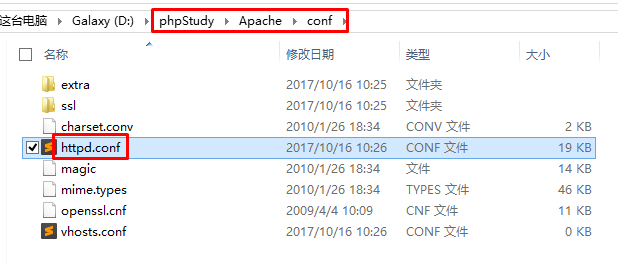


正确的状态：



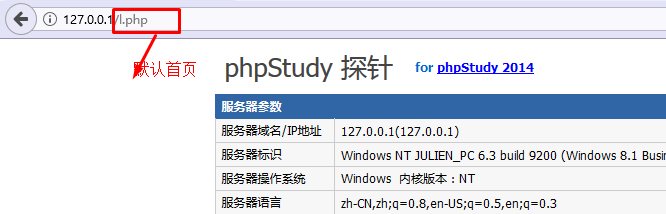
### 安装产生目录介绍

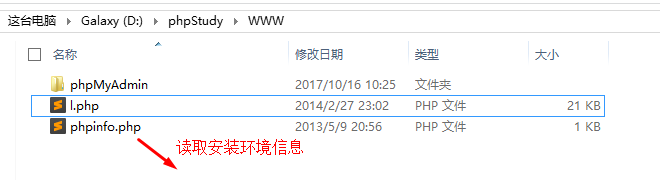
|  |  |
| --- | --- |
| www | 默认的网站程序目录 |
| php | PHP模块的路径，包含支持的各个版本PHP |
| mysql | mysql安装以及数据存储目录 |
| apache | apache服务器的安装目录 |
| apache/bin | 二进制程序目录 |
| apache/conf | apache配置文件目录 |

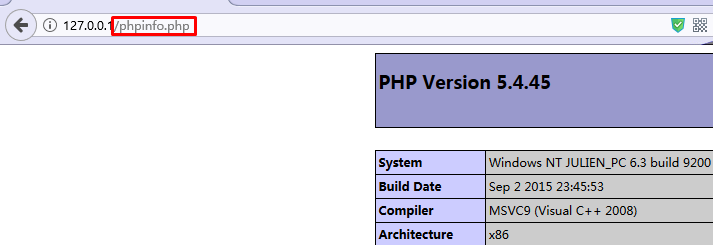


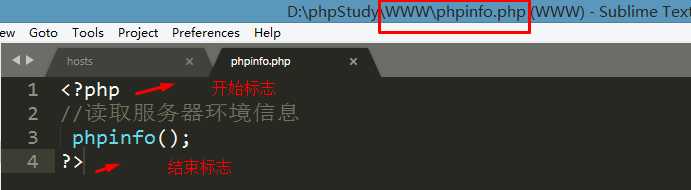
## 测试是否安装成功

默认的网站程序保存在安装目录下的WWW文件夹中，可以通过主机地址以及脚本名进行直接访问。其中l.php为默认的网站首页。









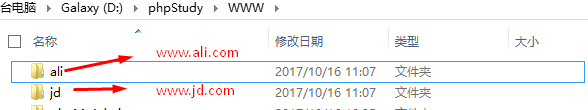
## 设置服务器时区

在PHP的配置文件中设置date.timezone选项，确保服务器可以读取正确的时区信息。

操作步骤：phpstudy🡪 其他选项菜单 🡪PHP扩展及设置 🡪 参数值设置🡪 date.timezone 🡪PRC

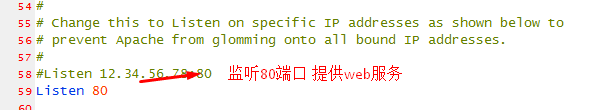
保存重启服务器（只有在重启时才会读取配置文件）

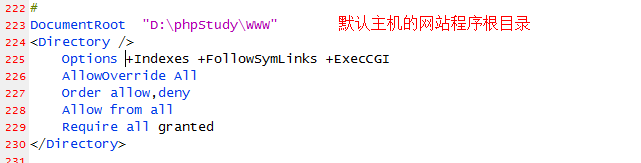
# 虚拟主机



## 虚拟主机介绍

默认主机的配置：

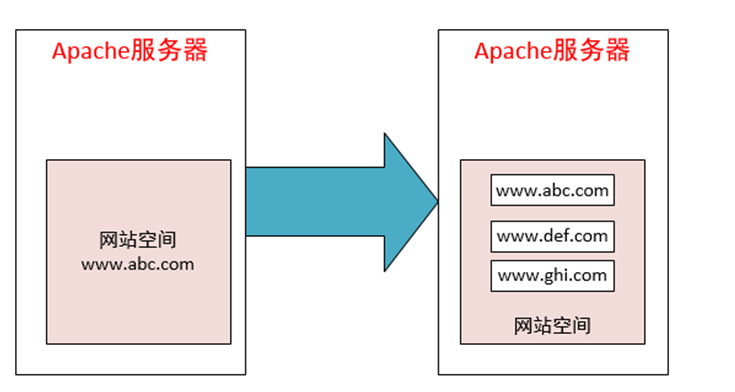




Apache服务器提供虚拟主机的功能。通过虚拟主机(Virtual Host)，可以在一台服务器上部署多个网站。

可以将网站根目录中的各个空间对应成不同的域名空间，通过访问对应的域名既可以访问网站程序，这样的方式为基于域名的虚拟主机。

**原理图如下：**



设置不同的网站程序目录：

[www.abc.com](http://www.abc.com) d:/www/a

[www.bcd.com](http://www.bcd.com) d: /www /b

[www. cde.com](http://www.abc.com) d: /www /c

## 开启虚拟主机配置文件

Apache服务器有两个配置文件: 主配置文件httpd.conf与虚拟主机的配置文件httpd-vhosts.conf。

主配置文件：

Listen 80 //监视服务器的80端口，提供web服务。

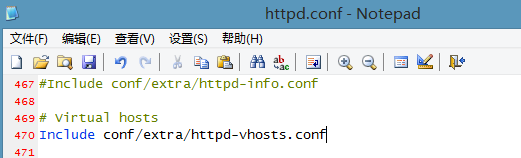
DocumentRoot "D:\phpStudy\WWW" //指定网站的根目录

通常情况下，只加载主配置文件httpd.conf。需要在主配置文件httpd.conf中，加载虚拟主机的配置文件httpd-vhosts.conf，这样在虚拟主机的配置项才会生效。

操作步骤：phpstudy🡪其他选项菜单🡪打开配置文件🡪httpd-conf🡪

Include conf/extra/httpd-vhosts.conf

保存重启Apache服务器。



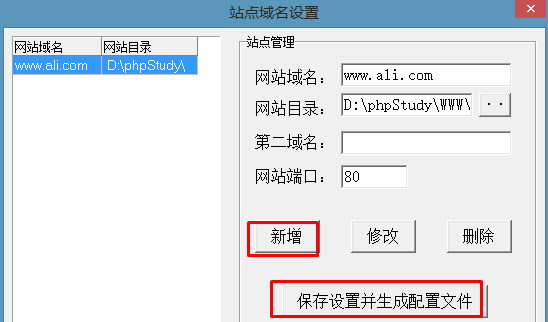
## 配置基于域名的站点信息

在虚拟主机的配置文件httpd-vhosts.conf中进行配置。

需要指定服务器名以及对应虚拟主机的网站根目录。配置完成后，点击保存并生成配置文件。

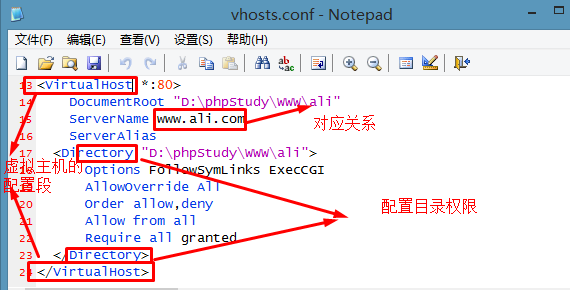
**步骤：**

**其他选项菜单🡪站点域名设置🡪**



检查自动生成的虚拟主机配置文件：

vhost-conf

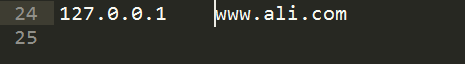


## 解析虚拟主机

互联网设置的唯一身份信息是IP地址，需要在hosts文件中，建立域名与IP地址的对应关系，才能确保域名在本地找到服务器地址，访问本地的虚拟主机中的程序。

因此，需要在本地hosts文件中添加一行解析记录：即IP地址与域名的对应关系

操作步骤：phpstudy🡪其他选项菜单🡪打开hosts

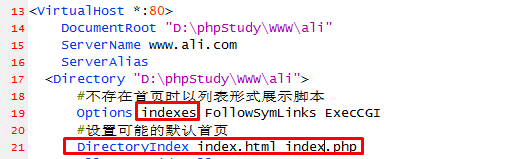


## 创建测试文件





## 设置列表显示脚本

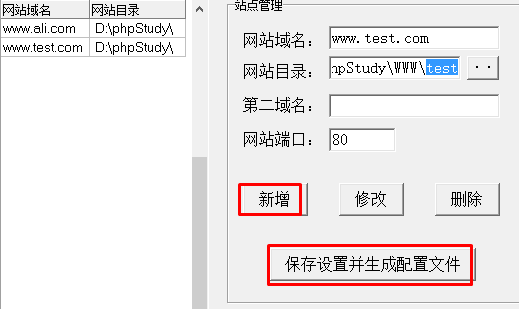


注意：

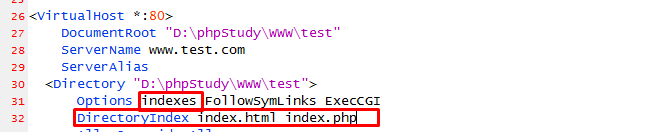
虚拟主机权限是高于默认主机。

## 配置[www.test.com](http://www.test.com)

### 添加站点



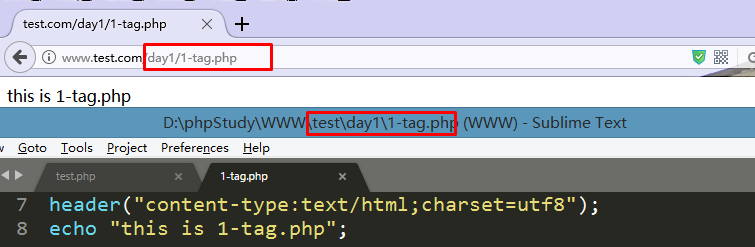
### 修改配置文件，列表形式显示脚本



重启

### 解析





# PHP基本语法

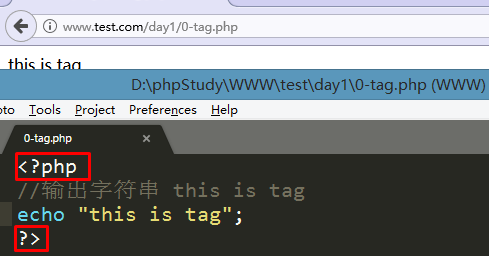
## PHP标记

PHP代码支持与HTML代码混编。PHP标记用来分区PHP代码，从指定位置开始解析到指定位置结束解析。

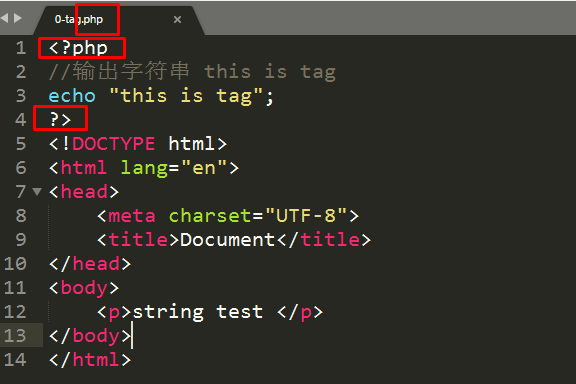
<?php

//PHP代码

?>



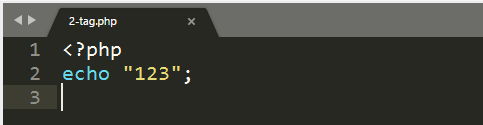
* PHP本身支持与HTML混编，执行时，先解析运行PHP代码，将结果与HTML部分一并返回给浏览器。



* 混编的文件后缀必须为.php，Apache才会调用PHP解析



* 纯PHP文件建议省略结束标记。

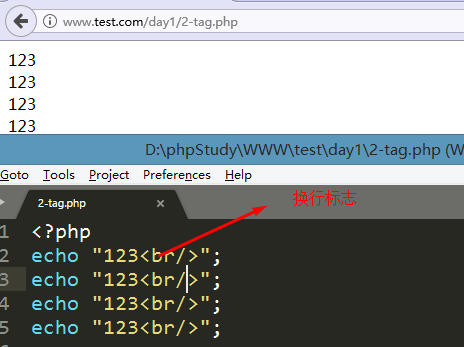


## 语句结束标记

使用英文状态下的分号;作为语句结束标志，意味着该行代码结束了。

**注意事项：**

* PHP只有一行代码时，可以省略语句结束标记
* 代码注意换行及缩进。实现一定功能的代码之间添加空白行，便于团队协作及代码维护。



## 注释

注释将打算不执行的代码包含起来。程序不会解析该部分内容。

完美的代码必须包含清晰的注释，包含变量的意义，函数的功能等。

### 单行注释

// 代码

快捷键：ctrl + /

### 块注释

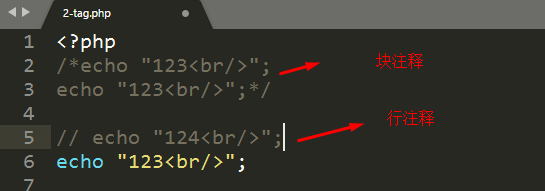
/\*

代码

\*/

多行注释，注释多行的代码

快捷键：ctrl ＋　shift + /



## 打造自己的PHP开发IDE

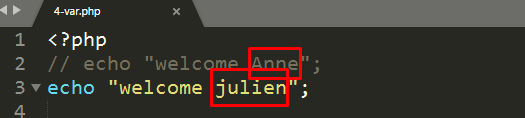
为了开发便利，可以安装一些插件如sublime tmpl、colorPicker，DocBlockr等。

参考地址：<http://blog.csdn.net/doubleface999/article/details/55798749>



# 变量

## 变量介绍



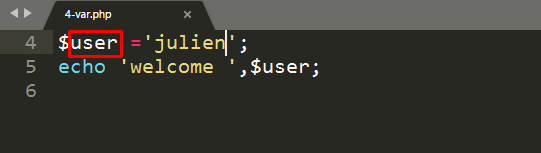
可以将姓名保存在变量中，避免每次修改原始文件。

### 定义

在脚本周期中，值可以改变的量(variable)。即程序运行中，用来存储数据的单位或者空间。

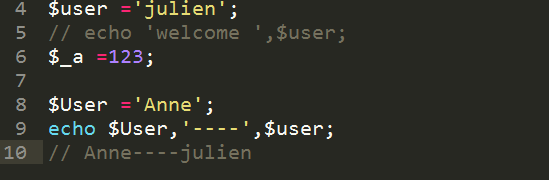
### 变量的组成

$开头，后面为变量名。



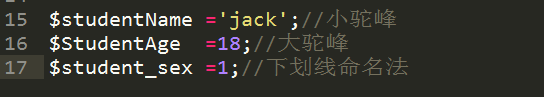
### 变量名的命名规则

* 由英文字母、数字、下划线等多个字符组成，以字母或者下划线开头。
* 变量命名严格区分大小写。



* 建议命名使用英文单词，见名知意。
* 变量名由多个单词组成：

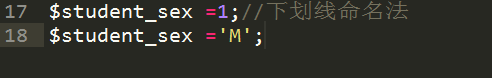
小驼峰法：第一个单词首字母小写，其余单词的首字母大写（小驼峰）推荐。$stuName



## 变量操作

### 变量声明

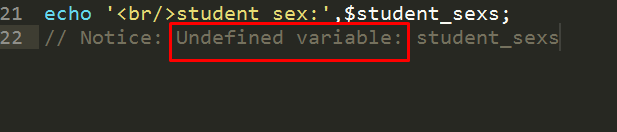
* 一般声明与赋值同时进行。
* 不需要单独指定变量的类型，变量的类型由值决定。
* PHP本身为弱类型语言，值发生改变时类型自动变化。



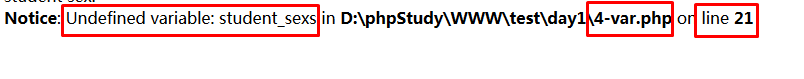
### 读取变量的值

变量必须先声明后使用，否则提示未定义变量“undefined variable”



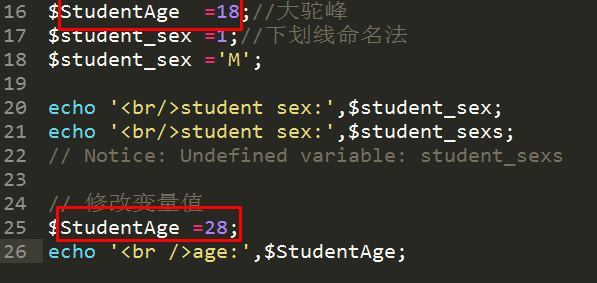


注意错误提示以及行号：



### 变量值的修改

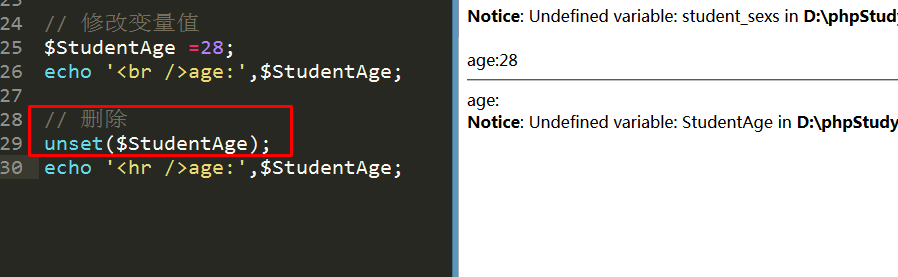
直接赋新值即可。



### 删除变量

unset($var);

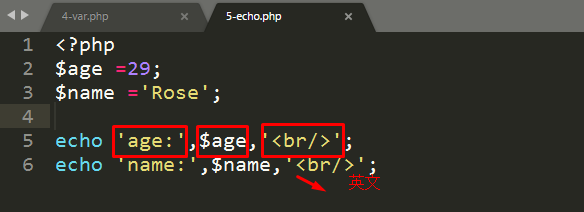
删除变量名及值之间的对应关系。变量名及变量值不会立即消失，PHP有专门的垃圾回收机制处理。



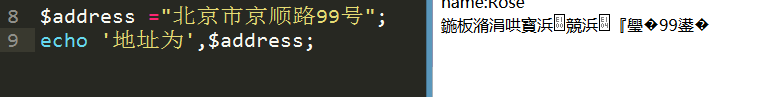
## 常见的输出语句

### echo语句

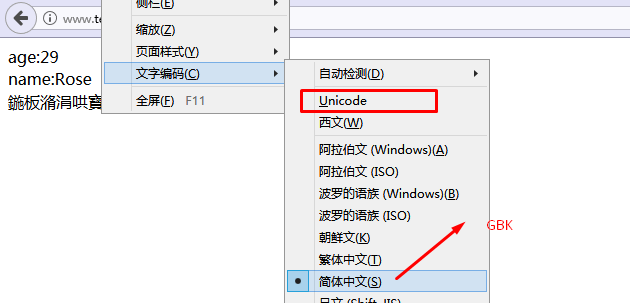
* 输出字符串、变量及表达式
* 可以同时输出多个变量及字符串。
* 不是函数，是语言结构



### 解决输出中文字符等特殊字符显示乱码问题



乱码：字符集出错。

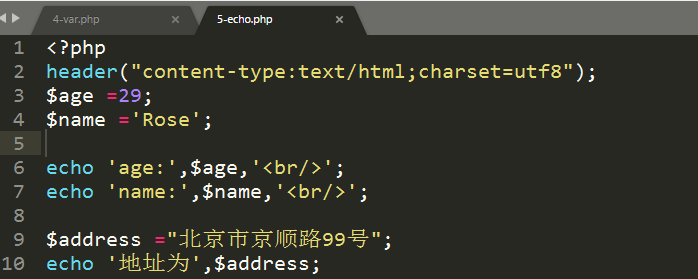


通过html的meta标记或者PHP的header函数实现：

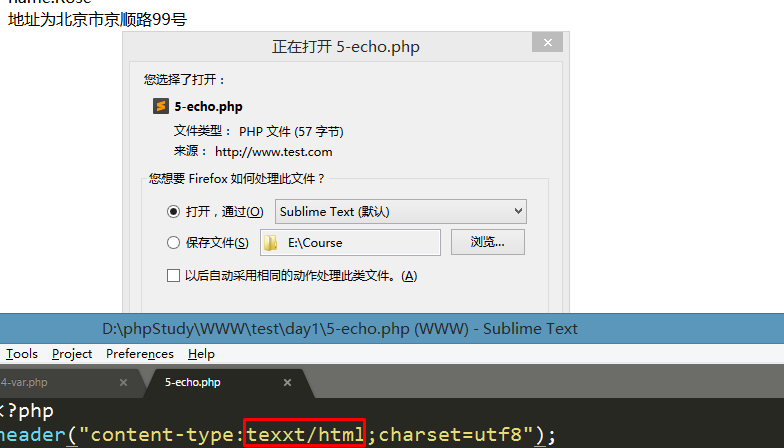
header(“content-type:text/html;charset=utf8”);

//告知浏览器返回数据的类型：text/html 将来使用html解析器解析

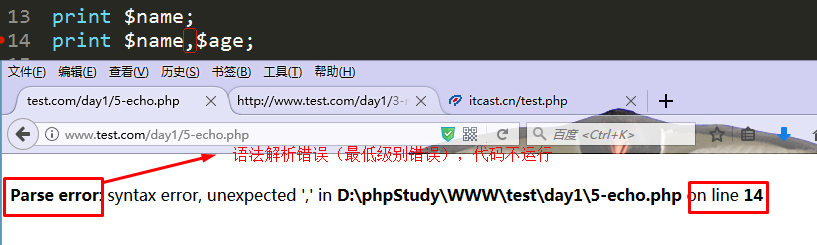
//使用utf8字符集展示



类型出错，提示下载：



### print语句



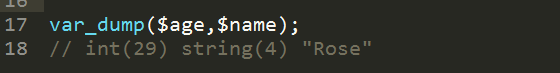
* 可以输出字符串及变量及表达式
* 与echo语句的区别为：

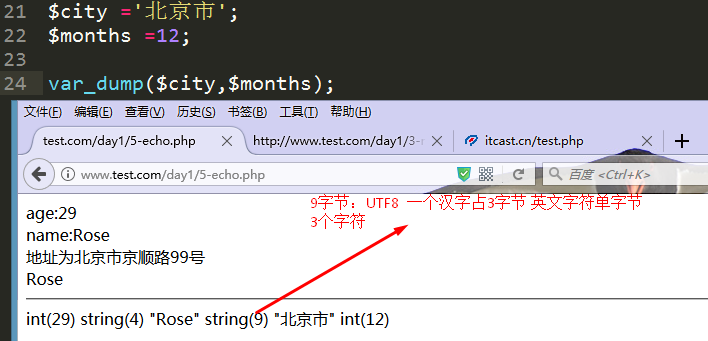
print不支持同输出多个字符串及变量。

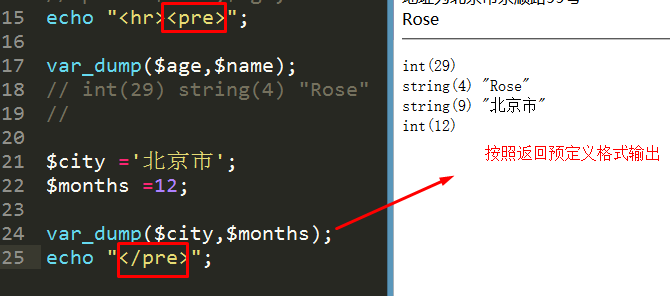
相同点：为语言结构，不是函数。

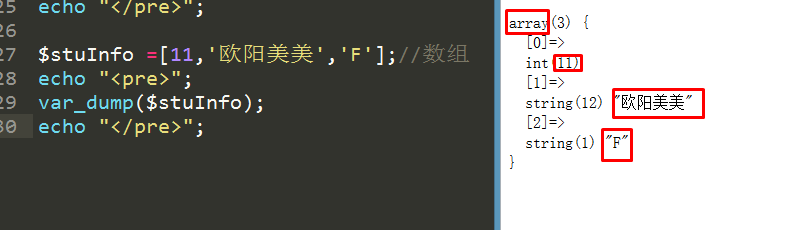
### var\_dump()

输出一个或者多个变量的类型，长度及值。一般用于打印复杂变量（结合pre标记），如数组,对象等。



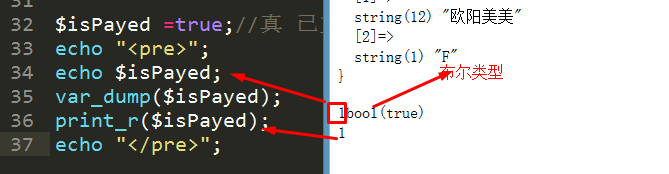




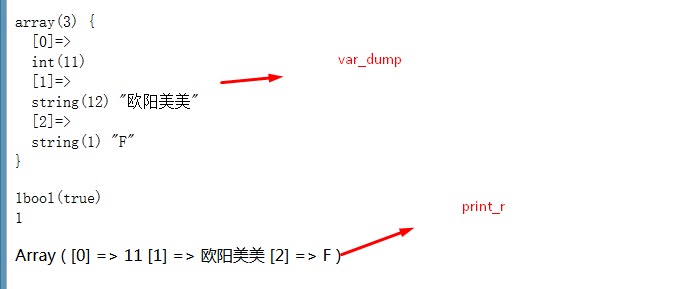


### print\_r()

输出变量的值。一般用于打印变量。对于字符串类型、整型、浮点型，打印变量本身。不能打印布尔类型的原始值。对于复杂类型的变量，如数组、对象，按照一定的格式显示。



打印的信息较var\_dump简单。



# PHP数据类型

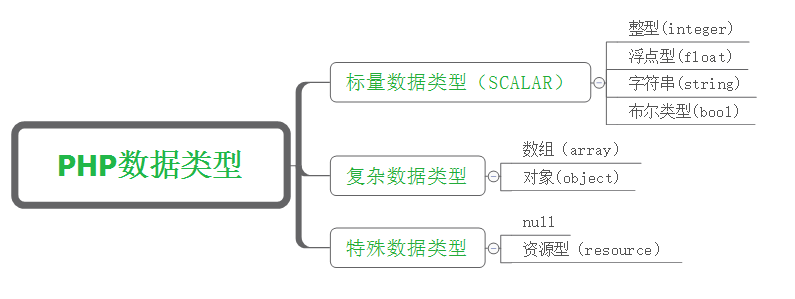
Pascal之父Nicklaus Wirth凭借一句话获得图灵奖：

算法 + 数据结构 = 程序

数据结构：数据的类型及在内存的表现形式。

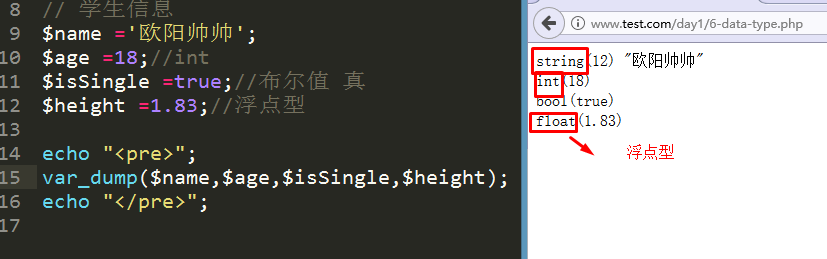
算法：高效解决问题的思路。

## PHP数据类型有哪些？



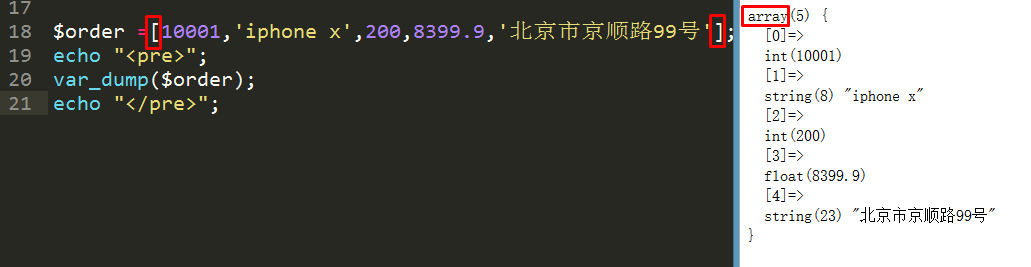
## 标量（scalar,简单）数据类型

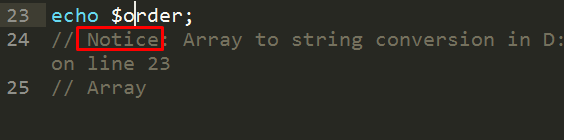
一般为单个数值。包括整型、浮点型、字符串类型、布尔类型。



## 复合数据类型

包括数组及对象。数组有多个元素，对象有多个成员、属性。

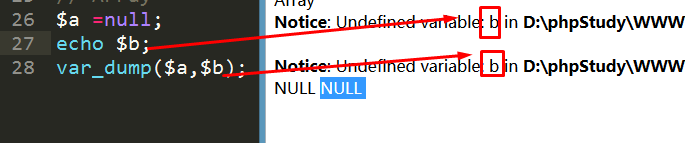




## 特殊类型

### null

为null的几种情况：变量未赋值、变量被删除、变量被赋值为null



### resource

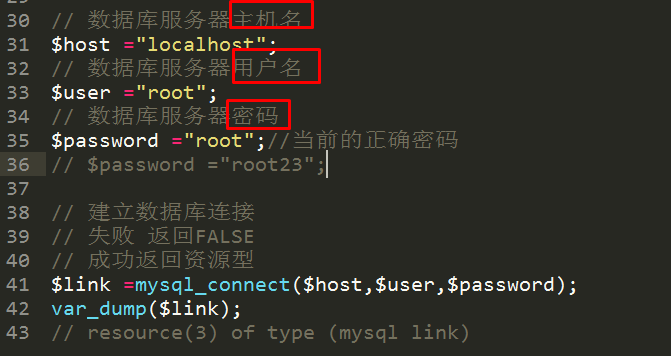
资源型，为到第三方数据的引用（如成功的数据库连接、绘制图像创建画布、打开文件、打开文件夹等）。

PHP连接MySQL：

mysql\_connect(数据库服务器主机名，用户名，密码)；

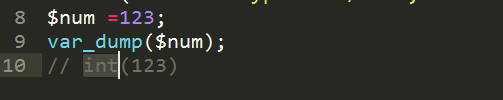
//连接成功 返回资源型 。失败false

资源通过函数进行建立，通过函数进行其他操作。

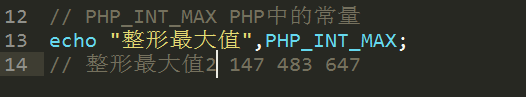


## 整型类型

整型，英文integer，类型描述字符串为int



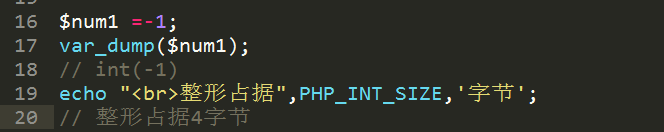
### 整数的表示形式



PHP的整型占据4个（PHP\_INT\_SIZE）字节空间，最大值为PHP\_INT\_MAX(21亿多)表示。整型的表示形式为二进制，八进制，十进制，十六进制。

1字节byte = 8 bit二进制位

4字节：有符号数（可存储负数），1（最高位表示符号位，1负0正）31数值位。



### 各种进制

生活中的进制：半斤八两（16进制），1小时等于60分（60进制）。

除十进制外，还有二进制（0-1），八进制（0-7），十六进制（0-9-A-F）。

进制的本质：**逢X进1**

|  |  |  |  |
| --- | --- | --- | --- |
| 不同进制之间的对比 | | | |
| 十进制 | **二进制** | **八进制** | **十六进制** |
| 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 |
| 2 | 10 | 2 | 2 |
| 3 | 11 | 3 | 3 |
| 4 | 100 | 4 | 4 |
| 5 | 101 | 5 | 5 |
| 6 | 110 | 6 | 6 |
| 7 | 111 | 7 | 7 |
| 8 | 1000 | 10 | 8 |
| 9 | 1001 | 11 | 9 |
| 10 | 1010 | 12 | a |
| 11 | 1011 | 13 | b |
| 12 | 1100 | 14 | c |
| 13 | 1101 | 15 | d |
| 14 | 1110 | 16 | e |
| 15 | 1111 | 17 | f |

### 其他进制准换为十进制

原则：按权展开求和，每位数字代表不同的权重。最低一位表示X0，依次为X1，X2…

(112)8= 2\*80+1\*81+1\*82 =2 + 8 + 64 =（74）10

(21)16 =1\*160+ 2\*161 = 1 + 32 =33

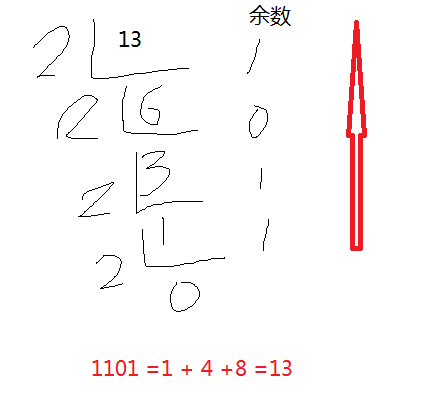
(1111)2=15

### 十进制转换为其他进制X

10 =8 + 2= 2^3 + 2^1 =1010

转换原则：

整数部分：除X取余，直至商为0，停止计算(反向组织).



### 进制转换函数

**decimal:十进制 10**

**binary:二进制 1010**

**octect:八进制 0123**

**hexadecimal:十六进制 0x123**

decbin（int 值）：十进制转二进制

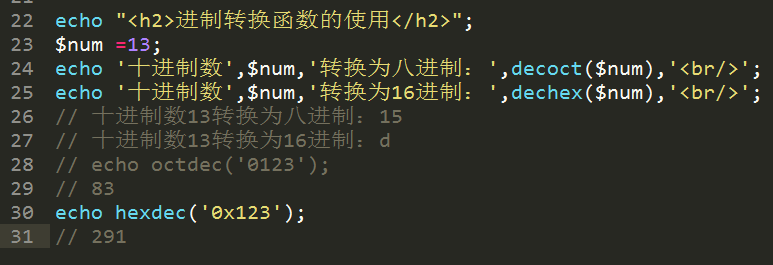
decoct(int 值): 十进制转八进制

dechex(int 值): 十进制转16进制

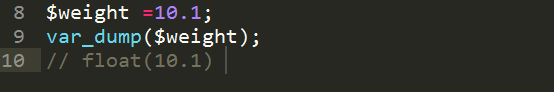
bindec（string 值）:二进制转十进制

octdec（string 值）：八进制转十进制

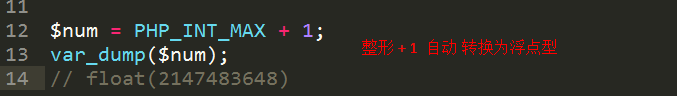
hexdec（string 值）:十六进制转十进制



## 浮点型类型

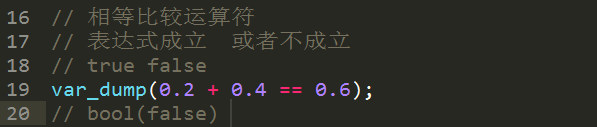


* 小数。一般用float,double表示。
* 一般程序语言有单精度、双精度之分（精度为小数的位数）。PHP只实现了双精度。
* 占据的空间为8字节。表示最大值范围比整数大很多。整型范围无法表示的数字可以使用浮点型表示,以及超出整数表示范围，自动使用浮点数表示



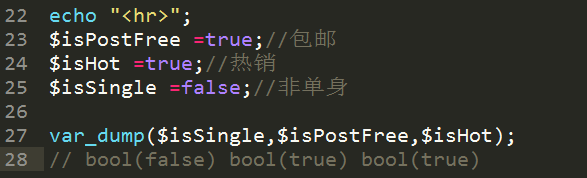
* 浮点数转换为二进制部分数值为不精确表示（进制转换小数部分的乘X取整，可能遇到无限不循环的情况）。由于内部表达方式的原因，比较两个浮点数是否相等是有问题的。





## 布尔类型介绍

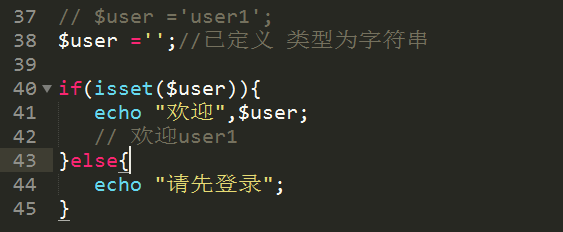
* 表示事物的真或者假状态，分别对应true,false。一般用来做条件判断（满足条件，不满足条件）
* true及false不区分大小写，TRUE，FALSE。
* 建议小写。
* 本身不是常量。



### isset（）

变量未设置或者设置为null,返回false（认为变量没有设置）。其余情况全部为true。一般用来判断变量是否设置，因为变量未设置，无法直接使用。

变量设置为空字符串或者数字0时，此时变量有具体的类型，表示已经设置了。



### empty()

判断变量是否为空。PHP中认为变量的值为：""、0、"0"、NULL、FALSE、[]时，变量虽然定义了，但是无实际的意义。为空。

