

## PHP 的意思

PHP 是一个基于服务端来创建动态网站的脚本语言，您可以用 PHP 和 HTML 生成网站主页

## 什么事面向对象？主要特征是什么？

面向对象是程序的一种设计方式，它利于提高程序的重用性，使程序结构更加清晰。主要特征：封装、继承、多态。

## SESSION 与 COOKIE 的区别是什么，请从协议，产生的原因与作用说明？

1、http 无状态协议，不能区分用户是否是从同一个网站上来的，同一个用户请求不同的页面不能看做是同一个用户。

2、SESSION 存储在服务器端，COOKIE 保存在客户端。Session 比较安全，cookie 用某些手段可以修改，不安全。Session 依赖于 cookie 进行传递。

禁用 cookie 后，session 不能正常使用。**Session 的缺点：**保存在服务器端，每次读取都从服务器进行读取，对服务器有资源消耗。Session 保存在服务器端的文件或数据库中，默认保存在文件中，文件路由 php 配置文件的 session.save\_path 指定。Session 文件是公有的。

## 不使用 cookie 向客户端发送一个 cookie.

理解：session\_start() 开启时，生成一个常量 SID，当 COOKIE 开启时，这个常量为空，当 COOKIE 关闭时，这个常量中存储了 PHPSESSID 的值。通过在 URL 后加一个 SID 参数来传递 SESSIONID 的值，从而使客户端页面可以使用 SESSION 里面的值。当客户端开启 COOKIE 和服务器端开启 SESSION 时。浏览器第一次请求，服务器会向浏览器端发送一个 COOKIE 里面存储 SESSIONID。当浏览器第二次请求时，会把已存在的 COOKIE 一起提交到服务器端。

## 简述 Cookie 的设置及获取过程

设置 COOKIE 的值：

Setcookie(名称，值，保存时间，有效域)；

获取值：\$\_COOKIE['名称']；

## HTTP 状态中 302、403、500、200、404、502 代码含义？

一二三四五原则：一. 消息系列 二. 成功系列 三. 重定向系列 四. 请求错误系列 五. 服务器端错误系列 302:临时转移成功，请求的内容已转移到新位置 403:禁止访问 500:服务器内部错误 401 代表未授权。 200 是请求成功，404 是文件未找到，502 是服务器内部错误。

## 请写出数据类型(int char varchar datetime text)的意思；请问 varchar 和 char 有什么区别？

19、Int 整数 char 定长字符 Varchar 变长字符 Datetime 日期时间型 Text 文本型 **Varchar 与 char 的区别** char 是固定长度的字符类型，分配多少空间，就占用多长空间。Varchar 是可变长度的字符类型，内容有多大就占用多大的空间，能有效节省空间。Varchar 是变长，节省存储空间，char 是固定长度。查找效率要 char 型快，因为 varchar 是非定长，必须先查找长度，然后进行数据的提取，比 char 定长类型多了一个步骤，所以效率低一些

## MyISAM 和 InnoDB 的基本区别？索引结构如何实现？

MyISAM 类型不支持事务处理等高级处理，而 InnoDB 类型支持。MyISAM 类型的表强调的是性能，其执

行速度比 InnoDB 类型更快，但是不提供事务支持，而 InnoDB 提供事务支持以及外部键等高级数据库功能。

创建索引：alert table tablename add index ( 字段名 )

## isset() 和 empty() 区别

Isset 判断变量是否存在，如果存在则返回真，empty 判断变量是否为空为假，如果为空为假则返回真。

## include 与 require 的区别？

1. include() 在执行文件时每次都要进行读取和评估 require() 文件只处理一次(实际上文件内容替换了 require() 语句)

2. require() 通常放在 PHP 脚本程序的最前面

include() 的使用和 require() 一样，一般放在流程控制的处理区段中，PHP 脚本文件读到 include() 语句时，才将它包含的文件读进来，这种方式，可以把程序执行时的流程简单化

3. require() 和 include() 语句是语言结构，不是真正的函数，可以像 PHP 的其他语言结构一样

4. include\_once() 和 require\_once() 语句也是在脚本执行期间包括并运行指定文件，与 include()require() 唯一的区别是如果文件中的代码已经被包括了，则不会再次包括。

5. require() 包含文件失败，停止执行，给出错误(致命的)

include() 常用于动态包含。

通常是自动加载的文件，即使加载出错，整个程序还是继续执行

一个页面声明，另一个页面调用

包函文件失败，继续向下执行，返回一条警告

## PHP 字符串中单引号与双引号的区别？

单引号不能解释变量，而双引号可以解释变量。

单引号不能转义字符，在双引号中可以转义字符。

## 请说明 PHP 中传值与传引用的区别。什么时候传值什么时候传引用？

按值传递：函数范围内对值的任何改变在函数外部都会被忽略

按引用传递：函数范围内对值的任何改变在函数外部也能反映出这些修改

优缺点：按值传递时，php 必须复制值。特别是对于大型的字符串和对象来说，这将会是一个代价很大的操作。按引用传递则不需要复制值，对于性能提高很有好处。

```
function test(&$a)
```

```
{
```

```
$a=$a+100;
```

```
}
```

```
$b=1;
```

```
echo $b;//输出 1
```

```
test($b); //这里$b 传递给函数的其实是$b 的变量内容所处的内存地址，通过在函数里改变$a 的值 就可以改变$b 的值了
```

```
echo "<br>";
```

```
echo $b;//输出101
```

## in, not in, exist, not exist 的作用和区别？

in 在什么中

Not in 不在什么中

Exists 存在

Not exists 不存在

## 面向对象中接口和抽象类的区别及应用场景？

- 1、有抽象方法的类叫做抽象类，抽象类中不一定有抽象方法，抽象方法必须使用 `abstract` 关键字定义。
- 2、接口中全部是抽象方法，方法不使用 `abstract` 定义。
- 3、当多个同类的类要设计一个上层，通常设计为抽象类，当多个异构的类要设计一个上层，通常设计为接口。

### echo(), print(), print\_r() 的区别?

`echo` 是 PHP 语句，`print` 和 `print_r` 是函数，语句没有返回值，函数可以有返回值(即使没有用)  
`Print()` 只能打印出简单类型变量的值(如 `int`, `string`)  
`print_r()` 可以打印出复杂类型变量的值(如数组，对象)  
`echo` 输出一个或者多个字符串

### mysql\_fetch\_row() 和 mysql\_fetch\_array 之间有什么区别?

`mysql_fetch_row()` 以索引数组的方式取查询的结果集，`mysql_fetch_array()` 以索引数组和关联数组两种方式取查询的结果集。

### 请描述出两点以上 XHTML 和 HTML 最显著的区别

- (1) XHTML 必须强制指定文档类型 `DocType`，HTML 不需要
- (2) XHTML 所有标签必须闭合，HTML 比较随意

### 33. HTTP 协议中 GET、POST 和 HEAD 的区别?

HEAD: 只请求页面的首部。  
 GET: 请求指定的页面信息，并返回实体主体。  
 POST: 请求服务器接受所指定的文档作为对所标识的 URI 的新的从属实体。

(1) HTTP 定义了与服务器交互的不同方法，最基本的方法是 GET 和 POST。事实上 GET 适用于多数请求，而保留 POST 仅用于更新站点。

(2) 在 FORM 提交的时候，如果不指定 Method，则默认为 GET 请求，Form 中提交的数据将会附加在 url 之后，以?分开与 url 分开。字母数字字符原样发送，但空格转换为“+”号，其它符号转换为%XX，其中 XX 为该符号以 16 进制表示的 ASCII (或 ISO Latin-1) 值。GET 请求提交的数据放置在 HTTP 请求协议头中，而 POST 提交的数据则放在实体数据中；GET 方式提交的数据最多只能有 1024 字节，而 POST 则没有此限制。

(3) GET 这个是浏览器用语向服务器请求最常用的方法。POST 这个方法也是用来传送数据的，但是与 GET 不同的是，使用 POST 的时候，数据不是附在 URI 后面传递的，而是要做为独立的行来传递，此时还必须发送一个 `Content_length` 标题，以标明数据长度，随后一个空白行，然后就是实际传送的数据。网页的表单通常是用 POST 来传送的。

### 表单中 get 与 post 提交方法的区别?

1. `get` 是把参数数据队列加到提交表单的 ACTION 属性所指的 URL 中，值和表单内各个字段一一对应，在 URL 中可以看到。`post` 是通过 HTTP post 机制，将表单内各个字段与其内容放置在 HTML HEADER 内一起传送到 ACTION 属性所指的 URL 地址。用户看不到这个过程。

2. 对于 `get` 方式，服务器端用 `Request.QueryString` 获取变量的值，对于 `post` 方式，服务器端用 `Request.Form` 获取提交的数据。

3. `get` 传送的数据量较小，不能大于 2KB。`post` 传送的数据量较大，一般被认为不受限制。但理论上，IIS4 中最大量为 80KB，IIS5 中为 100KB。  
`get` 安全性非常低，`post` 安全性较高。

### foo() 和 @foo() 之间有什么区别?(1分)

@foo() 控制错误输出

### 线性表和顺序表的区别?

- 1、线性表包括顺序表和链表。
- 2、顺序表中的元素的地址是连续的，链表里面节点的地址不是连续的，是通过指针连起来的。

### 什么是数据库索引，主键索引，唯一索引的区别，索引的缺点是什么?

索引用来快速地寻找那些具有特定值的记录。

主键索引和唯一索引的区别：主键是一种唯一性索引，但它必须指定为“PRIMARY KEY”，每个表只能有一个主键。唯一索引索引列的所有值都只能出现一次，即必须唯一。

索引的缺点：

- 1、创建索引和维护索引要耗费时间，这种时间随着数据量的增加而增加。
- 2、索引需要占用物理空间，除了数据表占数据空间之外，每一个索引还要占一定的物理空间，如果要建立聚簇索引，需要的空间就会更大。
- 3、当对表中的数据进行增加、删除、修改的时候，索引也要动态的维护，这样就降低了数据的维护速度。

### 浏览器 IE 和非 IE 浏览器的划分，区别是什么?

IE 浏览器指的是使用 IE 内核的浏览器，对一些 W3C 标准的网页代码的支持不是很好。

非 IE 浏览器指的是没有使用 IE 内核的浏览器，对 W3C 标准的网页代码有很好的支持。

### 数据库中的事务是什么?

事务 (transaction) 是作为一个单元的一组有序的数据库操作。如果组中的所有操作都成功，则认为事务成功，即使只有一个操作失败，事务也不成功。如果所有操作完成，事务则提交，其修改将作用于所有其他数据库进程。如果一个操作失败，则事务将回滚，该事务所有操作的影响都将取消。

### 解释:左连接,右连接,内连接,索引。

- 1、内连接仅选出两张表中互相匹配的记录。因此，这会导致有时我们需要的记录没有包含进来。内部连接是两个表中都必须有连接字段的对应值的记录，数据才能检索出来。
- 2、左连接和右连接都是外部连接，也就是区别于内部连接，它对不满足连接条件的行并不是象内部连接一样将数据完全过滤掉，而是保留一部分数据，行数不会减少。
- 3、左连接是只要左边表中有记录，数据就能检索出来，而右边有的记录必要在左边表中的记录才能被检索出来
- 4、右连接是只要右边表中有记录，数据就能检索出来；
- 5、全连接则会返回两个表中的所有记录

### 简述0000000中无限分类的实现原理。

- 1、数据库分类表的设计：id 字段、父 id 字段、类名称字段、path 字段
- 2、Path 字段存当前分类的路径，格式：0, 父类 id, 本身 id
- 3、查询分类：select \* from 分类表 order by path asc;
- 4、这样就查出来每个分类及其对应的子类。

### 能够使 html 和 php 分离开使用的模板?

Smarty、template、PHPLibTemplate、FastTemplate

### 使用那些工具进行版本控制?

Clear case、CVS、SVN、PVCS、Perforce、CCC、StarTeam、RCS

### 写出三种以上 MySQL 数据库存储引擎的名称 (提示: 不区分大小写)

MyISAM、InnoDB、BDB (Berkeley DB)、Merge、Memory (Heap)、Example、Federated、Archive、CSV、Blackhole、MaxDB 等等十几个引擎

### 什么是模板技术、能够使 HTML 和 PHP 分离使用的模板?

模板技术就是使程序的逻辑代码和界面分开的技术。  
能够使 HTML 和 PHP 分开的模板有: Smarty、Template、PHPLib Template、FastTemplate

### 实现中文字串截取无乱码的方法

Mb\_substr();

### 用 PHP 写出显示客户端 IP 与服务器 IP 的代码

获取客户端 IP: get\_env("REMOTE\_ADDR");  
获取服务器端 IP: \$\_SERVER["SERVER\_ADDR"];

### 数据库索引有几类, 分别是什么? 什么时候该用索引?

普通索引、主键索引、唯一索引  
并非所有的数据库都以相同的方式使用索引, 作为通用规则, 只有当经常查询列中的数据时才需要在表上创建索引。

### 写几个魔术方法并说明作用?

\_\_call() 当调用不存在的方法时会自动调用的方法  
\_\_autoload() 在实例化一个尚未被定义类是会调用次方法来加载类文件  
\_\_set() 当给未定义的变量赋值时会自动调用的方法  
\_\_get() 当获取未定义变量的值时会自动调用的方法  
\_\_construct() 构造方法, 实例化类时自动调用的方法  
\_\_destruct() 销毁对象时自动调用的方法  
\_\_unset() 当对一个未定义变量调用 unset() 时自动调用的方法  
\_\_isset() 当对一个未定义变量调用 isset() 方法时自动调用的方法  
\_\_clone() 克隆一个对象  
\_\_toString() 当输出一个对象时自动调用的方法

### \$\_REQUEST、\$\_POST、\$\_GET、\$\_COOKIE、\$\_SESSION、\$\_FILE 的意思是什么?

它们都是 PHP 预定义变量  
\$\_REQUEST 用来获取 post 或 get 方式提交的值  
\$\_POST 用来获取 post 方式提交的值  
\$\_GET 用来获取 get 方式提交的值  
\$\_COOKIE 用来获取 cookie 存储的值  
\$\_SESSION 用来获取 session 存储的值  
\$\_FILE 用来获取上传文件表单的值

### 数组中下标最好是什么类型的, 为什么?

数组的下标最好是数字类型的, 数字类型的处理速度快。

### ++i 和 i++ 哪一个效率高, 为什么?

++i 效率比 i++ 的效率更高, 因为 ++i 少了一个返回 i 的过程。

### magic\_quotes\_gpc()、magic\_quotes\_runtime() 的意思是什么?

Magic\_quotes\_gpc() 是 php 配置文件中的, 如果设置为 on 则会自动 POST, GET, COOKIE 中的字符串进行转义, 在 '之前加 \

Magic\_quotes\_runtime() 是 php 中的函数, 如果参数为 true 则会数据库取出来的单引号、双引号、反斜线自

动加上反斜杠进行转义。

### 框架中什么是单一入口和多入口, 单一入口的优缺点?

1、多入口就是通过访问不同的文件来完成用户请求。  
单一入口只 web 程序所有的请求都指向一个脚本文件的。

2、单一入口更容易控制权限, 方便对 http 请求可以进行安全性检查。

缺点: URL 看起来不那么美观, 特别是对搜索引擎来说不友好。

### 你对 Memcach 的理解, 优点有哪些?

Memcache 是一种缓存技术, 在一定的时间内将动态网页经过解析之后保存到文件, 下次访问时动态网页就直接调用这个文件, 而不必在重新访问数据库。使用 memcache 做缓存的好处是: 提高网站的访问速度, 减轻高并发时服务器的压力。

Memcache 的优点: 稳定、配置简单、多机分布式存储、速度快。

对关系型数据库而言, 索引是相当重要的概念, 请回答有关索引几个问题:

#### a) 索引的目的是什么?

- 1、快速访问数据表中的特定信息, 提高检索速度
- 2、创建唯一性索引, 保证数据库表中每一行数据的唯一性
- 3、加速表和表之间的连接
- 4、使用分组和排序子句进行数据检索时, 可以显著减少查询中分组和排序的时间

#### b) 索引对数据库系统的负面影响是什么?

负面影响: 创建索引和维护索引需要耗费时间, 这个时间随着数据量的增加而增加; 索引需要占用物理空间, 不光是表需要占用数据空间, 每个索引也需要占用物理空间; 当对表进行增、删、改的时候索引也要动态维护, 这样就降低了数据的维护速度。

#### c) 为数据表建立索引的原则有哪些?

- 1、在最频繁使用的、用以缩小查询范围的字段上建立索引
- 2、在平频繁使用的、需要排序的字段上建立索引

#### d) 什么情况下不宜建立索引?

- 1、对于查询中很少涉及的列或者重复值比较多的列, 不宜建立索引
- 2、对于一些特殊的数据类型, 不宜建立索引, 比如文本字段(text)等。

### web 应用中, 数据库的读取频率远高于写入频率, 如何优化 MySQL 而应对此种情景?

使用 memcache 缓存技术, 将动态数据缓存到文件, 访问动态页面时直接调用缓存文件, 而不必重新访问数据库, 这样就减少了查询数据库的次数。

如果网站的访问量很大, 可以把数据库读写服务器分开, 使用多态服务器去处理数据库查询, 使用较少的服务器去处理数据库的写入和修改。

### 如果是一个 Web 频繁访问的查询, 上题的查询如何优化?

使用 memcache 缓存技术, 在一定时间内将动态内容缓存到文件中, 访问动态页面时, 世界调用缓存文件而不必重新查询数据库。

### 数据库设计时, 常遇到的性能瓶颈有哪些, 常有的解决方案?

瓶颈主要有:

- 1、磁盘搜索 优化方法是：将数据分布在多个磁盘上
- 2、磁盘读/写 优化方法是：从多个磁盘并行读写。
- 3、CPU 周期 优化方法：扩充内存
- 4、内存带宽

### **请举例说明在你的开发过程中用什么方法来加快页面的加载速度**

要用到服务器资源时才打开，及时关闭服务器资源，数据库添加索引，页面可生成静态，图片等大文件单独服务器。使用代码优化工具。

### **描述一下大流量高并发量网站的解决方案**

- 1、确认服务器硬件是否足够支持当前的流量。
  - 2、使用 memcache 缓存技术，将动态数据缓存到内存中，动态网页直接调用这些文件，而不必在访问数据库。
  - 3、禁止外部的盗链。
  - 4、外部网站的图片或者文件盗链往往会带来大量的负载压力，因此应该严格限制外部对自身图片或者文件盗链，目前可以简单的通过 refer 来控制盗链，apache 自己就可以通过配置来禁止盗链。
  - 5、控制大文件的下载。
- 大文件的下载会占用很大的流量，对于非 SCSI 硬盘来说会消耗，使得网站响应能力下降。
- 6、使用不同的主机分流主要流量
  - 7、使用流量统计软件。
- 在网站上安装一个流量统计软件，可以即时知道哪些地方耗费了大量流量，哪些页面需要再进行优化。
- 8、分库分表。
  - 9、Sphinx 全文索引引擎。

### **对于大流量的网站，您采用什么样的方法来解决访问量问题？**

优化程序，优化数据库，如果程序和数据库已经最优化，使用以下解决方法：

- 1、确定当前服务器设备是否满足流量需求。
- 2、使用 Memcache 缓存技术，把动态内容缓存到文件中，动态网页直接调用这些文件，而不必再访问数据库。
- 3、禁止外部盗链，图片和文件外部盗链会给服务器带来大量的负载压力，可以通过 refer 来禁止外部盗链，或者使用 apache 来配置禁止盗链。
- 4、控制大文件的下载，大文件的下载对于非 SCSI 硬盘来说会占用大量的资源，导致服务器的响应能力下降。
- 5、使用不同的主机分流主要流量，使服务器均衡负载。
- 6、使用流量统计软件统计分析网站流量，可以知道哪些地方耗费了大量的流量，哪些页面需要再进行优化。

### **如何设计或配置 Mysql，才能达到高效使用的目的。**

- 1、数据库设计方面，设计结构良好的数据库，允许部分数据冗余。

选取最适用的字段属性，尽可能把字段设置为 NOTNULL，这样在查询的时候，数据库不用去比较 NULL 值。

- 2、系统架构设计方面，表散列，把海量数据散列到几个不同的表里面，集群，数据库查询和写入分开。写高效 sql 语句，以提高效率。

使用连接(join)来代替子查询

使用联合(union)来代替手动创建的临时表

所得皆必须，只从数据库取必须的数据。

必要的时候用不同的存储引擎，比如 Innodb 可以减少死锁，HEAP 可以提高一个数量级的查询速度。

使用事务

使用外键

使用索引

### **设定网站的用户数量在千万级，但是活跃用户的数量只有 1%，如何通过优化数据库提高活跃用户的访问速度？**

我们可以根据用户的活跃程度，把活跃的用户提取出来放到另外一张表里面，每次活跃的用户登陆的时候就直接到活跃用户表中进行查询，这样就提高了数据库的查询速度。

### **了解 XSS 攻击吗？如何防止？**

XSS 是跨站脚本攻击，首先是利用跨站脚本漏洞以一个特权模式去执行攻击者构造的脚本，然后利用不安全的 ActiveX 控件执行恶意的行为。

使用 htmlspecialchars() 函数对提交的内容进行过滤，使字符串里面的特殊符号实体化。

### **SQL 注入漏洞产生的原因？如何防止？**

SQL 注入产生的原因：程序开发过程中不注意规范书写 sql 语句和对特殊字符进行过滤，导致客户端可以通过全局变量 POST 和 GET 提交一些 sql 语句正常执行。

### **防止 SQL 注入：**

- 1、开启配置文件中的 magic\_quotes\_gpc 和 magic\_quotes\_runtime 设置
- 2、执行 sql 语句 // 使用 addslashes 进行 sql 语句转换
- 3、Sql 语句书写尽量不要省略小引号和单引号
- 4、过滤掉 sql 语句中的一些关键字：update、insert、delete、select、\*
- 5、提高数据库表和字段的命名技巧，对一些重要的字段根据程序的特点命名，取不易被猜到的。
- 6、Php 配置文件中设置 register\_globals 为 off，关闭全局变量注册
- 7、控制错误信息，不要再浏览器上输出错误信息，将错误信息写到日志文件中。

### **如何进行防 SQL 注入？**

- 1、过滤掉一些常见的数据库操作关键字：select, insert, update, delete, and, \* 等
- 或者通过系统函数：addslashes(需要被过滤的内容)来进行过滤。

- 2、在 PHP 配置文件中

Register\_globals=off; 设置为关闭状态 // 作用将注册全局变量关闭。

比如：接收 POST 表单的值使用 \$\_POST['user'], 如果将 register\_globals=on; 直接使用 \$user 可以接收表单的值。

- 3、SQL 语句书写的时候尽量不要省略小引号 (tab 键上面那个) 和单引号

- 4、提高数据库命名技巧，对于一些重要的字段根据程序的特点命名，取不易被猜到的

- 5、对于常用的方法加以封装，避免直接暴露 SQL 语句

- 6、开启 PHP 安全模式

Safe\_mode=on;

- 7、打开 magic\_quotes\_gpc 来防止 SQL 注入

Magic\_quotes\_gpc=off; 默认是关闭的，它打开后将自动把用户提交的 sql 语句的查询进行转换，把 ' 转为 \'

，这对防止 sql 注入有重大作用。

因此开启：magic\_quotes\_gpc=on;

- 8、控制错误信息

关闭错误提示信息，将错误信息写到系统日志。

- 9、使用 mysqli 或 pdo 预处理。

### **用 PHP 写出一个安全的用户登录系统需要注意哪些方面。**

- 1、密码要使用 MD5(密码+字符串)进行加密。
- 2、登录表单的名称不要跟数据库字段一样，以免暴露表字段。
- 3、用户表的表名、字段名、密码尽量用不容易被猜到的。
- 4、要使用验证码验证登陆，以防止暴力破解。

5、验证提交的数据是不是来自本网站。  
6、登录后台处理代码数据库部分使用预处理,做好过滤,防止 sql 注入。

### 如何实现多个线程安全的写入一个文件数据。

采用锁机制,当一个用户在对此文件进行读写操作时,将此文件锁定,操作完毕后解除锁定,在该用户进行读写操作过程中,其他用户不能操作此文件,需要排队等待。

### 什么是满二叉树?什么事完全二叉树?

满二叉树:除了叶子节点外的所有节点都有两个子节点。  
完全二叉树:每个节点最多有两个子节点,缺右不缺左,而且最多只能缺少一个叶子节点。

### 字符串转数组,数组转字符串,字符串截取,字符串替换,字符串查找的函数分别是什么?

implode(), explode(), substr(), str\_replace(), strpos, strrpos, strstr

### GD 库是做什么用的? (1分)

gd 库提供了一系列用来处理图片的 API,使用 GD 库可以处理图片,或者生成图片。

在网站上 GD 库通常用来生成缩略图或者用来对图片加水印或者对网站数据生成报表。

在网站上 GD 库通常用来生成缩略图或者用来对图片加水印或者对网站数据生成报表。

John

### 写出 Zend 框架的目录结构,简单说明目录作用?

application/ - 存放应用程序的目录,包括 MVC 系统、配置文件、服务以及引导程序 (Bootstrap.php)  
configs/ - 配置文件目录。

application/modules - 多模块目录结果例如 admin(后台前层), default(前台), shop(商城) 各个模块下都有自己的 controllers, modules, Views  
controllers/models/views/ - 控制器/模型/视图 目录。

application/cache - 存放缓存等数据

application/helpers/ - 这个是存放“行动助手”(action helper)的目录。这些助手类默认的命名空间是“Controller\_Helper\_”,如果是多模块那么命名空间将是“<Module>\_Controller\_Helper”。

Bootstrap.php - 这个是应用程序入口文件。这个类的主要工作是引导应用程序,注册并初始化组件

(component)。注意:不要在这个文件调用前置控制器(front controller)的 dispatch() 方法。

library/ - 存放类库的目录。第三方的类库和自己写的类库放在这里进行自动加载,但要注意要用自己独立的命名空间(建子目录)。

public/ - 存放公开文件的目录,也即网站的根目录,存放可以被用户访问的文件,例如 js、css 和图片等等。  
index.php 是应用程序的单一入口,其主要工作是建立 php 环境,引用 Bootstrap.php 来初始化,并调用前置控制器的 dispatch() 方法来分发请求。

### 谈谈对 mvc 的认识?

由模型(model), 视图(view), 控制器(controller)完成的应用程序

由模型发出要实现的功能到控制器,控制器接收组织功能传递给视图;

MVC 是一个设计模式,它强制性的使应用程序的输入、处理和输出分开。使用 MVC 应用程序被分成三个核心部件:模型、视图、控制器。它们各自处理自己的任务。视图是用户看到并与之交互的界面。

模型表示企业数据和业务规则。

控制器接受用户的输入并调用模型和视图去完成用户的

需求。

MVC 的优点:低耦合性、高重用性和可适用性、较低的生命周期成本、快速的部署、可维护性、可扩展性,有利于软件工程化管理

MVC 的缺点:没有明确的定义,完全理解 MVC 并不容易。不适合小型规模的应用程序。

### PHP 如何抛出和接收错误?

使用 try...catch,异常的代码放在 try 代码块内,如果没有触发异常,则代码继续执行,如果异常被触发,就会抛出一个异常。Catch 代码块捕获异常,并创建一个包含异常信息的对象。\$e->getMessage(), 输出错误的信息。

### PHP 的网站主要攻击方式有哪些?

- 1、命令注入(Command Injection)
- 2、eval 注入(Eval Injection)
- 3、客户端脚本攻击(Script Insertion)
- 4、跨网站脚本攻击(Cross Site Scripting, XSS)
- 5、SQL 注入攻击(SQL injection)
- 6、跨网站请求伪造攻击(Cross Site Request Forgeries, CSRF)
- 7、Session 会话劫持(Session Hijacking)
- 8、Session 固定攻击(Session Fixation)
- 9、HTTP 响应拆分攻击(HTTP Response Splitting)
- 10、文件上传漏洞(File Upload Attack)
- 11、目录穿越漏洞(Directory Traversal)
- 12、远程文件包含攻击(Remote Inclusion)
- 13、动态函数注入攻击(Dynamic Variable Evaluation)
- 14、URL 攻击(URL attack)
- 15、表单提交欺骗攻击(Spoofed Form Submissions)
- 16、HTTP 请求欺骗攻击(Spoofed HTTP Requests)

### 几个重要的 php.ini 选项

#### Register Globals

php>=4.2.0, php.ini 的 register\_globals 选项的默认值预设为 Off, 当 register\_globals 的设定为 On 时, 程序可以接收来自服务器的各种环境变量, 包括表单提交的变量, 而且由于 PHP 不必事先初始化变量的值, 从而导致很大的安全隐患。

#### safe\_mode

安全模式, PHP 用来限制文档的存取、限制环境变量的存取, 控制外部程序的执行。启用安全模式必须设置 php.ini 中的 safe\_mode = On

#### 1、限制文件存取

safe\_mode\_include\_dir = “/path1:/path2:/path3”  
不同的文件夹用冒号隔开

#### 2、限制环境变量的存取

safe\_mode\_allowed\_env\_vars = string  
指定 PHP 程序可以改变的环境变量的前缀, 如:safe\_mode\_allowed\_env\_vars = PHP\_, 当这个选项的值为空时, 那么 php 可以改变任何环境变量  
safe\_mode\_protected\_env\_vars = string  
用来指定 php 程序不可改变的环境变量的前缀

#### 3、限制外部程序的执行

safe\_mode\_exec\_dir = string  
此选项指定的文件夹路径影响 system、exec、popen、passthru, 不影响 shell\_exec 和 “`”。  
disable\_functions = string  
不同的函数名称用逗号隔开, 此选项不受安全模式影响 magic quotes

用来让 php 程序的输入信息自动转义，所有的单引号（“'”），双引号（“””），反斜杠（“\”）和空字符（NULL），都自动被加上反斜杠进行转义  
 magic\_quotes\_gpc = On 用来设置 magic quotes 为 On，它会影响 HTTP 请求的数据（GET、POST、Cookies）  
 程序员也可以使用 addslashes 来转义提交的 HTTP 请求数据，或者用 stripslashes 来删除转义

### 使用过哪些 PHP 框架。试描述其优缺点。

1、BroPHP 框架：

优点：轻量级学习型框架，配置简单，操作简单，容易上手，提供了比较全面的文档。

缺点：

2、ThinkPHP 框架：

优点：功能比较全面，配置相对比较简单，操作相对比较简单，有很多使用示例程序。

缺点：提供的操作方法太多，新手不知如何选择，文档不够全面。

### php 中，模板引擎的目的是什么？你用过哪些模板引擎？

使用模板引擎的目的是使程序的逻辑代码和 html 界面代码分离开，是程序的结构更清晰。

使用过的模板引擎：Smarty、ThinkPHP 的 ThinkTemplate

### 7. 如何在页面之间传递变量(至少两种方式)？

GET, POST, COOKIE, SESSION

8.

### 你所知道的 php 数组相关的函数？

array() —— 创建数组

array\_combine() —— 通过合并两个数组来创建一个新数组

range() —— 创建并返回一个包含指定范围的元素的数组

compact() —— 建立一个数组

array\_chunk() —— 将一个数组分割成多个

array\_merge() —— 把两个或多个数组合并成一个数组

array\_slice() —— 在数组中根据条件取出一段值

array\_diff() —— 返回两个数组的差集数组

array\_intersect() —— 计算数组的交集

array\_search() —— 在数组中搜索给定的值

array\_splice() —— 移除数组的一部分且替代它

array\_key\_exists() —— 判断某个数组中是否存在指定的 key

shuffle() —— 把数组中的元素按随机顺序重新排列

array\_flip() —— 交换数组中的键和值

array\_reverse() —— 将原数组中的元素顺序翻转，创建新的数组并返回

array\_unique() —— 移除数组中重复的值

### 你所知道的 PHP 的一些技术 (smarty 等)?

Smarty, jquery, ajax, memcache, div+css, js, mysqli, pdo, svn, thinkphp, brophp, yii

### 你所熟悉的 PHP 论坛系统 有哪些？

Discuz

### 你所熟悉的 PHP 商城系统 有哪些？

Ecshop

### 你所熟悉的 PHP 开发框架 有哪些？

Brophp, thinkphp

### 你所知道的设计模式有哪些？

工厂模式、策略模式、单元素模式、观察者模式、命令链模式

### 说说你对代码管理的了解？ 常使用那些代码版本控制软件？

通常一个项目是由一个团队去开发，每个人将自己写好的代码提交到版本服务器，由项目负责人按照版本进行管理，方便版本的控制，提高开发效率，保证需要时可以回到旧版本。

常用的版本控制器：SVN

### 说说你对 SVN 的了解？优缺点？

SVN 是一种版本控制器，程序员开发的代码递交到版本服务器进行集中管理。

SVN 的优点：代码进行集中管理，版本控制容易，操作比较简单，权限控制方便。

缺点：不能随意修改服务器项目文件夹。

### 怎么找到 PHP.ini 的路径？

一般都在 php 的安装目录下，或者 window 系统的 windows 目录下。

### PHP 加速模式/扩展？ PHP 调试模式/工具？

Zend Optimizer 加速扩展

调试工具：xdebug

### 你常用到的 mysql 命令？

Show databases

Show tables

Insert into 表名() values()

Update 表名 set 字段=值 where ...

Delete from 表名 where ...

Select \* from 表名 where 条件 order by ...

Desc/asc limit ... Group by ... Having ...

### 进入 mysql 管理命令行的命令？

Mysql -uroot -p 回车 密码

### show databases; 这个命令的作用？

显示当前 mysql 服务器中有哪些数据库

### show create database mysql; 这个命令的作用？

显示创建数据库的 sql 语句

### show create table user; 这个命令的作用？

显示创建表的 sql 语句

### desc user; 这个命令的作用？

查询 user 表的结构

### explain select \* from user; 这个命令的作用？

获取 select 相关信息

### show processlist; 这个命令的作用？

显示哪些线程正在运行

### SHOW VARIABLES; 这个命令的作用？

显示系统变量和值

### SHOW VARIABLES like ' %conn%' ; 这个命令的作用？

显示系统变量名包含 conn 的值

### LEFT JOIN 写一个 SQL 语句？

SELECT A. id, A. class FROM A LEFT JOIN B ON A. cid=B. id

### 请写出 5 个常用的 unix 操作命令。

1、Ls -l

2、tar -zxvf

3、Touch

- 4、Cat
- 5、Mkdir
- 6、rm

**请写出以下 5 个 UNIX 系统配置文件/服务的作用。**

/etc/passwd  
/etc/hosts  
/etc/fstab  
Crontab  
sshd

- 1、/etc/passwd：用户信息文件，其中包含用户名、用户是否有密码、用户 ID 及用户所属组
- 2、/etc/hosts：主机名数据库，包 IP 地址、主机名或域名、别名等信息，在此文件中添加记录可绕过 DNS
- 3、/etc/fstab 保存了系统所有分区信息的配置文件。
- 4、Crontab 计划任务命令
- 5、Sshd 远程连接管理命令

**请写出 5 个常用的 SQL 操作关键字。**

Select ,update, delete, insert into,  
 create ,group by ,order by

**请列出 5 个常用的 PHP 操作 MySQL 的函数**

- 1、Mysql\_connect()
- 2、Mysql\_select\_db()
- 3、Mysql\_query()
- 4、Mysql\_fetch\_row()
- 5、Mysql\_close()

**请列出 4 个常用的 HTML 标签名**

<table></table>  
 <a></a>  
 <form></form>  
 <div></div>

**怎么找到数据库的配置文件路径？**

在数据库安装目录下，my.ini

**什么是时间戳？如何取得当前时间戳？**

时间戳是从1970年1月1日 00:00:00到指定日期的秒数。  
 获取当前时间戳：time()

**有一个网页地址，比如 PHP 开发资源网主页：**

**http://www.phpres.com/index.html，如何得到它的内容？**

获取网页内容：

```
$url="http://www.phpres.com/index.html";
$str=file_get_contents($url);
```

**简述如何得到当前执行脚本路径，包括所得到参数**

```
$_SERVER['DOCUMENT_ROOT']. $_SERVER['REQUEST_URI']
;取得当前页面的完整路径和参数。
取得参数：$_SERVER['QUERY_STRING'];
```

**如何用 php 的环境变量得到一个网页地址的内容？ip 地址又要怎样得到？**

```
echo $_SERVER ["PHP_SELF"];
echo $_SERVER ["SERVER_ADDR"];
```

**求两个日期的差数，例如2007-2-5 ~ 2007-3-6 的日期差数**

```
$begin=strtotime("2007-2-5");
$end=strtotime("2007-3-6");
echo ($end-$begin)/(24*3600);
```

**给一个变量赋值为0123，但是输出该变量的值总是为其他数字，请问这是什么问题？**

PHP 解释器会把以0开始的数字当做是八进制的，所以它的值会变成八进制的。

**PHP 里面如何定义一个常量(constant)?**

通过 define() 指令，例如：define("MYCONSTANT", 100)

**如何对一个变量进行值传递？**

可以像 C++那样，在变量的前面加上&，例如：\$a =&\$b

**在 PHP 里面可以对 string “10” 和整型 11进行比较吗？**

可以，PHP 内部实现会把所有东西转换成整型，所以数字10和11可以进行比较

**在什么情况下需要用 endif 来结束条件语句？**

当 if 语句后面跟的是冒号”：”而不是大括号”{”时，需要用 endif 来结束条件语句

**优化 MYSQL 数据库的方法。(4分，多写多得)**

- 1、选取最适用的字段属性，尽可能减少定义字段长度，尽量把字段设置 NOT NULL，例如’省份，性别’，最好设置为 ENUM
- 2、使用连接 (JOIN) 来代替子查询：
- 3、使用联合 (UNION) 来代替手动创建的临时表
- 4、事务处理：
- 5、锁定表，优化事务处理：
- 6、使用外键，优化锁定表
- 7、建立索引：

a. 格式：

(普通索引)->

创建：CREATE INDEX <索引名> ON tablename (索引字段)

修改：ALTER TABLE tablename ADD INDEX [索引名] (索引字段)

创表指定索引：CREATE TABLE

tablename([...], INDEX[索引名] (索引字段))

(唯一索引)->

创建：CREATE UNIQUE <索引名> ON tablename (索引字段)

修改：ALTER TABLE tablename ADD UNIQUE [索引名] (索引字段)

创表指定索引：CREATE TABLE

tablename([...], UNIQUE[索引名] (索引字段))

(主键)->

它是唯一索引，一般在创建表是建立，格式为：

CREATA TABLE tablename ([...], PRIMARY KEY[索引字段])

8、优化查询语句

**MYSQL 取得当前时间的函数是？，格式化日期的函数是(2分)**

now(), date()

**实现中文字串截取无乱码的方法。(3分)**

```
function GBsubstr($string, $start, $length) {
    if(strlen($string)>$length) {
        $str=null;
        $len=$start+$length;
        for($i=$start;$i<$len;$i++) {
            if(ord(substr($string,$i,1))>0xa0) {
                $str.=substr($string,$i,2);
                $i++;
            }else{
                $str.=substr($string,$i,1);
            }
        }
    }
}
```

```

    }
    return $str.'...';
} else {
    return $string;
}
}

```

分别指出 `php.ini` 中 `magic_quotes_gpc`、`magic_quotes_runtime` 两项参数的作用。

`Magic_quotes_gpc` 的作用是在 POST、GET、COOKIE 数据上使用 `addslashes()` 自动转义。

`Magic_quotes_runtime` 参数的作用是设置状态，当状态为0时则关闭自动转义，设置为1则自动转义，将数据库中取出来的单引号、双引号、反斜线这些字符加上反斜杠转义。

用面向对象来实现 A 对象继承 B 和 C 对象

```

Interface B{ ... }
Interface C{ ... }
Class A implements B,C{ ... }

```

写出 Smarty 模板引擎中你最常用的关键词

```

Assign  Display  Foreach
Section  Loop      Item
$smartyy  Now      Const    get

```

如何快速下载一个远程 http 服务器上的图片文件到本地？

```

$file=" ";
$fp=fopen($file,'rb');
$img=fread($fp,10000);
$dir=" ./" ;
$local=fopen($dir.' /' .basename($file),'w');
Fwrite($local,$img);

```

一个字节占多少 bit ? 一个 IPv4 地址占几个字节？一个 IPv6 地址呢？

一个字节占8bit，一个 IPV4 占用4字节，一个 IPV6 占用16字节。

M ADSL 宽带连接，理想情况下，最大下载速度是多少 KB/s ?

256KB/s

**Fatal error: Call to undefined method ge\_user() in /website/index.php on line 39**

调用了未定义的方法 `ge_user()`，检查程序中有没有定义此方法

**Fatal error: Class 'client' not found in /website/index.php on line 173**

类 `client` 没有找到，检查文件中有没有 `client` 类，或者有没有包含 `client` 类文件

**Warning: Cannot modify header information - headers already sent by (output started at /website/index.php:1) in /website/index.php on line 3**

提示文件前面有输出，检查是否有输出，或者编码

**Warning: session\_start(): open(/website/tmp/sess\_47e067121facf033785f9alc b16d243b, 0 RDWR) failed: No such file or directory (2) in /website/index.php on line 10**

没有找到文件或目录，检查文件是否存在

**Parse error: syntax error, unexpected T\_STRING in /website/index.php on line 18**

18行语法错误，检查语法

**Warning: fopen(welcome.txt) [function.fopen]: failed to open stream: No such file or directory in /website/index.php on line 2**

没有找到 `welcome.txt` 文件，检查文件是否存在  
抓取远程图片到本地，你会用什么函数？

`fsockopen`, `A`

用 PHP 打印出前一天的时间，打印格式是2007年5月10日22:21:21

```

Echo date('Y-m-d H:i:s',strtotime(' -1 day' ));

```

//正在浏览当前页面用户的 IP 地址:127.0.0.1

```

echo $_SERVER["REMOTE_ADDR"]. " <br />";

```

//查询 (query) 的字符串 (URL 中第一个问号 ? 之后的内容):id=1&bi=2

```

echo $_SERVER["QUERY_STRING"]. " <br />";
//当前运行脚本所在的文档根目录:d:\inetpub\wwwroot
echo $_SERVER["DOCUMENT_ROOT"]. " <br />";

```

在 PHP 中，`heredoc` 是一种特殊的字符串，它的结束标志必须?(1分)

`heredoc` 的语法是用 " <<<" 加上自己定义成对的标签，在标签范围内的文字视为一个字符串

例子:

```

$str = << my name is Jiang Qihui!
SHOW;

```

如何声明一个名为 "myclass" 的没有方法和属性的类?(1分)

```

class myclass{ }

```

如何实例化一个名为 "myclass" 的对象?

```

new myclass()

```

你如何访问和设置一个类的属性?(2分)

```

$object = new myclass();
$newstr = $object->test;
$object->test = "info";

```

指出一些在 PHP 输入一段 HTML 代码的办法。(1分)

```

echo "aaa";

```

下面哪个函数可以打开一个文件，以对文件进行读和写操作?[ c ]

(a) `fget()` (b) `file_open()` (c) `fopen()` (d) `open_file()`

简述 Linux 下安装 PHP 的过程?

安装软件之前先安装编译工具 `gcc`、`gcc-c++`  
拷贝源码包，解包解压缩

`Cd /lamp/php` 进入 `php` 目录

`./configure -prefix=/usr/local/php -with-config-file-path=/usr/local/php/etc` 指定安装目录和配置文件目录

`Make` 编译

`Make install` 安装

简述 Linux 下安装 Mysql 的过程?

`Groupadd mysql` 添加一个用户组 `mysql`

`Useradd -g mysql mysql` 添加一个 `mysql` 用户指定分组为 `mysql`





```

    if($r > 1){
        $right = QuickSort($right);
    }
    for($i=0;$i<$r;$i++){
        $new_arr[] = $right[$i];
    }
    return $new_arr;
}

```

**使用 PHP 描述顺序查找和二分查找（也叫做折半查找）算法，顺序查找必须考虑效率，对象可以是一个有序数组**

```

//二分查找（数组里查找某个元素）
function bin_sch($array, $low, $high, $k){
    if ($low <= $high){
        $mid = intval(($low+$high)/2);
        if ($array[$mid] == $k){
            return $mid;
        }elseif ($k < $array[$mid]){
            return bin_sch($array, $low, $mid-1, $k);
        }else{
            return bin_sch($array, $mid+1, $high, $k);
        }
    }
    return -1;
}

//顺序查找（数组里查找某个元素）
function seq_sch($array, $n, $k){
    $array[$n] = $k;
    for($i=0; $i<$n; $i++){
        if($array[$i]==$k){
            break;
        }
    }
    if ($i<$n){
        return $i;
    }else{
        return -1;
    }
}

```

**写一个二维数组排序算法函数，能够具有通用性，可以调用 php 内置函数**

//二维数组排序， \$arr 是数据，\$keys 是排序的键值，\$order 是排序规则，1是升序，0是降序

```

function array_sort($arr, $keys, $order=0) {
    if (!is_array($arr)) {
        return false;
    }
    $keyvalue = array();
    foreach($arr as $key => $val) {
        $keyvalue[$key] = $val[$keys];
    }
    if($order == 0){
        asort($keyvalue);
    }else {
        arsort($keyvalue);
    }
    reset($keyvalue);
    foreach($keyvalue as $key => $vals) {
        $keysort[$key] = $key;
    }
    $new_array = array();

```

```

    foreach($keysort as $key => $val) {
        $new_array[$key] = $arr[$val];
    }
    return $new_array;
}

```

**请以空格作为间隔，拆分字符串' Apple Orange Banana Strawberry'，组成数组\$fruit，**

**\* 数组中所有元素都用小写字母，并按照字母先后次序排序**

```

class sort {
    private $str;
    public function __construct($str) {
        $this->str=strtolower($str);
    }
    private function explodes() {
        if(empty($this->str)) return array();
        $arr=explode(" ", $this->str);
        return is_array($arr)?$arr:array($arr);
    }
    public function sort() {
        $explode=$this->explodes();
        sort($explode);
        return $explode;
    }
}

$str=' Apple Orange Banana Strawberry';
$sortob=new sort($str);
var_dump($sortob->sort());

```

**对于用户输入一串字符串\$string，要求\$string中只能包含大于0的数字和英文逗号，请用正则 表达式验证，对于不符合要求的\$string 返回出错信息**

```

class regx {
    public static function check($str) {
        if(preg_match("/^([1-9,])+$/", $str)) {
            return true;
        }
        return false;
    }
}

$str="12345,6";
if(regx::check($str)) {
    echo "suc";
} else {
    echo "fail";
}

```

**请写一段程序，在服务器创建一个文件 fruit.dat，将试题3中得到的数组写入到改文件中，然后写一段程序从文件中读取并还原数组@author zhuwenqiong**

```

class sort {
    private $str;
    public function __construct($str) {
        $this->str=strtolower($str);
    }
    private function explodes() {
        if(empty($this->str)) return array();
        $arr=explode(" ", $this->str);
        return is_array($arr)?$arr:array($arr);
    }
    public function sort() {
        $explode=$this->explodes();
        sort($explode);
        return $explode;
    }
}

```

```

    }
}
class file {
    private $sort=null;
    private $filepath;
    public function __construct($arrobj,$path) {
        $this->sort=$arrobj;
        $this->filepath=$path;
    }
    private function getresource($filename,$mode)
    {
        return
        fopen($this->filepath.$filename,$mode);
    }
    private function closeresource($resource) {
        fclose($resource);
    }
    public function savefile($filename) {
        $arr=$this->sort->sort();

        $fopen=$this->getresource($filename,"a+");
        if(!$fopen){
            echo "文件打开失败! ";
            exit;
        }
        var_dump($arr);
        foreach($arr as $key=>$value) {
            fwrite($fopen,$value."\n");
        }
        $this->closeresource($fopen);
    }
    public function readfile($filename) {
        $this->savefile($filename);

        $fopen=$this->getresource($filename,"r");
        if(!$fopen){
            echo "文件打开失败! ";exit;
        }
        $arr=array();
        while(!feof($fopen)) {
            $get=fgets($fopen);
            if(!empty($get))

            $arr[]=str_replace("\n","", $get);
        }
        $this->closeresource($fopen);
        return $arr;
    }
}
$file=new file(new sort('Apple Orange Banana
Strawberry'),"E:\");
$arr=$file->readfile("fruit.dat");
var_dump($arr);

```

#### 单例模式，创建 mysqli 数据库链接的单例对象

```

class Db {
    private static $instance;
    public $handle;
    Private function
    __construct($host,$username,$password,$dbname) {
        $this->handle=NULL;

        $this->getcon($host,$username,$password,$db
name);
    }
}

```

```

    public static function getBb() {
        self::$instance=new Db();
        return self::$instance;
    }
    private function
    getcon($host,$username,$password,$dbname) {
        if($this->handle!=NULL){
            return true;
        }

        $this->handle=mysqli_connect($host,$usernam
e,$password,$dbname);
    }
}

```

#### PHP session 扩展默认将 session 数据储存在哪里? D

- A) SQLite Database
- B) MySQL Database
- C) Shared Memory
- D) File System
- E) Session Server

#### 如果你想要自动加载类，下面哪种函数声明是正确的 C

- A) function autoload(\$class\_name)
- B) function \_\_autoload(\$class\_name, \$file)
- C) function \_autoload(\$class\_name)
- D) function \_autoload(\$class\_name)
- E) function autoload(\$class\_name, \$file)

#### PHP 程序使用 utf-8 编码，以下程序输出结果是什么? B

```

<?php
$str = ' hello 你好世界' ;
echo strlen($str);
?>
A) 9      B) 13      C) 18      D) 17

```

#### 设计一个页面(4个 div 第一个 div 宽960px 居中;第 2-4个 div 3等分960px;)

```

<style>
    Body{ Text-align:center; Margin:0;
    Padding:0; }
    #box{ Width:960px; Margin:0 auto; }
    .small{ Width:320px; Float:left; }
</style>
<div id=' box' >
    <div class=' small' ></div>
    <div class=' small' ></div>
    <div class=' small' ></div>
</div>

```

#### 简述如何得到当前执行脚本路径，包括所得到参数。(2分)

```

$script_name = basename(__file__);
print_r($script_name);

```

#### js 中网页前进和后退的代码

```

前进: history.forward();=history.go(1);
后退: history.back();=history.go(-1);

```

#### JS 表单弹出对话框函数是?获得输入焦点函数是? (2分)

弹出对话框: alert(), prompt(), confirm()  
获得输入焦点 focus()

#### JS 的转向函数是?怎么引入一个外部 JS 文件?(2分)

```

window.location.href,
<script type="text/javascript"

```

```
src="hello.js"></script>
```

用 javascript 取得一个 input 的值? 取得一个 input 的属性?

```
document.getElementById( 'name' ).value;
document.getElementById( 'name' ).type;
```

用 Jquery 取得一个 input 的值? 取得一个 input 的属性?

```
$( "input[name='aa'] " ).val();
$( "input[name='aa'] " ).attr( 'name' );
```

写一个简单的 jquery 显示隐藏代码?

```
$("#aa").hide();
$("#aa").show();
```

javascript 能否定义二维数组, 如果不能你如何解决?

javascript 不支持二维数组定义, 可以用 arr[0] = new array() 来解决

下列不属于 Javascript 语法关键/保留字的是 B

- A. var
- B. \$
- C. function
- D. while

请使用 JavaScript 写出三种产生一个 Image 标签的方法 (提示: 从方法、对象、HTML 角度考虑)

- (1) var img = new Image();
- (2) var img = document.createElement( "image" );
- (3) img.innerHTML = ""

假设 a.html 和 b.html 在同一个文件夹下面, 用 javascript 实现当打开 a.html 五秒钟后, 自动跳转到 b.html。

```
<script>
function go2b() {
    window.location = "b.html";
    window.close();
}
setTimeout( "go2b()", 5000 ); //5秒钟后自动执行
go2b()
</script>
```

请您写一段 ajax 提交的 js 代码, 或者写出 ajax 提交的过程逻辑。

```
var xmlhttp;
if(window.XMLHttpRequest){
    xmlhttp=new XMLHttpRequest();
}else if(window.ActiveXObject){
    xmlhttp=new ActiveXObject( 'Microsoft.XMLHTTP' );
}
xmlhttp.open( 'GET' , '1.php?aa=name' , true);
xmlhttp.onreadystatechange=function() {
    if(xmlhttp.readyState==4){
        if(xmlhttp.status==200){
            var text=xmlhttp.responseText;
        }
    }
}
xmlhttp.send(null);
如下 user 表结构
```

名称	类型	说明
Uid	Int unsigned	主键
Name	Varchar(20)	

```
Age Tinyint unsigned
需求:
```

1 增加一个字段性别 sex, 写出修改语句  
Alert table user add sex enum( ' 0' , ' 1' );

查询出年龄介于20岁到30岁之间的用户

```
Select * from user where age>20 and age<30
```

写一个函数, 算出两个文件的相对路径

如 \$a = ' /a/b/c/d/e. php' ;  
\$b = ' /a/b/12/34/c. php' ;

计算出 \$b 相对于 \$a 的相对路径应该是

http://www.cnblogs.com/c/d 将() 加上  
\$a=" http://www.cnblogs.com/a/b/c/d/e. php" ;  
\$b=" http://www.cnblogs.com/a/b/12/34/c. php" ;

```
$ainfo=parse_url($a);
$binfo=parse_url($b);
$apath=ltrim($ainfo['path'],'/');
$bpath=ltrim($binfo['path'],'/');
$arr=explode('/', $apath);
$brr=explode('/', $bpath);
```

```
$count=count($arr);
for($i=0;$i<$count;$i++){
    if($arr[$i]==$brr[$i]){
        $brr[$i]='..';
    }else{
        break;
    }
}
$cha=implode('/', $brr);
print_r($cha);
```

写一个函数, 能够遍历一个文件夹下的所有文件和子文件夹。

```
function my_scandir($dir){
    $files = array();
    if ( $handle = opendir($dir) ){
        while ( ($file = readdir($handle)) !==
false ) {
            if ( $file != ".." && $file != "." )
            {
                if ( is_dir($dir . "/" . $file) )
                {
                    $files[$file] =
scandir($dir . "/" . $file);
                }else {
                    $files[] = $file;
                }
            }
        }
        closedir($handle);
        return $files;
    }
}
```

Linux 的目录进行遍历, 编写 shell 脚本

```
#!/bin/bash
Tree
```

指出以下代码片段中的 SQL 注入漏洞以及解决方法 (magic quotes gpc = off)

```
mysql_query( "select id,title from content where
catid=' ${_GET[catid]}' and title like "
```

```

%$_GET[keywords]%' ", $link);
注入漏洞主要存在用户提交的数据上, 这里的注入漏洞
主要是$_GET[catid]和$_GET[keyword]
解决注入漏洞:
$_GET[catid]=intval($_GET[catid]);
$sql=" select id,title from content where catid='
{$_GET[catid]}' and title like '
%$_GET[keywords]%' ";
$sql=addslashes($sql);
Mysql_query($sql);

```

### 写出以下 php 代码的运行结果:

```

<?php
function foo($i) {
    $i++;
    echo $i ;
}
function bar(&$i) {
}
$i = 10 ;
echo $i++ , ++$i;    输出: 10,12
foo($i); 输出: 13
bar($i);    输出: 无输出

```

在 HTTP 1.0 中, 状态码 401 的含义是未授权; 如果返回“找不到文件”的提示, 则可用 header 函数, 其语句为 header(“HTTP/1.0 404 Not Found”); 401 表示未授权; header(“HTTP/1.0 404 Not Found”);

### 把 John 新增到 users 阵列?

```
$users[] = 'john'; array_push($users, 'john');
```

### 在 PHP 中 error\_reporting 这个函数有什么作用?

error\_reporting() 设置 PHP 的报错级别并返回当前级别。

### 如何修改 SESSION 的生存时间(1分).

方法1: 将 php.ini 中的 session.gc\_maxlifetime 设置为 9999 重启 apache  
 方法2: \$savePath = “./session\_save\_dir/”;  
 \$lifeTime = 小时 \* 秒;  
 session\_save\_path(\$savePath);  
 session\_set\_cookie\_params(\$lifeTime);  
 session\_start();  
 方法3:  
 setcookie() and  
 session\_set\_cookie\_params(\$lifeTime);

### 有一个网页地址, 比如 PHP 开发资源网主页:

<http://www.phpres.com/index.html>, 如何得到它的内容?(1分)

方法1 (对于 PHP5 及更高版本):  
 \$readcontents =  
 fopen(“http://www.phpres.com/index.html”, “rb”);  
 \$contents = stream\_get\_contents(\$readcontents);  
 fclose(\$readcontents);  
 echo \$contents;  
 方法2:  
 echo  
 file\_get\_contents(“http://www.phpres.com/index.html”);

### 写一个函数, 尽可能高效的, 从一个标准 url 里取出文件的扩展名

例如: <http://www.sina.com.cn/abc/de/fg.php?id=1>  
 需要取出 php 或 .php

答案1:

```

function getExt($url) {
    $arr = parse_url($url);
    $file = basename($arr['path']);
    $ext = explode(“.”, $file);
    return $ext[1];
}

```

答案2:

```

function getExt($url) {
    $url = basename($url);
    $pos1 = strpos($url, “.”);
    $pos2 = strpos($url, “?”);
    if(strpos($url, “?”)) {
        Return substr($url, $pos1 + 1, $pos2 - $pos1 - 1);
    } else {
        return substr($url, $pos1);
    }
}

```

### 使用五种以上方式获取一个文件的扩展名

要求: dir/upload.image.jpg, 找出 .jpg 或者

jpg ,  
 必须使用 PHP 自带的处理函数进行处理, 方法不能明显重复, 可以封装成函数  
 get\_ext1(\$file\_name),  
 get\_ext2(\$file\_name)  
 function get\_ext1(\$file\_name) {  
 return strrchr(\$file\_name, ‘.’);  
 }  
 function get\_ext2(\$file\_name) {  
 return substr(\$file\_name, strpos(\$file\_name, ‘.’));  
 }  
 function get\_ext3(\$file\_name) {  
 return array\_pop(explode(‘.’, \$file\_name));  
 }  
 function get\_ext4(\$file\_name) {  
 \$p = pathinfo(\$file\_name);  
 return \$p['extension'];  
 }  
 function get\_ext5(\$file\_name) {  
 return strrev(substr(strrev(\$file\_name), 0, strpos(strrev(\$file\_name), ‘.’)));  
 }

```

<?php
$str1 = null;
$str2 = false;
echo $str1==$str2 ? ‘相等’ : ‘不相等’;
$str3 = ”;
$str4 = 0;
echo $str3==$str4 ? ‘相等’ : ‘不相等’;
$str5 = 0;
$str6 = ‘0’;
echo $str5==$str6 ? ‘相等’ : ‘不相等’;
??
相等 相等 不相等

```

### 求两个日期的差数, 例如2007-2-5 ~ 2007-3-6 的日期差数

方法一:

```

<?php
class Dtime{

```

```
function get_days($date1, $date2){
    $time1 = strtotime($date1);
    $time2 = strtotime($date2);
    return ($time2-$time1)/86400;
}
}
$Dtime = new Dtime;
echo $Dtime->get_days('2007-2-5', '2007-3-6');
?>
方法二:
<?php
$temp = explode('-', '2007-2-5');
$time1 = mktime(0, 0, 0, $temp[1], $temp[2],
$temp[0]);
$temp = explode('-', '2007-3-6');
$time2 = mktime(0, 0, 0, $temp[1], $temp[2],
$temp[0]);
echo ($time2-$time1)/86400;
方法三: echo abs(strtotime("2007-2-1")-strtotime("2007-3-1"))/60/60/24 计算时间差
```

请写一个函数，实现以下功能：

字符串“open\_door”转换成“OpenDoor”、“make\_by\_id”转换成“MakeById”。

方法:

```
function str_explode($str){
    $str_arr=explode("_",$str);$str_implode=implode(
    " ",$str_arr); $str_implode=implode
    (" ",explode(" ",ucwords($str_implode)));
    return $str_implode;
}
$str=explode(str_explode("make_by_id");print_r($s
trexplode);
方法二: $str=" make_by_id!";
$expStr=explode(" ",$str);
for($i=0;$i<count($expStr);$i++){
    echo ucwords($expStr[$i]);
}
方法三: echo
str_replace(' ','',ucwords(str_replace('_',',',
'','open_door')));
```

一个表中的Id有多个记录，把所有这个id的记录查出来，并显示共有多少条记录数，用SQL语句及视图、存储过程分别实现。

```
DELIMITER //
create procedure proc_countNum(in columnId int,out
rowsNo int)
begin
select count(*) into rowsNo from member where
member_id=columnId;
end
call proc_countNum(1,@no);
select @no;
方法: 视图:
create view v_countNum as select member_id,count(*)
as countNum from member group by
member_id
select countNum from v_countNum where member_id=1
echo count("abc"); 输出什么?
```

答案:1

count — 计算数组中的单元数目或对象中的属性个数  
int count (mixed \$var [, int \$mode]) , 如果 var 不是数组类型或者实现了 Countable 接口的对象,将返回 1, 有一个例外, 如果 var 是 NULL 则结果是 0。

对于对象, 如果安装了 SPL, 可以通过实现 Countable 接口来调用 count()。该接口只有一个方法 count(), 此方法返回 count() 函数的返回值。

有一个一维数组，里面存储整形数据，请写一个函数，将他们按从大到小的顺序排列。要求执行效率高。并说明如何改善执行效率。(该函数必须自己实现，不能使用php函数)

```
<?php
function BubbleSort(&$arr){
    $cnt=count($arr);
    $flag=1;
    for($i=0;$i<$cnt;$i++){
        if($flag==0){
            return;
        }
        $flag=0;
        for($j=0;$j<$cnt-$i-1;$j++){
            if($arr[$j]>$arr[$j+1]){
                $tmp=$arr[$j];
                $arr[$j]=$arr[$j+1];
                $arr[$j+1]=$tmp;
                $flag=1;
            }
        }
    }
    $test=array(1,3,6,8,2,7);
    BubbleSort($test);
    var_dump($test);
```

31、. 以下的代码会产生什么？为什么？

```
$num =10;
function multiply(){
    $num =$num *10;
}
multiply();
echo $num;
```

由于函数 multiply() 没有指定 \$num 为全局变量 (例如 global \$num 或者 \$\_GLOBALS['num']), 所以 \$num 的值是 10。

下面的类是否正确，如果正确请补全方法输入类属性

\$ a;错误请修改并输出属性\$ a

```
<?php
Class A{
    protected$a=1;
    publicfunction test(){
        echo$this->a;
    }
}
答:
$a=new A;
$a->test();
```

请取出数组中的最大值？

\$a=array(1,9,3,5,6,8);  
\$a=array(1,2,3,4,5,9);

```
答:
$b=max($a);
echo $b;
or
$a=array(1,9,3,5,6,8);
rsort($a);
Echo array_shift($a);
```

请问同时满足这样条件的数：被10除于9，被9除于8，被8除于7，在100—1000之间，有几个这样的数？（并写出算法）

A 不存在, B 1个 C 2个 D 3个  
分析过程:

P 除以10余9:P+1就被10整除  
P 除以9余8:P+1就被9整除  
P 除以8余7:P+1就被8整除  
所以 P=8, 9, 10 的公倍数-1  
8=2\*2\*2, 9=3\*3, 10=2\*5  
则 P+1一定是2\*2\*2\*3\*3\*5的倍数(只能约掉一个2)  
又100<P<=1000  
101<P+1<=1001  
所以 P+1=360, 720  
所以 P=359, 719

写出一个简单的静态数组？

```
Class A{
    Static $a=array('11'=>'1111');
}
Print_r(A::$a);
```

从表 login 中选出 name 字段包含 admin 的前10条结果所有信息的 sql 语句

```
Select *from login where name like '%admin%'
limit 0,10;
```

写出发贴数最多的十个人名字的 SQL，利用下表：

members(id, username, posts, pass, email) (2分)

```
SELECT * FROM `members` ORDER BY posts DESC
limit 0,10;
```

使用 php 写一段简单查询，查出所有姓名为“张三”的内容并打印出来 (2分)

姓名	User	Name	Tel	Content	Date
张三	13333663366	大专毕业	2006-10-11		
张三	13612312331	本科毕业	2006-10-15		
张四	021-55665566	中专毕业	2006-10-15		

请根据上面的题目完成代码：

```
$mysql_db=mysql_connect("local","root","pass");
@mysql_select_db("DB",$mysql_db);
$result = mysql_query("SELECT * FROM `user`
WHERE name='张三'");
while($rs = mysql_fetch_array($result)){
    echo
$rs["tel"].$rs["content"].$rs["date"];
}
PDO 写法:
Try{
    $dsn = "mysql:host=localhost;dbname=user";
    $pdo = new PDO($dsn,'root','123456');
    $pdo->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION);
    $pdo->query('set names utf8');
    $sql = "SELECT * FROM user WHERE name='张三'";
    $result = $pdo->query($sql);
    Foreach($result as $v){
        Echo $v['tel']. $v['content']. $v['date'];
    }
}catch(PDOException $e){
    Echo $e->getCode().':'. $e->getMessage();
}
Mysqli 写法:
$mysqli = new
```

```
mysqli('localhost','root','123456','user');
If(mysqli_connect_errno()){
    Die('连接数据库失败, '.mysqli_connect_error());
}
$mysqli->set_charset('utf8');
$result = $mysqli->query("SELECT * FROM user
WHERE name='张三'");
If($result && $result->affected_row()){
    While($row = $result->fetch_object()){
        Echo $row->tel. $row->content. $row->date;
    }
}
```

如何使用下面的类,并解释下面什么意思?(3)

```
class test{
function Get_test($num){
    $num=md5(md5($num)."En");
    return $num;
}
}
```

答:\$testnum = "123";  
\$object = new test();  
\$encrypt = \$object->Get\_test(\$testnum);  
echo \$encrypt;  
类test里面包含Get\_test方法,实例化类调用方法多字符串加密

写出以下程序的输出结果 (1分)

```
<?php
$b=201;
$c=40;
$a=$b>$c?4:5;
echo $a;
?>
```

答:4

检测一个变量是否有设置的函数是否?是否为空的函数是?(2分)

```
isset($str),empty($str);
```

取得查询结果集总数的函数是?(1分)

答:mysql\_num\_rows(\$result);

\$arr = array('james', 'tom', 'symfony'); 请打印出第一个元素的值 (1分)

答:echo \$array[0];

请将 41 题的数组的键值','号分隔并合并成字符串输出(1分)

答:for(\$i=0;\$i

\$a = 'abcdef'; 请取出\$a 的值并打印出第一个字母(1分)

答:echo \$a{0} 或 echo substr(\$a,0,1)

PHP 可以和 sql server/oracle 等数据库连接吗?(1分)

答:当然可以

请写出 PHP5 权限控制修饰符(3分)

答:public(公共),private(私有),protected(继承)

请写出 php5 的构造函数和析构函数(2分)

答:\_\_construct , \_\_destruct

完成以下:

**(一)创建新闻发布系统,表名为 message 有如下字段 (3分)**

id 文章 id  
title 文章标题  
content 文章内容  
category\_id 文章分类 id  
hits 点击量

答:CREATE TABLE 'message' (  
'id' int(10) NOT NULL auto\_increment,  
'title' varchar(200) default NULL,  
'content' text,  
'category\_id' int(10) NOT NULL,  
'hits' int(20),  
PRIMARY KEY('id');  
)ENGINE=InnoDB DEFAULT CHARSET=utf8;

**(二)同样上述新闻发布系统:表 comment 记录用户回复内容,字段如下 (4分)**

comment\_id 回复 id  
id 文章 id, 关联 message 表中的 id  
comment\_content 回复内容  
现通过查询数据库需要得到以下格式的文章标题列表,并按照回复数量排序,回复最高的排在最前面  
文章 id 文章标题 点击量 回复数量  
用一个 SQL 语句完成上述查询,如果文章没有回复则回复数量显示为 0  
答:SELECT message.id id,message.title title,IF(message.`hits` IS NULL,0,message.`hits`) hits,IF(comment.`id` IS NULL,0,count(\*)) number FROM message LEFT JOIN comment ON message.id=comment.id GROUP BY message.`id`;

**(三)上述内容管理系统,表 category 保存分类信息,字段如下 (3分)**

category\_id int(4) not null auto increment;  
category\_name varchar(40) not null;  
用户输入文章时,通过选择下拉菜单选定文章分类  
写出如何实现这个下拉菜单

答:function categoryList(){  
\$result=mysql\_query("select category\_id,category\_name from category")  
or die("Invalid query: " . mysql\_error());  
print(" ... ");  
}

**写一个函数,算出两个文件的相对路径**

如 \$a = '/a/b/c/d/e.php';  
\$b = '/a/b/12/34/c.php';  
计算出 \$b 相对于 \$a 的相对路径应该是  
http://www.cnblogs.com/c/d 将()添上

```
function getRelativePath($a, $b) {
    $returnPath = array(dirname($b));
    $arrA = explode('/', $a);
    $arrB = explode('/', $returnPath[0]);
    for ($n = 1, $len = count($arrB); $n < $len; $n++) {
        if ($arrA[$n] != $arrB[$n]) {
            break;
        }
    }
    if ($len - $n > 0) {
```

```
$returnPath = array_merge($returnPath,
array_fill(1, $len - $n, '..'));
}
$returnPath = array_merge($returnPath,
array_slice($arrA, $n));
return implode('/', $returnPath);
}
echo getRelativePath($a, $b);
```

填空题:

1. 在 PHP 中,当前脚本的名称(不包括路径和查询字符串)记录在预定义变量\_\_\$\_SERVER['PHP\_SELF']\_\_中;而链接到当前页面的 URL 记录在预定义变量\_\_\$\_SERVER['HTTP\_REFERER']\_\_中

2. 执行程序段将输出\_\_0\_\_。

4. 数组函数 arsort 的作用是\_\_对数组进行逆向排序并保持索引关系\_\_;语句 error\_reporting(2047)的作用是\_\_报告所有错误和警告\_\_。

6. 7. 以 Apache 模块的方式安装 PHP, 在文件 http.conf 中首先要用语句\_\_动态装载 PHP 模块\_\_,然后再用语句\_\_使得 Apache 把所有扩展名为 php 的文件都作为 PHP 脚本处理\_\_。

10. 一个函数的参数不能是对变量的引用,除非在 php.ini 中把\_\_allow\_call\_time\_pass\_reference\_\_设为 on。

11. SQL 中 LEFT JOIN 的含义是\_\_自然左外链接\_\_。

7.

**编写一个在线留言本,实现用户的在线留言功能,留言信息存储到数据库?**

假设现在有数据库: test 表: mess 字段: id, name, message

1、html 页面,填写留言:

```
<form method="post" action="post.php">
  留言者: <input type="text" name="name" />
  留言内容: <input type="text" name="message" />
```

/>

```
<input type="submit" value="提交" />
</form>
```

2、post.php 页面,处理留言:

```
$name = $_POST['name'];
$message = $_POST['message'];
```

```
Try{
    $dsn = 'mysql:host=localhost;dbname=test;
    $pdo = new PDO($dsn,'root','112233');
    $pdo->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION);
```

```
$pdo->query('set names utf8');
$sql = "insert into test(name,message)
values('$name','$message)";
```

```
$result = $pdo->exec($sql);
If($result){
```

```
Echo '留言成功';
}else{
```

```
Echo '留言失败';
}
```

```
}catch(PDOException $e){
    Echo $e->getCode().' : '.$e->getMessage();
```

}

**实现不使用第 3 个变量,交换\$a、\$b 的值,\$a、\$b 的初始值自己定。**

```
$a=4; $b=5;
$a=explode(' ', $a.' '.$b); //把用逗号把 4,5 分开。
```



```

$b=$a[0];
$a=$a[1];
echo $a,$b;
在数据库 test 中的一个表 student, 字段是 name,
class, score. 分别代表姓名、所在班级、分数。
1) 学出每个班级中的学生, 按照成绩降序排序;
    select * from student order by score desc;
2) 查出每个班的及格人数和不及格人数, 格式为:
class、及格人数、不及格人数;
Select A.class,A.及格人数,B.不及格人
数 from (select class, count(*) as '及格人数
' from student where score>=60 group by class)
as A, (select class, count(*) as '不及格人数
' from student where score<60 group by class) a
s B where A.class=B.class;
3) 用 PHP 写入连接数据库(“localhost”, “msuser”, “
mypass”), 执行以上 SQL、显示结果、判断错误、关闭
数据库的过程;
$conn=mysql_connect( 'localhost', 'root', '12345
' );
Mysql_select_db( 'test' );
$sql=" ";
$result=Mysql_query($sql);
Mysql_close();

```

ftp://pub.foo.bar.com:2121/incoming/rls.sfv 请给出这个 URL 各部分的意义

- 1、ftp://; 使用的协议是: ftp
- 2、Pub.foo.bar.com 是域名
- 3、2121 是使用的端口
- 4、Incoming/rls.sfv 是当前访问的文件。

下列不属于 Javascript DOM 操作事件的是 C

- A. onclick
- B. onkeydown
- C. onbodyload
- D. onmouseover

有两张数据表, 表 user(id, name) 记录了用户的 ID 和昵称, 表 article(id, title, content, time) 记录了用户发表的文章标题、内容和时间, 写一个 SQL 语句打印出每个用户的昵称及其发表的文章总数。

```

Select user.name,A.id,A.总
数 from user, (select id, count(*) as '总数
' from article group by id) as A where user.id=
A.id;

```

下列 PHP 函数中不能直接在页面输出字符串的函数是 B

- A. echo
- B. sprintf
- C. printf

在 PHP 中, ‘+’ 操作符的功能不包括 A

- A. 字符串连接
- B. 数组数据合并
- C. 变量数据相加

下面哪个选项没有将 john 添加到 users 数组中? C

- (A) \$users[] = 'john';
- (B) array\_push(\$users, 'john');
- (C) \$users ||= 'john';
- (D) array\_unshift(\$users, 'john');

请列出 3 种 PHP 数组循环操作的语法, 并注明每种循环的优缺点

- 1、Foreach() 可以遍历关联数组和索引数组, 操作简单。
- 2、For() 只能遍历索引数组。
- 3、Each list while 以关联和索引两种方式遍历数组, 操作复杂。

请列出 3 个 PHP 中的预定义全局变量

- 1、\$\_SESSION
- 2、\$\_COOKIE
- 3、\$\_SERVER
- 4、\$\_GET
- 5、\$\_POST

76. 从 PHP4.2.0 开始 PHP 默认配置将

register\_globals 配置设定为 off, 请指出这个设定的作用和对 PHP 相关程序的影响。

Register\_globals=off; 的时候, 关闭了全局变量注册, 访问 POST 方式提交值时用 \$\_POST[]; 来访问, 访问 GET 方式提交的值时用 \$\_GET[]; 方式访问。

写一个函数实现字符串翻转

```

1、Strrev()
2、自己写:
Function str($a){
    $len=strlen($a);
    $b=" ";
    For($i=$len-1;$i>=0;$i--){
        $b.=$a[$i];
    }
    Return $b;
}

```

请写出下列 PHP 代码的执行结果

```

$var1 = 5;
$var2 = 10;
function foo(&$my_var){
    global $var1;
    $var1 += 2;
    $var2 = 4;
    $my_var += 3;
    return $var2;
}
$my_var = 5;
echo foo($my_var) . " \n";      4
echo $my_var . " \n";           8
echo $var1 . " \n";             7
echo $var2 . " \n";             10
$bar = 'foo';
$my_var = 10;
echo $bar($my_var) . " \n";     4

```

不断在文件 hello.txt 头部写入一行 “Hello World” 字符串, 要求代码完整

```

$fp=fopen( 'hello.txt', 'r' );
$str='hello!' . " \n";
$str.=fread($fp, filesize( 'hello.txt' ));
fclose($fp);
$fpl=fopen( 'hello.txt', 'w' );
fwrite($fpl, $str);

```

输出用户的 IP 地址, 并且判断用户的 IP 地址是否在 192.168.1.100 — 192.168.1.150 之间

```

Echo $ip=getenv( 'REMOTE_ADDR' );
$ip=Str_replace( '.', '', $ip);
If($ip<1921681100 && $ip>1921681100){
Echo 'ip 在 192.168.1.100—192.168.1.150 之间';
}

```

```

}Else{
Echo ' ip 不在 192.168.1.100--192.168.1.150 之间
';
}

```

**请将 2 维数组按照 name 的长度进行重新排序，按照顺序将 id 赋值。**

```

<?php
$Tarray = array(
array( 'id' => 0, 'name' => '123' ),
array( 'id' => 0, 'name' => '1234' ),
array( 'id' => 0, 'name' => '1235' ),
array( 'id' => 0, 'name' => '12356' ),
array( 'id' => 0, 'name' => '123abc' )
)
foreach($tarray as $key=>$val){
$c[]=$val['name'];
}
function aa($a,$b){
if(strlen($a)==strlen($b)) return 0;
return strlen($a)>strlen($b)?-1:1;
}
usort($c,'aa');
$len=count($c);
for($i=0;$i<$len;$i++){
$t[$i]['id']=$i+1;
$t[$i]['name']=$c[$i];
}
echo ' <pre>';
print_r($t);
echo ' </pre>';

```

**输出为 Mozilla/4.0(compatible;MSIE5.01;Window NT 5.0)时，可能的输出语句是：B**

- A. \$\_SERVER['HTTP\_USER\_AGENT\_TYPE'];
- B. \$\_SERVER['HTTP\_USER\_AGENT'];
- C. \$\_SERVER['USER\_AGENT'];
- D. \$\_SERVER['AGENT'];

**下面功能 PHP 无法实现的是：D**

- A. 服务器端脚本运行
- B. 命令行脚本运行
- C. 客户端图形界面（GUI）程序
- D. 浏览器端执行 DOM 操作

**下面说法不正确的是：C**

- A. PHP 有四种标量类型：布尔型(boolean)，整型(integer)，浮点型(float)，字符串(string)
- B. 浮点型(float)与双精度型(double)是同一种类型
- C. 符合类型包括：数组(array)，对象(object)，资源(resource)
- D. 伪类型：混合型(mixed)，数字型(number)，回调(callback)

**执行下面的代码后，A**

```

<?php echo 和 print 不是系统函数（看手册）
echo function_exists('print');
imagecreatetruecolor
??
得到的输出是：
A. 空
B. true
C. false
D. FALSE

```

**下面不是 PHP 语法的组成部分的函数是：AD**

- A. array
- B. eval
- C. each
- D. list

**执行下面的代码的结果是什么？A**

```

<?php
$bool = TRUE;
echo gettype($bool);
echo is_string($bool);
??
A. boolean
B. boolean0
C. booleanFALSE
D. booleanfalse

```

**写出下面代码执行的结果：12 10 18**

```

<?php
$a = 12;
$b = 012;
$c = 0x12;
echo$a,"\\n", $b,"\\n", $c;
??

```

**下面代码的执行结果是什么？C**

```

<?php
echo 1+2+"3+4+5";
??
A. 0
B. 3
C. 6
D. "3+4+5";

```

**下面代码加入下面那个函数后返回 TRUE:B**

```

return ? == 'A';
A. ord(65)
B. chr(65)
C. 65+' '
D. '' +65

```

**下面代码的输出正确的是：C**

```

<?php
$a =array(1=>5, 5=>8, 22, 2=>'8', 81);
echo $a[7];
echo $a[6];
echo $a[3];
??
A. 空 2281 B. 空 8122C. 8122 空 D. 空空空

```

**下面代码的输出结果：D**

```

<?php
$a[bar]='hello';
echo $a[bar];
echo $a['bar'];
??
A. hello B. 空空 C. 报错 D. hellohello

```

**写出下面代码的结果：C**

**左右相当于乘 2，右移相当于除 2**

```

<?php
二进制位移后
echo 1>>0; 0001 0001
echo 2>>1; 0010 0001
echo 3<<2; 0011 1100
??
A. 012 B. 106 C. 1112 D. 123

```

**下面代码的执行结果: D**

```
<?php 中间的 i 少个$
for($i=0;$i<10;$i++){
    print $i;
}
?>
```

- A. 0123456789 B. 012345678910 C. 无输出  
D. 死循环

**下面对于 echo, print 的描述正确的是: C**

- A. echo, print 都可以打印多个参数  
B. print 可以打印多个参数, echo 不可以  
C. echo 可以打印多个参数, print 不可以  
D. echo, print 都不可以打印多个参数

**对于正面的代码**

```
<?php
$fruits =
array('strawberry'=>'red','banana'=>'yellow');
?>
```

能够正确得到结果 'yellow' 的代码是: ACD

- A. echo "A banana is{\$fruits['banana']}";  
B. echo "A banana is\$fruits['banana']";  
C. echo "A banana is{\$fruits[banana]}";  
D. echo "A banana is\$fruits[banana]";

**下面代码执行完成后的结果是什么? 1 2**

```
<?php
function change(){
    static $i = 0;
    $i++;
    return $i;
}
print change(); //1
print change(); //2
?>
```

**下面的 include 语句不正确的是? A**

- A. include "mms://www.abc.com/a.php";  
B. include "http://www.abc.com/a.php";  
C. include "https://www.abc.com/a.php";  
D. include "ftp://www.abc.com/home/a.php";

```
<?php
$foo = 'test';
$bar = <<<'EOT'
$foobar
EOT;
echo$bar;
?>
```

**上面的语句输出结果 (a): A**

- a. \$foo bar;  
b. 'EOT' \$foo bar EOT;  
c. test bar;  
d. 'EOT' test bar EOT;

\$a=3;\$b=4;

```
if($a||$b=5){ ||或和|的区别
    echo 'todo';
}
```

\$b 的值是 (a): A

- a. 4;  
b. 5;  
c. 3;  
d. False

**如何提高 php 应用的性能**

- 如果能将类的方法定义成 static, 就尽量定义成 static, 它的速度会提升将近4倍。
- \$row['id'] 的速度是\$row[id]的7倍。
- echo 比 print 快, 并且使用 echo 的多重参数 (译注: 指用逗号而不是句点) 代替字符串连接, 比如 echo\$str1,\$str2。
- 在执行 for 循环之前确定最大循环数, 不要每循环一次都计算最大值, 最好运用 foreach 代替。
- 注销那些不用的变量尤其是大数组, 以便释放内存。
- 尽量避免使用 \_\_get, \_\_set, \_\_autoload。
- require\_once() 代价昂贵。
- include 文件时尽量使用绝对路径, 因为它避免了 PHP 去 include\_path 里查找文件的速度, 解析操作系统路径所需的时间会更少。
- 如果你想知道脚本开始执行 (译注: 即服务器端收到客户端请求) 的时刻, 使用 \$\_SERVER['REQUEST\_TIME'] 要好于 time()。
- 函数代替正则表达式完成相同功能。
- str\_replace 函数比 preg\_replace 函数快, 但 strpos 函数的效率是 str\_replace 函数的四倍。
- 如果一个字符串替换函数, 可接受数组或字符作为参数, 并且参数长度不太长, 那么可以考虑额外写一段替换代码, 使得每次传递参数是一个字符, 而不是只写一行代码接受数组作为查询和替换的参数。
- 使用选择分支语句 (译注: 即 switch case) 好于使用多个 if, else if 语句。
- 用 @ 屏蔽错误消息的做法非常低效, 极其低效。
- 打开 apache 的 mod\_deflate 模块, 可以提高网页的浏览速度。
- 数据库连接当使用完毕时应关掉, 不要用长连接。
- 错误消息代价昂贵。
- 在方法中递增局部变量, 速度是最快的。几乎与在函数中调用局部变量的速度相当。
- 递增一个全局变量要比递增一个局部变量慢2倍。
- 递增一个对象属性 (如: \$this->prop++) 要比递增一个局部变量慢3倍。
- 递增一个未预定义的局部变量要比递增一个预定义的局部变量慢9至10倍。
- 仅定义一个局部变量而没在函数中调用它, 同样会减慢速度 (其程度相当于递增一个局部变量)。PHP 大概会检查是否存在全局变量。
- 方法调用看来与类中定义的方法的数量无关, 因为我 (在测试方法之前和之后都) 添加了10个方法, 但性能上没有变化。
- 派生类中的方法运行起来要快于在基类中定义同样的方法。
- 调用带有一个参数的空函数, 其花费的时间相当于执行7至8次的局部变量递增操作。类似的方法调用所花费的时间接近于15次的局部变量递增操作。
- Apache 解析一个 PHP 脚本的时间要比解析一个静态 HTML 页面慢2至10倍。尽量多用静态 HTML 页面, 少用脚本。
- 除非脚本可以缓存, 否则每次调用时都会重新编译一次。引入一套 PHP 缓存机制通常可以提升25%至100%的性能, 以免除编译开销。
- 尽量做缓存, 可使用 memcached。memcached 是一款高性能的内存对象缓存系统, 可用来加速动态 Web 应用程序, 减轻数据库负载。对运算码 (OP code) 的缓存很有用, 使得脚本不必为每个请求做重新编译。
- 当操作字符串并需要检验其长度是否满足某种要求时, 你想当然地会使用 strlen() 函数。此函数执行起来相当快, 因为它不做任何计算, 只返回在 zval 结

构(C的内置数据结构,用于存储 PHP 变量)中存储的已知字符串长度。但是,由于 strlen()是函数,多多少少会有些慢,因为函数调用会经过诸多步骤,如字母小写化(译注:指函数名小写化,PHP不区分函数名大小写)、哈希查找,会跟随被调用的函数一起执行。在某些情况下,你可以使用 isset() 技巧加速执行你的代码。

(举例如下)

```
if (strlen($foo) < 5) { echo "Foo is too short"
$ $ }
```

(与下面的技巧做比较)

```
if (!isset($foo{5})) { echo "Foo is too short"
$ $ }
```

调用 isset() 恰巧比 strlen() 快,因为与后者不同的是,isset() 作为一种语言结构,意味着它的执行不需要函数查找和字母小写化。也就是说,实际上在检验字符串长度的顶层代码中你没有花太多开销。

34、当执行变量 \$i 的递增或递减时, \$i++ 会比 ++\$i 慢一些。这种差异是 PHP 特有的,并不适用于其他语言,所以请不要修改你的 C 或 Java 代码并指望它们能立即变快,没用的。++\$i 更快是因为它只需要3条指令(opcodes), \$i++ 则需要4条指令。后置递增实际上会产生一个临时变量,这个临时变量随后被递增。而前置递增直接在原值上递增。这是最优化处理的一种,正如 Zend 的 PHP 优化器所作的那样。牢记这个优化处理不失为一个好主意,因为并不是所有的指令优化器都会做同样的优化处理,并且存在大量没有装配指令优化器的互联网服务提供商(ISPs)和服务器。

35、并不是事必面向对象(OOP),面向对象往往开销很大,每个方法和对象调用都会消耗很多内存。

36、并非要用类实现所有的数据结构,数组也很有用。

37、不要把方法细分得过多,仔细想想你真正打算重用的是哪些代码?

38、当你需要时,你总能把代码分解成方法。

39、尽量采用大量的 PHP 内置函数。

40、如果在代码中存在大量耗时的函数,你可以考虑用 C 扩展的方式实现它们。

41、评估检验(profile)你的代码。检验器会告诉你,代码的哪些部分消耗了多少时间。Xdebug 调试器包含了检验程序,评估检验总体上可以显示出代码的瓶颈。

42、mod\_zip 可作为 Apache 模块,用来即时压缩你的数据,并可以让数据传输量降低80%。

43、在可以用 file\_get\_contents 替代 file、fopen、feof、fgets 等系列方法的情况下,尽量用 file\_get\_contents,因为他的效率高得多!但是要注意 file\_get\_contents 在打开一个 URL 文件时候的 PHP 版本问题;

44、尽量少的进行文件操作,虽然 PHP 的文件操作效率也不低的;

45、优化 Select SQL 语句,在可能的情况下尽量少的进行 Insert、Update 操作;

46、尽可能的使用 PHP 内部函数(但是我却为了找个 PHP 里面不存在的函数,浪费了本可以写出一个自定义函数的时间,经验问题啊!);

47、循环内部不要声明变量,尤其是大变量:对象(这好像不只是 PHP 里面要注意的问题吧?);

48、多维数组尽量不要循环嵌套赋值;

49、在可以用 PHP 内部字符串操作函数的情况下,不要用正则表达式;

50、foreach 效率更高,尽量用 foreach 代替 while 和 for 循环;

51、用单引号替代双引号引用字符串;

52、“用 i+=1 代替 i=i+1。符合 c/c++ 的习惯,效率还高”;

53、对 global 变量,应该用完就 unset() 掉。

### 13、谈谈 asp, php, jsp 的优缺点(1分)

ASP 全名 Active Server Pages, 是一个 WEB 服务器端的开发环境, 利用它可以产生和运行动态的、交互的、高性能的 WEB 服务应用程序。ASP 采用脚本语言 VB Script (Java script) 作为自己的开发语言。

PHP 是一种跨平台的服务器端的嵌入式脚本语言。它大量地借用 C, Java 和 Perl 语言的语法, 并耦合 PHP 自己的特性, 使 WEB 开发者能够快速地写出动态生成页面。它支持目前绝大多数数据库。还有一点, PHP 是完全免费的, 不用花钱, 你可以从 PHP 官方站点 (<http://www.php.net>) 自由下载。而且你可以不受限制地获得源码, 甚至可以从其中加进你自己需要的特色。

JSP 是 Sun 公司推出的新一代站点开发语言, 他完全解决了目前 ASP, PHP 的一个通病——脚本级执行 (据说 PHP4 也已经在 Zend 的支持下, 实现编译运行)。Sun 公司借助自己在 Java 上的不凡造诣, 将 Java 从 Java 应用程序 和 Java Applet 之外, 又有新的硕果, 就是 Jsp——Java Server Page。Jsp 可以在 Servlet 和 JavaBean 的支持下, 完成功能强大的站点程序。

三者都提供在 HTML 代码中混合某种程序代码、由语言引擎解释执行程序代码的能力。

但 JSP 代码被编译成 Servlet 并由 Java 虚拟机解释执行, 这种编译操作仅在对 JSP 页面的第一次请求时发生。在 ASP、PHP、JSP 环境下, HTML 代码主要负责描述信息的显示样式, 而程序代码则用来描述处理逻辑。普通的 HTML 页面只依赖于 Web 服务器, 而 ASP、PHP、JSP 页面需要附加的语言引擎分析和执行程序代码。程序代码的执行结果被重新嵌入到 HTML 代码中, 然后一起发送给浏览器。ASP、PHP、JSP 三者都是面向 Web 服务器的技术, 客户端浏览器不需要任何附加的软件支持。