# 复习

## IP地址 4个字节，每个字节最大值255

## 域名 Domain ,便于记忆的计算机名称

## DNS 双向解析，解析域名到IP地址，解析IP到域名

## hosts文件 本地的相当DNS解析文件，优先级最高（相对于网络上的DNS）

## WAMP

WAMP： Windows Apache MySQL PHP

LAMP： 互联网网站建设黄金组合

## Apache 配置

配置文件 /conf/httpd.conf

注释符 #

### 配置文件解读

1. ServerRoot 是指Apache安装的位置

2.Listen 监听端口，可以配置多个Apache端口，端口设置值0-65535,1024以下端口为系统所用。

3. ServerName localhost:80 服务器名称

4.DocumentRoot "c:/wamp/www" 指定文档根目录

5. <Directory "c:/wamp/www/">。。。 </Directory>设置文档根目录访问权限

### Alias 别名

用alias 别名 实际物理地址

如：alias /pic "c:/images/"

添加别名后，记住重启Apache。

### 虚拟主机

#### 通过端口号设置虚拟主机

* Listen 8080 添加8080端口的监听
* 用NameVirtualHost 设置虚拟主机，如：NameVirtualHost \*：8080
* <virtualhost \*:8080> ...</virtualhost>设置虚拟主机

如：

<virtualhost \*:8080>

ServerName localhost

DocumentRoot "c:/wamp/www"

<directory "c:/wamp/www">

options indexes

allow from all

</directory>

</virtualhost>

* 重启Apache

#### 通过域名创建虚拟主机

* 本地编辑hosts文件 ，添加域名并指向127.0.0.1
* 编辑Apache配置文件httpd.conf
* 添加 NameVirtualHost \*:80
* 添加以下内容：

<virtualhost \*:80>

ServerName www.dangdang.com

DocumentRoot "c:/wamp/www"

<directory "c:/wamp/www">

options indexes

order deny,allow

allow from all

</directory>

</virtualhost>

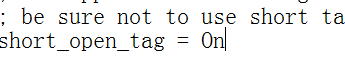
# PHP基础一

## 定界符

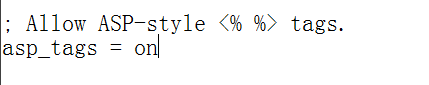
PHP有四种定界符。

1.以<?php 开始 ，以?>结束，称为标准PHP定界符，推荐使用

2.以<?开始，以?>结束，要在php配置文件中设置为打开，php配置文件php.ini中设置shor\_open\_tag on（默认是off）如下图：



3.使用ASP定界符以<%开始，以 %>结束。使用时，现在配置文件php.ini中修改如下图：



4.以<script language="PHP">开始，以</script>结束。

## 变量与常量

变量定义：以$符号开始后面是变量名称

变量名称必须字母下划线开头后面可以是字母数字和下划线，变量区分大小写。

常量定义：不会发生变化量，使用define(常量名,常量值,是否区分大小写(默认是false区分大小写))，如：define('pi','3.14',true) #不区分大小写。

defined()函数可以用来判断是否常量名已经被定义。

常量名不允许重复定义。

## 数据类型

|  |  |  |
| --- | --- | --- |
| **数据类型** | **描述** | **举例** |
| ***标量类型*** | | |
| 布尔类型 | Boolean | True, false |
| 整型 | Integer | 100,-100 |
| 浮点型 | Float | 3.6 ,1.3E9 |
| 字符串 | String | "Hello" |
| ***组合类型、结构类型*** | | |
| 数组 | Array | Array（1,2,3） |
| 对象 | Object |  |
| ***特殊类型*** | | |
| 资源类型 | Resource |  |
| 空类型 | Null |  |

### 字符串

字符串定义有3种方法：

1. 用单引号定义

如：'hello'

单引号值包含纯粹的字符串，不会对字符串的内容进行处理。

2. 双引号定义

如："hello"

可以对字符串的内容进行处理。比如把双引号中变量转换成值。

在字符串的处理中，如果字符串中包含变量，建议加上{}对变量进行整体化区分处理。

推荐使用写法: ”...{变量}..."。

3. 定界符定义

用<<<开始后面跟标签（该标签自行定义），最后以标签独占一行结束。

如：

$a=<<<strtag

hello php

welcome everybody

strtag;

注意：最后的结束标签一定要独占一行并靠左写，然后要加分号";"结束。

### 数组

数组的定义：

索引数组

定义数组要使用array函数，格式：

array(数组元素值)

数组下标从0开始。

可以设置数组元素的下标，方法：用"=>"。

注意：下标是可以自行设置。

数组中键与值得概念：

键key，值value。

$str=array(1,4=>2,'hello',8);

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| key | 0 | 4 | 5 | 6 |
| value | 1 | 2 | hello | 8 |

关联数组：键为字符串的数组。

$nbaer=array("name"=>"Jordan","sex"=>"male","age"=>53);

|  |  |  |  |
| --- | --- | --- | --- |
| key | Name | Sex | Age |
| value | Jordan | Male | 53 |

注意：键具有唯一性，重复使用同一键名，同一键名时，后面的元素会覆盖前面的元素。

问题：

$test=array('name'=>'Smith',5=>'male',53)

结构是：

Array ( [name] => Smith [5] => male [6] => 53 )

## 表达式和语句

表达式：具有完整值的都称为表达式，由常量，变量，并且用运算符连接起来就有值都称为表达式。

语句：就是表达式后面加个分号";"。

如：

$a+$b 是表达式

$a\*$b 是表达式

$a\* 不是表达式

$a\*$b; 不是表达式是语句

PHP中必须用分号做语句的结束符。

## 输出语句

4个常用的

|  |  |  |
| --- | --- | --- |
| ***名称*** | ***作用*** | ***例子*** |
| echo | 输出函数，结构可以同时输出多个，并用逗号","隔开,并且没有返回值。 | Echo $a,$b,$c; |
| Print | 只能输出一个元素，并具有返回值，返回为0表示输出失败，为1表示成功。 | Print $a; |
| Print\_r | 用于输出完整数组。 | Print\_r($arr); |
| Var\_dump | 输出元素，最强的输出函数，是PHPer的首选函数。 | Var\_dump($arr); |

举例：

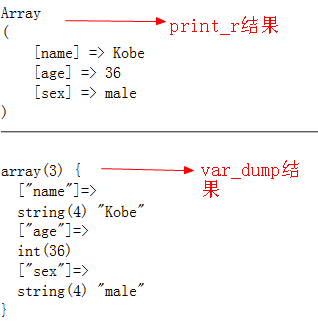
$arr=array('name'=>'Kobe','age'=>36,'sex'=>'male');

print\_r($arr);

echo "<hr>";

var\_dump($arr);

运行结果：



## 算术运算

|  |  |  |
| --- | --- | --- |
| 运算 | 描述 | 举例 |
| + | 加 | $a+$b |
| - | 减，负数功能 | $a-$b |
| \* | 乘 | $a\*$b |
| / | 除 | $a/$b |
| % | 模 | $a%$b |

+运算有其特殊性：可以将字符串转换成数值型进行计算。

%运算：是求余运算。10%3 值为1。

思考：

$a%2==1判断$a是否为奇数。

一目运算：是仅操作一个对象的运算，

++ 自增运算

如

$a++ 该使用表达式的值，在自增

++$a 先自增再计算表达式

-- 自减运算，使用与++一致

注意：++和--运算只能使用在变量上。

## 字符串运算

只有一个运算符：连接运算，运算符"."。

如：$a.$b或者$a."hello".$b;

## 关系运算

也称为比较运算

|  |  |  |
| --- | --- | --- |
| ***运算符*** | ***描述*** | ***用法*** |
| == | 相等 | $a==$b |
| === | 全等1.值要相等，2类型也一致 | $a===$b |
| != | 不等 | $a!=$b |
| !== | 不全等表示值不相等或类型不相等 | $a!==$b |
| <> | 不等 | $a<>$b |
| > | 大于 | $a>$b |
| < | 小于 | $a<$b |
| >= | 大于等于 | $a>=$b |
| <= | 小于等于 | $a<=$b |

## 逻辑运算

|  |  |
| --- | --- |
| ***运算符*** | ***描述*** |
| && | true&&true 值才为真 |
| || | false||false 值才为假 |
| ! | !ture 为假 !false 为真 |

## 赋值运算

|  |  |  |
| --- | --- | --- |
| 运算符 | 描述 | 举例 |
| = | 赋值 | $a=10 |
| += | 加等 | $a+=$b 相当于 $a=$a+$b |
| -= | 减等 | $a-=$b 相当于 $a=$a-$b |
| \*= | 乘等 | $a\*=$b相当于 $a=$a\*$b |
| /= | 除等 | $a/=$b相当于 $a=$a/$b |
| %= | 模等 | $a%=$b相当于 $a=$a%$b |

## 三目运算

语法：

表达式1?表达式2：表达式3

当表达式1为真执行表达式2，否则运行表达式3

代码：

<?php

$a=99;

print($a%2?"奇数":"偶数");

?>

结果是：奇数

## 判断奇偶数实例：

1.创建表单

2.编写PHP程序

程序如下：

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="zh-cn">

<head>

<meta http-equiv="Content-Type" content="text/html;charset=UTF-8" />

<title>网页标题</title>

<meta name="keywords" content="关键字列表" />

<meta name="description" content="网页描述" />

<link rel="stylesheet" type="text/css" href="" />

<style type="text/css"></style>

<script type="text/javascript"></script>

</head>

<body>

<?php

/\*

if(isset($\_GET['num'])) {

$n=$\_GET['num'];

\*/

if(isset($\_POST['num'])){

$n=$\_POST['num'];

$str=($n%2==1?"奇数":"偶数");

print($n."是".$str);

//var\_dump($n);

}

?>

<!--

<form method="get">

-->

<form method="post">

<input type=text name=num />

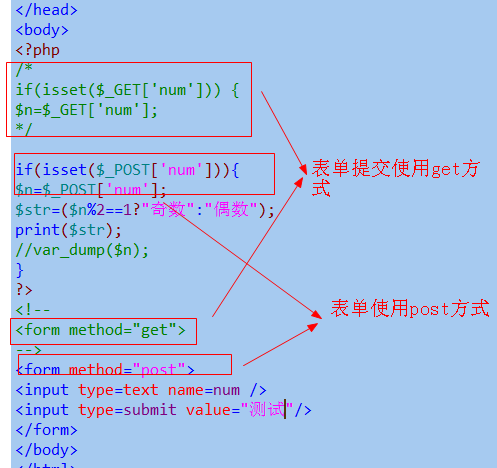
<input type=submit value="测试"/>

</form>

</body>

</html>

程序说明



程序运行结果：

