**PHP学习总结**

1.<?php

...

?>

<%

...

%>

2.引用文件 require "conf.php" 或 require ("conf.php") 则会产生一个致命错误

require\_once('conf.php')只引用一次

3.include 'other.php' 如果引用文件不存在，include会产生警告

include $somefile

include ('other.php')

4.注释 // 单行

/\*

\*\*\*多行注释

\*/

5.命名规则

所有字母均使用小写，使用"\_"作为每个词的分界

6.gettype(2.2) 输出该数值的类型

7.三类八种

A基本类型

Boolean integer float string

B复合类型

array object

C特殊数据类型

resource NULL无值

8.自定义常量

define(string name,mixed value[,bool case\_insensitive])默认为false即大小写敏感

define("hostname","localhost")

9.系统常量

get\_defined\_constants()一次获取PHP系统中所有的预定义常量

PHP\_VERSION string 当前PHP版本

PHP\_OS string 当前所使用的操作系统类型

PHP\_SAPI string Web服务器与PHP之间的接口

PHP\_INT\_MAX integer 最大的整型数

PHP\_INT\_SIZE integer 整型数的大小

DEFAULT\_INCLUDE\_PATH string php默认的包含路径

PEAR\_INSTALL\_DIR string pear的安装路径

PEAR\_EXTENSION\_DIR string pear扩展路径

PHP\_BINDIR string 执行的路径

PHP\_LIBDIR string php的扩展模块所在路径

E\_ERROR integer 指向最近的错误处

E\_WARNING integer 指向最近的警告处

E\_NOTICE integer 指向最近的注意处

M\_E integer 自然对数e的值

M\_PI integer 数学上圆周率的值

魔术常量

\_\_LINE\_\_ string 当前代码所在的行数

\_\_FILE\_\_ string 当前文件路径名

\_\_FUNCTION\_\_ string 当前被调用的函数名

\_\_CLASS\_\_ string 类的名称名

\_\_METHOD\_\_ string 类的方法名

10.php 九大类预定义常量

$\_SERVER $\_ENV $\_COOKIE $\_GET $\_POST $\_FILE $\_REQUEST $\_SESSION $\_GLOBALS

$\_SERVER["PHP\_SELF"] 当前正在执行的脚本的文件名

$\_SERVER["argv"] 传递给该脚本的参数

$\_SERVER["argc"] 包含传递给程序的命令行参数的个数

$\_SERVER["GATEWAY\_INTERFACE"] 服务器使用的CGI的规范版本

$\_SERVER["SERVER\_ADDR"] 当前运行脚本所在服务器的Ip地址

$\_SERVER["SERVER\_NAME"] 当前运行脚本所在服务器的主机名称

$\_SERVER["SERVER\_SOFTSARE"] 服务器标识字符串，在响应请求时的头信息中给出

$\_SERVER["SERVER\_PROTOCOL"] 请求页面时通信协议的名称和版本

$\_SERVER["SERVER\_METHOD"] 访问页面是的请求方法

$\_SERVER["REQUEST\_TIME"] 请求开始的时间戳

$\_SERVER["QUERY\_STRING"] 查询的字符串，URL第一个？之后的内容

$\_SERVER["DOCUMENT\_ROOT"] 当前运行脚本所在的文档根目录

$\_SERVER["HTTP\_ACCEPT"] 当前请求的accept头内容

$\_SERVER["HTTP\_ACCEPT\_CHARSET"] 当前请求的accept-change头信息的内容

$\_SERVER["HTTP\_ACCEPT\_LANGUAGE"] 当前请求的accept-language头信息的内容

$\_SERVER["HTTP\_CONNECTION"] 当前请求的connection头信息的内容

$\_SERVER["HTTP\_HOST"] 当前请求的host头信息的内容

$\_SERVER["HTTP\_REFERER"] 链接到当前页面的前一页面的URL地址

$\_SERVER["HTTP\_USER\_AGENT"] 当前请求的user-agent头信息的内容

$\_SERVER["HTTPS"] 如果脚本是通过https协议被访问的，则被设为一个非空的值

$\_SERVER["REMOTE\_ADDR"] 正在浏览当前页面用户的IP地址

$\_SERVER["REMOTE\_HOST"] 正在浏览当前页面用户的主机名

$\_SERVER["REMOTE\_PORT"] 用户连接到服务器时所使用的端口

$\_SERVER["SCRIPT\_FILENAME"] 当前执行脚本的绝对路径名

$\_SERVER["SERVER\_ADMIN"] 该值指明了Apache服务器配置文件中的server\_admin参数

$\_SERVER["SERVER\_PORT"] 服务器所使用的端口

$\_SERVER["SERVER\_SIGNATURE"] 包含服务器版本和虚拟主机名的字符串

$\_SERVER["PATH\_TRANSLATED"] 当前脚本所在文件系统的基本路径

$\_SERVER["SCRIPT\_NAME"] 包含当前脚本的路径

$\_SERVER["REQUEST\_URI"] 访问此页面所需的Uri

$\_SERVER["PHP\_AUTH\_DIGEST"] 当作为Apache模块运行时，进行HTTP Digest认证的过程中，此变量被设置为客户端发送的authorization HTTP头内容

$\_SERVER["PHP\_AUTH\_USER"] 当PHP运行在Apache或IIs模块方式下，并且正在使用HTTP认真功能时，这个变量便是用户输入的用户名

$\_SERVER["PHP\_AUTH\_PW"]当PHP运行在Apache或IIs模块方式下，并且正在使用HTTP认真功能时，这个变量便是用户输入的密码

$\_SERVER["AUTH\_TYPE"] 当php运行在Apache模块方式下，并且正在使用HTTP认证功能时，这个变量便是认证的类型

$\_ENV["ALLUSERSPROFILE"] 当前系统所有用户的概况文件夹

$\_ENV["CommomProgramFiles"] 当前系统的公共程序安装目录

$\_ENV["COMPUTERNAME"] 当前系统的计算机名称

$\_ENV["ComSpec"] 当前系统的命令

$\_ENV["FP\_NO\_HOST\_CHECK"] 当前系统是否有服务器验证

$\_ENV["NUMBER\_OF\_PROCESSORS"] 当前系统的Cpu处理器的数量

$\_ENV["OS"] 当前系统的操作系统

$\_ENV["Path"] 当前系统的路径

$\_ENV["PATHEXT"] 当前系统中可以直接执行的文件扩展名

$\_ENV["PROCESSOR\_ARCHITECTURE"] 当前系统的cpu处理器的架构

$\_ENV["PROCESSOR\_IDENTIFIER"] 当前系统的cpu处理器的标识

$\_ENV["PROCESSOR\_LEVEL"] 当前系统的cpu处理器的层

$\_ENV["PROCESSOR\_REVISION"] 当前系统的cpu处理器的修正版本

$\_ENV["ProgramFiles"] 当前系统的程序安装目录

$\_ENV["SystemDrive"] 当前系统的系统盘符

$\_ENV["SystemRoot"] 当前系统的系统路径

$\_ENV["TEMP"] 当前系统的临时文件夹

$\_ENV["TMP"] 当前系统的临时文件夹

$\_ENV["USERPROFILE"] 当前系统的用户概况文件夹

$\_ENV["windir"] 当前系统的Windows路径

$\_ENV["AP\_PARENT\_PID"] 当前系统的Web服务进程标识

$\_COOKIE 表示所有注册的cookie数组

$\_GET 所有get请求的数组

$\_POST 所有post请求的数组

$\_FILES["upfile"]["name"] 表示上传文件的名称

$\_FILES["upfile"]["type"] 表示文件的类型

$\_FILES["upfile"]["tmp\_name"] 表示上传文件的临时文件名(包括路径)

$\_FILES["upfile"]["error"] 表示上传文件是否有误

$\_FILES["upfile"]["size"] 表示上传文件的大小

$\_REQUEST 该数组包含所有的$\_GET,$\_POST,$\_COOKIE的内容

$\_SESSION 所有注册的session数组

$\_GLOBAL 该数组包含所有已经注册的全局变量

11.$a=array(0=>0,1=>1,2=>"a"...)

12.print\_r($a) 输出数组内容

13.function my\_fun()

{

echo "hello world";

...

}

my\_fun();

14.

mysql\_connect("localhost","root","");

$arr=array("a","c","b","123",87,"ching");

$num=count($arr);

15.class my\_class

{

var $my\_name;

var $my\_address;

function my\_fun()

{

echo "欢迎光临";

}

}

实例化一个类 $obj=new my\_class;

$obj\_name->my\_name="八位";

$obj\_address->my\_address="云类峡";

$obj\_name->my\_fun();

16.class child\_class extends parent\_class

{

...

}

子类中引用父类三种方法：

$this->ParentFunction();

PatentClassName::PatentFunction();

patent::PatentFunction();

17.get\_class\_methods("类名") 返回类的所有的方法，以数组形式返回

get\_class\_vars("类名") 以数组形式返回类的所有变量

get\_class("变量名") 返回指定对象的类名

class\_exists("类名") 判断指定类是否已经存在

property\_exists("类名","属性") 根据类名或者对象是否存在某个属性而返回相应的布尔值

method\_exists("类名","方法名") 根据类名或者对象是否存在某个方法而返回相应的布尔值

abs() 求绝对值

ceil(3.5) 小数取整，结果为4，返回不小于指定小数的最大值

floor(6.2) 取小于6.2的最大整数值，结果为6

round(9.6) 四舍五入取整

deg2rad() 角度转化为弧度

rad2deg() 弧度转化为角度

pow($a,$b) $a的$b次方

sqrt() 求平方根

sin() 求正弦值

cos() 求余弦值

tan() 求正切值

asin() 求反正弦值

acos() 求反余弦值

atan() 求反正切值

max() 求一组数的最大值

min() 求一组数的最小值

rand(0,9)生成0-9之间的数

decbin() 将十进制转化为二进制

bindec() 将二进制转化为十进制

decoct() 将十进制转化为八进制

octdec() 将八进制转化为十进制

dechex() 将十进制转化为十六进制

hexdec() 将十六进制转化为十进制

任意进制之间的转化

base\_convert(string 需要进行转化的数值,int 基础进制,int 目标进制)

exp() 计算e的指数

fmod() 求余数函数

hypot() 计算直角三角形的斜边长

log10() 以10为底的对数

log() 自然对数

pi() 获取圆周率，无需任何参数

print("hello world") 打印函数

$str="hello";

$number=123;

printf("%s world, day number%u",$str,$number);

strlen() 求字符串长度

substr(string $string,int $start[,int $length]); 字符串截取

strtr("","") 一字符串某些字符替换

substr\_replace(string $string,string $replacement,int $start[,int $length]) 将字符串中一部分替换为另一字符串

str\_replace(string $search,string $replace,string $subject[,int $count]) 搜索所有$search字符在$subject中，替换为$replace

explode(string $sep,string $string[,int $limit]) 字符串分割

strcasecmp() 字符串比较，不区分大小写

strnatcmp()字符串比较，区分大小写

strtoupper(string $str) 字符串中所有的内容转化为大写

strtolower(string $str) 字符串中所有的内容转化为小写

ucfirst(string $str) 字符串首字母大写

ucwords(string $str) 字符串中每个单词的首字母大写

ord() 将字符转化为ASCII码

chr() ASCII码转化为字符

htmlentities() 转化html特殊字符

htmlspecialchars() 转化html特殊字符

strops("ching","g") 字符串查找

count\_chars() 统计字符串中所使用的字符信息

md5(string $str[,bool $raw\_output]) 加密,若为true则返回16位长的二进制格式，默认为false，返回32位的md5字符

strrev() 翻转字符串

str\_repeat(string $str[,int $multiplier]) 字符串重复的次数

str\_repeat("ab",5);重复5次

array\_count\_values(array $input) 检查数组中值出现的次数

arsort(array &$array[,int $sort\_flag]) 按照数组的大小逆向排序,

可选参数为：sort\_regular，sort\_numberic，sort\_string，sort\_locale\_string

asort()顺向排，其他和上面函数一样

krsort() 按照数组键名的大小逆向排序，参数和上面两函数一样

ksort() 顺向排，其他同上

rsort() 按照数组的大小逆向排序,参数同上

sort() 顺向排，其他同上

uasort(array &$array,callback $cmp\_function) 自定义排序，第二参数为用户指定排序规则的函数

移动数组指针

current() 返回数组当前指针所指向的指，初始的数组指向第一个值

each() 返回数组当前指针所指向的键值对，并将指针后移一位

next(() 将数组指针后移一位

prev() 将数组指着前移一位

reset() 重置数组，即将数组指针移到第一位

array\_unique(array $array) 移除数组中的重复值

合并数组用“+”

array\_merge(array $array1[,array $array2,...]) 合并数组

array\_merge\_recursive(array $array1[,array $array2,...]) 函数递归合并数组

in\_array() 检查数组中是否存在某个值 in\_array(6,$arr) 判断数组中是否包含6这个值

array\_intersect() 计算两个数组的交集

array\_key\_exists() 检查数组中是否有某个键名

array\_keys() 返回数组键名

array\_sum() 计算数组所有值得和

date(string format[,int timestamp]) 格式化日期函数

getdate() 获取日期时间

time() 可以获取当前的Unix时间戳

strtotime() 将使用英文描述的日期时间解析为Unix时间戳

打印前一天的日期 date("Y-m-d",strtotime("-1" day));

magic\_quotes\_gpc 为on时会进行转义，自动运行addslashes()

这样会导致双层转义，遇到这种情况可以使用get\_magic\_quotes\_gpc()进行检测。

18.数据库操作

$con=mysql\_connect("localhost","root","123456"); 连接数据库

mysql\_query(string query[,resource $link\_identifier]) 执行SQL语句

show databases 显示数据库名

mysql\_num\_rows(resource $result) 获取结果记录集

mysql\_fetch\_array(resource $result[,int $result\_type]) 遍历结果集

create database my\_go;

show table

mysql\_select\_db("test") 选择test数据库

drop database db\_name;

drop table table\_name;

mysql\_insert\_id() 获取最后插入的Id

mysql\_fetch\_row() 获取一行记录

% 表示任意0个或者多个字符

\_ 表示任意单个字符

[] 表示在括号内所列字符中的一个

[^] 表示不在括号所列之内的单个字符

abs(x) 绝对值

greatest(x1,x2,x3...) 返回集合中的最大值

least(x1,x2,x3...) 返回集合中的最小值

round(x,y) 返回参数x的四舍五入的有y位小数的指

avg(col) 返回指定列的平均值

count(col) 返回指定列中非null值的个数

min(col) 返回指定列的最小值

max(col) 返回指定列的最大值

sum(col) 返回指定列的所有值之和

ASCII(char) 返回字符串的ASCII值

concat(sep,s1,s2,s3...sn) 将s1,s2,s3...sn连接上字符串，并用sep隔开

insert(str,x,y,instr) 将字符串str从x位置开始，y个字符长的字串替换为字符串instr

lcase(str)或lower(str) 将所有字符串改变为小写

left(str,x) 返回字符串str中最左边的x个字符

length(s) 返回字符串s中的字符数

position(substr,str) 返回字串substr在字符串str中第一次出现的位置

quote(str) 是用反斜杠转义str中的单引号

trim(str) 去除首部和尾部的所有空格

right(str,x) 返回字符串str中最右边的x个字符

ucase(str) upper(str) 返回字符串str中所有字符的大写结果

curdate() current\_date() 返回当前日期

curtime() current\_time() 返回当前时间

date\_add(date,interval int keyword) 日期date时间加上间隔int的结果

date\_sub(date, interval int keyword) 日期date时间减去间隔int的结果

now() 返回当前日期和时间

select now();

19.方法例子

<script language="javascript">

function check(f)

{

if(f.user.value == "")

{

alert("请输入用户名！");

f.user.focus();

return (false);

}

if(f.pass.value == "")

{

alert("请输入用户密码！");

f.pass.focus();

return (false);

}

if(f.pass.value != f.pass2.value)

{

alert("重复密码与密码不一致");

f.pass2.focus();

return (false);

}

if(f.age.value =="")

{

alert("请输入年龄");

f.age.focus();

return (false);

}

if(f.mail.value =="")

{

alert("请输入电子邮箱");

f.mail.focus();

return (false);

}

}

</script>

<style type="text/css">

<!--

...

-->

</style>

20.php连接mysql创建表语句

<?php

$con=mysql\_connect("localhost","root","");

if($con)

{

mysql\_select\_db("test",$con);

#sql="create table user\_data

(

id int(5) not null auto\_increment primary key,

name char(10) not null default '',

password char(12) not null default '',

age int(3) not null default 0,

sex char(2) not null default '男',

mail char(50) not null default '',

qq char(12) not null default '',

degree char(10) not null default '',

fav char(50) not null default ''

) ";

$do=mysql\_query($sql,$con);

if($do)

{

echo "成功在test数据库中创建用户表";

}

else echo "建表时出现错误！";

}

else

{

echo "连接到服务器时出现错误！";

}

?>

21.后台页面获取数据

<?php

if($\_POST)

{

$user=$\_POST["user"];

$pass=$\_POST["pass"];

$age=$\_POST["age"];

$sex=$\_POST["sex"];

$mail=$\_POST["mail"];

$qq=$\_POST["qq"];

$degree=$\_POST["degree"];

$fav=$\_POST["fav"];

$len=count($fav);

$fav\_z=" ";

for($i=0; $i<$len;$i++)

{

$fav\_z=$fav\_z.$fav[$i];

if($i<$len-1) $fav\_z=$fav\_z.",";

}

$con=mysql\_connect("localhost","root","");

mysql\_select\_db("test");

mysql\_query("set names GB2312");

$sql="select count(\*) from user\_data where name='$user' ";

或者$sql="select id from user\_data where name='$user' and password='$pass'";

$result=mysql\_query($sql);

$num=mysql\_fetch\_row($result);

或者$num=mysql\_num\_rows($result);

if($num[0]>0)

{

echo "存在同名用户，请您重新选择用户名";

}

else

{

$sql="insert into user\_data(name,password,age,sex,mail,qq,degree,fav) values('$user','$pass','$age','$sex','$mail','$qq','$degree','$fav\_z')";

$re=mysql\_query($sql);

if($re) echo "成功插入记录";

else echo "插入记录失败";

echo "<p>";

}

或者

if($num>0)

{

echo "成功登录!".$user."<p>";

echo "点击这里进入主界面";

}

else

{

echo "输入的用户名或者密码有误！";

echo "<p>";

echo "点<a href=\"login.html\">这里</a>重新登录";

echo "<p>";

echo "如果还没有注册，点<a href=\"reg.html\">这里</a>进行注册";

}

}

else

{

echo "没有任何提交内容！<br>";

}

echo "<br>点<a href='reg.html'>这里</a>返回";

?>

22.模拟下载文件

建表

<?php

$con=mysql\_connect("localhost","root","");

mysql\_select\_db("test",$con);

mysql\_query("set names GB2312");

$sql="create table 'file\_resource'

(

id int(5) not null auto\_increment primary key,

file\_name char(30) not null default '',

show\_name char(40) not null default '',

role\_name char(40) not null default '',

down\_num int(5) not null default 0,

introduce char(200) not null default ''

)";

$do=mysql\_query($sql,$con);

if($do)

{

echo "成功在test数据库中创建用户表！<p>";

$sql="insert into 'file\_resource' ('id','file\_name','show\_name','role\_name','down\_num','introduce') values(1,'dragon.rmvb','火龙对决','黎明/任贤齐/王宝强/叶璇',0,'影片介绍，故事情节，发展主线，错综纠缠，处理方法，故事结尾')"；

$ins=mysql\_query($sql);

if($ins) echo "成功插入记录!";

else echo "插入记录失败!";

}

else echo "建表时出现错误！";

?>

文件下载前台

<?php

$con=mysql\_connect("localhost","root","");

mysql\_select\_db("test");

mysql\_query("set names GB2312");

$sql="select \* from file\_resource order by down\_num desc";

$result=mysql\_query($sql,$con);

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

<!--

...

-->"

echo "<h2>最新电影下载排行榜</h2>";

echo "<table border=\"1\" cellspacing=\"0\" cellpadding=\"1\" bordercolordark=\"#ffffff\" bordercolorlight=\"#ff0000\" width=\"680\">";

echo "<tr bgcolor=\"#ffcccc\">

<td width=\"%15\">名称</td>

<td width=\"%20\">主演</td>

<td width=\"%10\">文件名</td>

<td width=\"%5\">下载次数</td>

<td width=\"%45\">介绍</td>

<td width=\"%5\">下载</td>

</td>"

while($row=mysql\_fetch\_array($result))

{

echo "<tr>";

echo "<td>".$row[show\_name]."</td>";

echo "<td>".$row[role\_name]."</td>";

echo "<td>".$row[file\_name]."</td>";

echo "<td>".$row[down\_name]."</td>";

echo "<td>".$row[introduce]."</td>";

echo "<td><a href=\"download.php?id=".$row[id]."&show\_name=".$row[show\_name]."\">下载</a></td>";

echo "</tr>";

}

echo "</table>";

?>

文件下载后台

<?php

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

<!--

...

-->

</style>";

if($\_GET[id] and $\_GET[show\_name])

{

$id=$\_GET[id];

$show\_name=$\_GET[show\_name];

echo "点击这里开始下载";

echo "<a href=\"javascript:alert('呵呵！')\">".$show\_name."</a>";

$con=mysql\_connect("localhost","root","");

mysql\_select\_db("test");

mysql\_query("set nams GB2312");

$sql="update file\_resource set down\_num=down\_num+1 where id='$id'";

mysql\_query($sql,$con);

echo "<p>";

echo "点<a href=\"list.php\">这里</a>返回上一页";

}

else

{

echo "没有提供ID值，或者电影名称！<p>无法完成下载";

}

?>

23.mkdir(string $pathname[,int $mode[,bool $recursive[,resource $content]]]) 创建目录

rmdir(string $dirname) 删除目录

opendir(string $path[,resource $content]) 打开目录

readdir(resource $dir\_handle) 读取目录内的内容

rewinddir(resource $dir\_handle) 将句柄重新指向目录的开头

closedir(string $pathname) 关闭打开目录句柄以释放其占用的内存空间

is\_dir(string $filename) 判断文件是否为目录

id\_dir() 判断文件是否为目录并递归

disk\_total\_space(string $directory) 根据相应的文件系统或者磁盘分区返回所有的字节数

disk\_free\_space(string $directory) 根据相应的文件系统或者磁盘分区返回所有可用的字节数

file\_exists(string $filename) 判断文件是否存在

fopen(string $filename,string $mode[,bool $use\_include\_path[,resource $zcontext]]) 打开文件

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 参数 | 读写方式 | 指针位置 | 是否清空原文件 | 文件不存在是否创建 |
| r | 只读 | 文件头 | 否 | 否 |
| r+ | 读写 | 文件头 | 否 | 否 |
| w | 可写 | 文件头 | 是 | 是 |
| w+ | 读写 | 文件头 | 是 | 是 |
| a | 可写 | 文件尾 | 否 | 否 |
| a+ | 读写 | 文件尾 | 否 | 否 |
| x | 写入 | 文件头 | 否 | 是，文件存在调用失败 |
| x+ | 读写 | 文件头 | 否 | 是，文件存在调用失败 |

bool fclose(resource $handle) 关闭文件 $handle为文件打开的操作句柄

fgetc(resource $handle) 按字符读取文件 $handle为文件打开的操作句柄

fgets(resource $handle[,int length]) 按行读取文件

array file(string $filename[,int $user\_include\_path[,resource $context]])

将文件读入数组

string file\_get\_contents(string $filename[,bool $use\_include\_path[,resource $context[,int $offset[,int $maxlen]]]])

将文件读入一个字符串中

int fwrite(resource $handle,string $string[,int $length])

将字符串写入文件

int file\_put\_contents(string $filename,string $data[,int $flags[,resource $context]]) 将指定内容写入文件

$flags的值有file\_use\_include\_path,file\_append,lock\_ex(获得一个独占锁定)

bool rewind(resource $handle) 倒回文件指针

int filesize(string $filename) 获取文件的大小

filectime() 获取文件创建时间

filemtime() 获取文件修改时间

fileatime() 获取文件最后访问时间

eg：$file="test3.txt";

date("Y-m-d H:i:s",filectime($file))

date("Y-m-d H:i:s",filemtime($file))

date("Y-m-d H:i:s",fileatime($file))

bool copy(string $source,string $dest) 复制文件

bool rename(string $oldname,string $newname[,resource $context]) 重命名文件

bool unlink(string $filename) 删除文件

24.文件上传

html页面名称为up\_forward.php

文件上传后台

<?php

if(!$\_POST["upfile"] && $\_FILES["upfile"]["name"] == "")

{

echo "没有选择文件";

echo "<p>";

echo "点<a href=\"up\_forward.html\">这里</a>返回";

}

else

{

$filepath="upload/";

$name=$filepath.$\_FILES["upfile"]["name"];

while(file\_exists($name))

{

$temp=explode(".",$name);

$name=$temp[0]."0".".".$temp[1];

}

if(move\_uploaded\_file($\_FILES["upfile"]["tmp\_name"],$name))

{

if($\_POST["owner"])

{

$owner=$\_POST["owner"];

}

else

{

$owner="无名氏";

}

if($\_POST["describe"])

{

$describe=$\_POST["describe"];

}

else

{

$describe="没有描述";

}

$time=date("Y-m-d H:i:s");

$content=$\_FILES["upfile"]["name"]."||".$owner."||".$describe."||".$time."\n";

file\_put\_contents("record.dat",$content,FILE\_APPEND);

echo "名为：".$\_FILES["upfile"]["name"];

echo "<p>";

echo "的文件成功上传";

echo "点<a href=\"up\_forward.html\">这里</a>返回！";

}

else

{

echo "文件上传过程中出现错误！";

echo "<p>";

echo "文件上传失败";

echo "<p>";

echo "点<a href=\"index.php\">这里</a>查看！";

}

}

?>

25.下载模块

使用header() 函数

<?php

$filename="upload/".$\_GET['name'];

header("Content-Type:application/force-download");

header("Content-Dispositon:"atachment;filename=".basename($filename));

readfile($filename);

?>

26.bool setcookie(string $name[,string $value[,int $expire[,string $path[,string $domain[bool $secure[,bool $httponly]]]]]])

创建cookie

$\_cookie["name"] 读取cookie

setcookie("name","");

setcookie("name");这两种均可删除cookie

void unset(mixed $var...) 释放给定的变量

eg：unset($\_cookie["work"])

session\_start() 创建一个session会话

session\_destory() 销毁所有的session数据

eg：$\_session['favcolor']='绿';

$\_session['animal']='猫';

$\_session['time']=time();

string session\_name([string $name]) 获取或设置session名称

string session\_id([string $id]) 获取或设置session ID

$lifetme=24\*3600;

session\_set\_cookie\_params($lifetime) 设置session的生存周期

isset($\_cookie["num"]) 判断是否存在cookie

session\_register()和session\_unregister()对应

27.gd\_info() 获取GD库信息

array getimagesize(string $filename[,array &$imageinfo])

int imagesx(resource $image) 获取图像的宽度

int imagesy(resource $image) 获取图像的高度

string image\_type\_to\_extension(int $imagetype[,bool $include\_dot])

获取图像类型后缀

resource imagecreate(int $x\_size,int $y\_size) 创建图像

resource imagecreatefromgif(string $filename) 从gif文件或url新建一个图像

resource imagecreatefrompng(string $filename) 从png文件或url新建一个图像

resource imagecreatefromjpg(string $filename) 从jpeg文件或url新建一个图像

resource imagecreatefromstring(string $image) 从string文件或url新建一个图像

resource imagecreatetruecolor(int $x\_size,int $y\_size) 创建一个真彩色图像资源

bool imagegif(resource $image[,string $filename]) 以gif图像输出到浏览器或文件

bool imagejpeg(resource $image[,string $filename[,int $quality]]) 以jpeg图像输出到浏览器或文件

bool imagepng(resource $image[,string $filename]) 以png图像输出到浏览器或文件

bool imagedestory(resource $image) 销毁图像

**int imagecolorallocate(resource $image,int $red,int $green,int $blue) 设定绘图颜色**

bool imagesetpixel(resource $image,int $x,int $y,int $color) 画一个像素点

bool imageline(resource $image,int $x1,int $y1,int $x2,int $y2,int $color)

画线段

bool imagerectangle(resource $image,int $x1,int $y1,int $x2,int $y2,int $col) 画矩形

bool imageellipse(resource $image,int $cx,int $cy,int $w,int $h,int $color)

画椭圆

bool imagearc(resource $image,int $cx,int $cy,int $w,int $h,int $s,int $e,int $color) 画椭圆弧

bool imagepolygon(resource $image,array $points,int $num\_points,int $color) 画多边形

bool imagestring(resource $image,int $font,int $x,int $y, int $s,int $col)

画字符串

array imagettftext(resource $image,float $size,float $angle,int $x,int $y,int $color,string $fontfile,string $text) 画中文字符串

bool imagecopy(resource $dst\_im,resource $src\_im,int $dst\_x,int $dst\_y,int $src\_x,int $src\_y,int $src\_w,int $src\_h) 复制图像的一部分

图片缩略图处理模块

。。。

imagefilledarc() 画饼形图

图像加水印

。。。

28.PHP (PHP Hypertext Preprocessor)

29.系统函数

Number abs(number) 取绝对值

Float sin(float) 正弦计算

Float Cos(float) 余弦计算

Float Sqrt(float) 开平方根计算

Float Log(float) 自然对数计算

Float Log10(float) 10基底的对数值

Float Exp(float) 自然对数e的次方值

Float Pow(float x,float y) 次方x^y

Double pi() 圆周率

Void srand(int) 配置随机数种子

Int rand([int min],[int max]) 取得随机数值

30.string strtolower(strng str) 字符串全转为小写

String strtoupper(string str) 字符串全转为大写

String trim(string str) 截取字符串首尾的空格

String substr(string string,int start,int[length]) 取子串

本函数将字符串string的第start位起的字符串取出length个字符，若start为负数，则从字符串尾端算起，若可省略的参数length存在，但为负数，则表示取到倒数第length个字符，下标从0开始。

Int strlen(string str) 取得字符串长度

Int strops(string 母串,string 待查字符,int [offset]) 寻找字符串中某字符最先出现处

Int strcmp(string str1,string str2) 字符串比较

31.string md5(string str) 加密

Int filesize(string filename) 获得文件的大小

Int feof(int fp) 测试文件指针是否指到文件尾

Int fopen(string filename,string mode) 打开文件或者URL

Int fclose(int fp) 关闭已打开的文件

String fread(int fp,int n) 读取文件n个字节

String fgets(int fp,int n) 取得我呢见指针所指的行

Int fwrite(int fp,string string string,int [n]) 写入文件

Int fputs(int fp,string str,int [n]) 写入文件

32.rpm安装 mysql数据库

Rpm –ivh mysql

Tar xzfv mysql-x.xx.tar.gz 解压

Ed mysql-x.xx

./configure 生成

Make 编译

Make install 安装

mysql\_install\_db

33.select hex(‘cat’); 十六进制 ‘636174’

Create table t(‘my order’ char(10))

Create tablet(a decimal(5,2));

34.date(“Y-m-d”)

Mktime(hour,minute,second,month,day,year,is\_dst)

$tomorrow=Mktime(0,0,0,date(“m”),date(“d”)+1,date(“Y”)+1)

Echo date(“Y-m-d”,$tomorrow);

35.include ‘header.php’ require ‘menu.php’

36.$file = fopen(“welcome.txt”,“r”);

R 只读,在文件的开头开始

R+ 读/写，在文件的开头开始

W 只写，打开并清空文件的内容，如果文件不存在，则创建新文件

W+ 读/写，打开并清空文件的内容，如果文件不存在，则创建新文件

A 追加，打开并向文件的末端进行写操作，如果文件不存在，则创建新文件

A+ 读/追加，通过向文件末端写内容，来保持文件内容

X 只写，创建新文件，如果文件已存在，则返回false

X+ 读/写，创建新文件，如果文件已存在，则返回false和一个错误

如果fopen()无法打开指定文件，则返回0(false)

$file = fopen(“welcome.txt”,”r”) or exit(“unable to open file!”);

Fclose($file);

Feof($file) 检测是否已达到文件的末端

Fgets($file)用于从文件中逐行读取文件

Fgetc($file)用于从文件逐字符的读取文件

37.上传文件

<html>

<body>

<form action="upload\_file.php" method="post"

enctype="multipart/form-data">

<label for="file">Filename:</label>

<input type="file" name="file" id="file" />

<br />

<input type="submit" name="submit" value="Submit" />

</form>

</body>

</html>

"upload\_file.php"

* $\_FILES["file"]["name"] - 被上传文件的名称
* $\_FILES["file"]["type"] - 被上传文件的类型
* $\_FILES["file"]["size"] - 被上传文件的大小，以字节计
* $\_FILES["file"]["tmp\_name"] - 存储在服务器的文件的临时副本的名称
* $\_FILES["file"]["error"] - 由文件上传导致的错误代码
* **上传限制**
* 在这个脚本中，我们增加了对文件上传的限制。用户只能上传 .gif 或 .jpeg 文件，文件大小必须小于 20 kb：

**注释：**对于 IE，识别 jpg 文件的类型必须是 pjpeg，对于 FireFox，必须是 jpeg。

<?php

if ((($\_FILES["file"]["type"] == "image/gif")

|| ($\_FILES["file"]["type"] == "image/jpeg")

|| ($\_FILES["file"]["type"] == "image/pjpeg"))

&& ($\_FILES["file"]["size"] < 20000))

{

if ($\_FILES["file"]["error"] > 0)

{

echo "Return Code: " . $\_FILES["file"]["error"] . "<br />";

}

else

{

echo "Upload: " . $\_FILES["file"]["name"] . "<br />";

echo "Type: " . $\_FILES["file"]["type"] . "<br />";

echo "Size: " . ($\_FILES["file"]["size"] / 1024) . " Kb<br />";

echo "Temp file: " . $\_FILES["file"]["tmp\_name"] . "<br />";

if (file\_exists("upload/" . $\_FILES["file"]["name"]))

{

echo $\_FILES["file"]["name"] . " already exists. ";

}

else

{

move\_uploaded\_file($\_FILES["file"]["tmp\_name"],

"upload/" . $\_FILES["file"]["name"]);

echo "Stored in: " . "upload/" . $\_FILES["file"]["name"];

}

}

}

else

{

echo "Invalid file";

}

?>

38.设置cookie

Setcookie(“keyName”,”value”,time()+3600);

Echo $\_COOKIE[“keyName”]

If(isset($\_COOKIE[“keyName”])){

Echo “welcome” . $\_COOKIE[“keyName”] . “<br/>”

}else{

Echo “welcome guest!<br/>”

}

39.session

注释：session\_start() 函数必须位于 <html> 标签之前：

<?php

session\_start();

if(isset($\_session[‘views’]))

$\_session[‘views’] = $\_session[‘views’] + 1;

Else

$\_session[‘views’] = 1;

?>

<html>

<head><titl></title></head>

<body>

<?php

Echo “Pageviews=” . $\_session[‘views’] ;

?>

</body>

</html>

终结 session

<?php

Unset($\_session[‘views’]);

?>

或者

<?php

Session\_destroy();

?>

40.

### 语法

mail(to,subject,message,headers,parameters)

|  |  |
| --- | --- |
| **参数** | **描述** |
| to | 必需。规定 email 接收者。 |
| subject | 必需。规定 email 的主题。注释：该参数不能包含任何新行字符。 |
| message | 必需。定义要发送的消息。应使用 LF (\n) 来分隔各行。 |
| headers | 可选。规定附加的标题，比如 From、Cc 以及 Bcc。  应当使用 CRLF (\r\n) 分隔附加的标题。 |
| parameters | 可选。对邮件发送程序规定额外的参数。 |

<?php

$to = [someone@example.com](mailto:someone@example.com);

$subject = “Test mail”;

$message = “Hello! This is a simple email message.”;

$from = [someonelse@example.com](mailto:someonelse@example.com);

$headers = “Form:$from”;

Mail($to,$subject,$message,$headers);

Echo “Mail Sent.”;

?>

<?php

If(isset($\_REQUEST[‘Eemail’]))

{

$email = $\_request[‘email’];

$subject = $\_request[‘subject’];

$message = $\_request[‘message’];

Mail(“someone@example.com”,”subject:$subject”,$message,”From:$email”);

Echo “Thank you for using our mail form”;

}else{

Echo “<form method = ‘post’ action = ‘mailform.php’>

Email: <input name=’email’ type=’text’/><br/>

Subject:<input name=’subject’ type=’text’/><br/>

Message:<br/>

<textarea name=’message’ rows=’15’ cols=’40’>

</textarea><br/>

<input type=’submit’/>

</form>”;

}

?>

为了防止注入

<?php

Function spamcheck($field)

$field = filter\_var($field,FILTER\_SANITIZE\_EMAIL);

If(filter\_var($field,FILTER\_VALIDATE\_EMAIL))

{

Return true;

}

Else

{

Return false;

}

If(isset($\_REQUEST[‘Eemail’]))

{

$mailcheck = spamcheck($\_request[‘email’]);

If($mailcheck == false)

{

Echo “invalid input”;

}

Else

{

$email = $\_request[‘email’];

$subject = $\_request[‘subject’];

$message = $\_request[‘message’];

Mail(“someone@example.com”,”subject:$subject”,$message,”From:$email”);

Echo “Thank you for using our mail form”;

}

}else{

Echo “<form method = ‘post’ action = ‘mailform.php’>

Email: <input name=’email’ type=’text’/><br/>

Subject:<input name=’subject’ type=’text’/><br/>

Message:<br/>

<textarea name=’message’ rows=’15’ cols=’40’>

</textarea><br/>

<input type=’submit’/>

</form>”;

}

?>

41.错误处理机制

<?php

If(!file\_exists(“welcome.txt”))

{

Die(“file not found”);

}

Else

{

$file=fopen(“welcome.txt”,”r”);

}

?>

### 语法

error\_function(error\_level,error\_message,

error\_file,error\_line,error\_context)

|  |  |
| --- | --- |
| **参数** | **描述** |
| error\_level | 必需。为用户定义的错误规定错误报告级别。必须是一个值数。  参见下面的表格：错误报告级别。 |
| error\_message | 必需。为用户定义的错误规定错误消息。 |
| error\_file | 可选。规定错误在其中发生的文件名。 |
| error\_line | 可选。规定错误发生的行号。 |
| error\_context | 可选。规定一个数组，包含了当错误发生时在用的每个变量以及它们的值。 |

### 错误报告级别

这些错误报告级别是错误处理程序旨在处理的错误的不同的类型：

|  |  |  |
| --- | --- | --- |
| **值** | **常量** | **描述** |
| 2 | E\_WARNING | 非致命的 run-time 错误。不暂停脚本执行。 |
| 8 | E\_NOTICE | Run-time 通知。  脚本发现可能有错误发生，但也可能在脚本正常运行时发生。 |
| 256 | E\_USER\_ERROR | 致命的用户生成的错误。这类似于程序员使用 PHP 函数 trigger\_error() 设置的 E\_ERROR。 |
| 512 | E\_USER\_WARNING | 非致命的用户生成的警告。这类似于程序员使用 PHP 函数 trigger\_error() 设置的 E\_WARNING。 |
| 1024 | E\_USER\_NOTICE | 用户生成的通知。这类似于程序员使用 PHP 函数 trigger\_error() 设置的 E\_NOTICE。 |
| 4096 | E\_RECOVERABLE\_ERROR | 可捕获的致命错误。类似 E\_ERROR，但可被用户定义的处理程序捕获。(参见 set\_error\_handler()) |
| 8191 | E\_ALL | 所有错误和警告，除级别 E\_STRICT 以外。  （在 PHP 6.0，E\_STRICT 是 E\_ALL 的一部分） |

自定义错误处理机制

<?php

Function customError($errno,$errstr)

{

Echo “<b>error:</b> [$errno] $errstr”;

}

Set\_error\_handler(“customError”);

// Set\_error\_handler(“customError”,E\_USER\_WARNING);

Echo($test);

Trigger\_error(“Value must be 1 or below”);

Trigger\_error(“Value must be 1 or below”,E\_USER\_EARNING);

?>

* E\_USER\_ERROR - 致命的用户生成的 run-time 错误。错误无法恢复。脚本执行被中断。
* E\_USER\_WARNING - 非致命的用户生成的 run-time 警告。脚本执行不被中断。
* E\_USER\_NOTICE - 默认。用户生成的 run-time 通知。脚本发现了可能的错误，也有可能在脚本运行正常时发生。
* 我们开始创建 exception 类：
* <?php
* class customException extends Exception
* {
* public function errorMessage()
* {
* //error message
* $errorMsg = 'Error on line '.$this->getLine().' in '.$this->getFile()
* .': <b>'.$this->getMessage().'</b> is not a valid E-Mail address';
* return $errorMsg;
* }
* }
* $email = "someone@example...com";
* try
* {
* //check if
* if(filter\_var($email, FILTER\_VALIDATE\_EMAIL) === FALSE)
* {
* //throw exception if email is not valid
* throw new customException($email);
* }
* }
* catch (customException $e)
* {
* //display custom message
* echo $e->errorMessage();
* }
* ?>

Filter\_var()通过一个指定的过滤器来过滤单一的变量

Filter\_var\_array()通过相同的或不同的过滤器来过滤多个变量

Filter\_input获取一个输入变量，并对它进行过滤

Filter\_input\_array获取多个输入变量，并通过相同的或不同的过滤器对它们进行过滤