【1】页面之间无法传递变量 get,post,session在最新的php版本中自动全局变量是关闭的，所以要从上一页面取得提交过来得变量要使用$\_GET['foo'],$\_POST['foo'],$\_SESSION['foo']来得到  
当然也可以修改自动全局变量为开(php.ini改为register\_globals = On)；考虑到兼容性，还是强迫自己熟悉新的写法比较好。  
================================  
PHP中的超全局变量  
从PHP 4.2.0 开始，register\_globals 的默认值为 off，这样一来，以前的很多可以直接使用的变量，如 $PHP\_SELF 或者你设定的SESSION变量都不能用 “$变量名”的形式访问了，这可能会给你带来很多不变，但却有助于安全性的提高。访问这些变量，你需要使用PHP 超全局变量，如下：  
  
$\_SERVER  
变量由 Web 服务器设定或者直接与当前脚本的执行环境相关联。类似于旧数组 $HTTP\_SERVER\_VARS 数组。以前的$PHP\_SELF对应$\_SERVER['PHP\_SELF']，你可以使用phpinfo来查看你的$\_SERVER变量。  
  
$\_GET  
经由 HTTP GET 方法提交至脚本的变量。类似于旧数组 $HTTP\_GET\_VARS 数组。  
  
$\_POST  
经由 HTTP POST 方法提交至脚本的变量。类似于旧数组 $HTTP\_POST\_VARS 数组。  
  
$\_COOKIE  
经由 HTTP Cookies 方法提交至脚本的变量。类似于旧数组 $HTTP\_COOKIE\_VARS 数组。  
  
$\_SESSION  
当前注册给脚本会话的变量。类似于旧数组 $HTTP\_SESSION\_VARS 数组。  
  
$\_FILES  
经由 HTTP POST 文件上传而提交至脚本的变量。类似于旧数组 $HTTP\_POST\_FILES 数组。  
  
$\_ENV  
执行环境提交至脚本的变量。类似于旧数组 $HTTP\_ENV\_VARS 数组。  
  
====================================================================  
对于$\_FILES变量：（文件域字段为“myfile”）  
  
$\_FILES['myfile']['name']    
客户端机器文件的原名称（包括路径）。  
  
$\_FILES['myfile']['type']    
文件的 MIME 类型，需要浏览器提供该信息的支持，例如“image/gif”。  
  
$\_FILES['myfile']['size']    
已上传文件的大小，单位为字节。    
  
$\_FILES['myfile']['tmp\_name']    
文件被上传后在服务端储存的临时文件名（包括路径）。  
  
$\_FILES['myfile']['error']    
和该文件上传相关的错误代码。['error'] 是在 PHP 4.2.0 版本中增加的。    
  
当 php.ini 中的 register\_globals 被设置为 on 时，$myfile\_name 等价于 $\_FILES['myfile']['name']，$myfile\_type 等价于 $\_FILES['myfile']['type']等。    
  
  
  
  
=========================================  
【2】Win32下apache2 用get方法传递中文参数会出错  
test.php?a=你好&b=你也好  
传递参数是会导致一个内部错误  
解决办法:"test.php?a=".urlencode(你好)."&b=".urlencode(你也好)  
  
.............  
  
  
【3】win32下的session不能正常工作  
php.ini默认的session.save\_path = /tmp  
这显然是linux下的配置，win32下php无法读写session文件导致session无法使用  
把它改成一个绝对路径就可以了，例如session.save\_path = c:windowstemp

【4】显示错误信息  
当php.ini的display\_errors = On并且error\_reporting = E\_ALL时，将显示所有的错误和提示，调试的时候最好打开以便纠错，如果你用以前php写法错误信息多半是关于未定义变量的。变量在赋值以前调用会有提示，解决办法是探测或者屏蔽  
例如显示$foo，可以if(isset($foo)) echo $foo 或者echo @$foo

【5】Win32下mail()不能发送电子邮件  
在linux下配置好的sendmail可以发送，在win32下需要调用smtp服务器来发送电子邮件  
修改php.ini的SMTP = ip //ip是不带验证功能的smtp服务器(网上很难找到)  
php发送邮件的最好解决方法是用socket直接发送到对方email服务器而不用转发服务器

【6】初装的mysql如果没有设置密码，应该使用  
update mysql.user set password="yourpassword" where user="root"  
修改密码

【7】header already sent  
这个错误通常会在你使用HEADER的时候出现，他可能是几种原因：1，你在使用HEADER前PRING或者ECHO了2.你当前文件前面有空行3.你可能INCLUDE了一个文件,该文件尾部有空行或者输出也会出现这种错误。！

【8】更改php.ini后没有变化  
重新启动web server，比如IIS，Apache等等，然后才会应用最新的设置

【9】php在2003上面安装（ISAPI的安装方法恳请高手指教）  
PHP4的php4isapi.dll好像和2003有些冲突，只能用CGI模式安装  
步骤一，先[www.php.net](http://www.php.net/)下在一个安装程序，我是装的是：php-4.2.3-installer.exe，你也可以去找最新的版本，在安装php-4.2.3-installer.exe之前保证你的IIS6.0启动了，并能够访问。 安装好以后，在默认网站-->应用程序配置  
步骤二：点击 web服务扩展 -->新建web服务扩展.  
步骤三： 扩展名-->php,然后添加  
步骤四：找到php.exe的路径添加上去。  
步骤五： 确定就可以了！  
步骤六： 选择php的服务扩展，然后点击允许。

【10】  
有时候sql语句不起作用，对数据库操作失败  
最简便的调试方法，echo那句sql，看看变量的值能得到不

【11】include和require的区别  
两者没有太大的区别，如果要包含的文件不存在，include提示notice，然后继续执行下面的语句，require提示致命错误并且退出  
据我测试，win32平台下它们都是先包含后执行，所以被包含文件里最好不要再有include或require语句，这样会造成目录混乱。或许\*nux下情况不同，暂时还没测试  
如果一个文件不想被包含多次可以使用include\_once或require\_once## 读取，写入文档数据  
function r($file\_name) {  
$filenum=@fopen($file\_name,"r");  
@flock($filenum,LOCK\_SH);  
$file\_data=@fread($filenum,filesize($file\_name));  
@fclose($filenum);  
return $file\_data;  
}  
function w($file\_name,$data,$method="w"){  
$filenum=@fopen($file\_name,$method);  
flock($filenum,LOCK\_EX);  
$file\_data=fwrite($filenum,$data);  
fclose($filenum);  
return $file\_data;  
}  
  
【12】isset()和empty()的区别  
两者都是测试变量用的  
但是isset()是测试变量是否被赋值，而empty()是测试一个已经被赋值的变量是否为空  
如果一个变量没被赋值就引用在php里是被允许的,但会有notice提示  
如果一个变量被赋空值，$foo=""或者$foo=0或者 $foo=false,那么empty($foo)返回真，isset($foo)也返回真，就是说赋空值不会注销一个变量。  
要注销一个变量，可以用 unset($foo)或者$foo=NULL

【13】mysql查询语句包含有关键字  
php查询mysql的时候，有时候mysql表名或者列名会有关键字  
这时候查询会有错误。例如表名是order,查询时候会出错  
简单的办法是sql语句里表名或者列名加上`[tab键上面]来加以区别  
例如select \* from `order`

【14】通过HTTP协议一次上传多个文件的方法  
有两个思路，是同一个方法的两种实现。具体程序还需自己去设计  
1,在form中设置多个文件输入框，用数组命名他们的名字，如下：  
<form action="" method=post>  
<input type=file name=usefile[]>  
<input type=file name=usefile[]>  
<input type=file name=usefile[]>  
</form>  
这样，在服务器端做以下测试  
echo "<pre>";  
print\_r($\_FILES);  
echo "</pre>";  
  
1,在form中设置多个文件输入框，但名字不同，如下：  
<form action="" method=post>  
<input type=file name=usefile\_a>  
<input type=file name=usefile\_b>  
<input type=file name=usefile\_c>  
</form>  
在服务器端做同样测试：  
echo "<pre>";  
print\_r($\_FILES);  
echo "</pre>";