核心编程第2天

[学习目标 2](#_Toc521255055)

[PHP操作目录 2](#_Toc521255056)

[1、目录操作概述 2](#_Toc521255057)

[2、创建新目录mkdir() 2](#_Toc521255058)

[3、判断是否是一个目录is\_dir() 3](#_Toc521255059)

[4、判断目录或文件是否存在file\_exists() 3](#_Toc521255060)

[5、删除目录rmdir() 3](#_Toc521255061)

[6、更改目录的访问权限chmod() 4](#_Toc521255062)

[7、取得目录或文件访问权限fileperms() 4](#_Toc521255063)

[8、重命名或移动文件或目录rename() 5](#_Toc521255064)

[9、打开目录opendir() 5](#_Toc521255065)

[10、读取目录中条目readdir() 6](#_Toc521255066)

[11、显示中文目录或文件iconv() 6](#_Toc521255067)

[12、关闭目录句柄closedir() 7](#_Toc521255068)

[综合实例：递归遍历phpMyAdmin下的所有条目 8](#_Toc521255069)

[1、递归思想 8](#_Toc521255070)

[2、递归实现的条件 8](#_Toc521255071)

[3、递归遍历phpMyAdmin目录下所有条目 8](#_Toc521255072)

[PHP操作文件 10](#_Toc521255073)

[1、打开文件fopen() 10](#_Toc521255074)

[2、打开文件的方式 10](#_Toc521255075)

[3、关闭文件fclose() 11](#_Toc521255076)

[4、读取指定大小文件内容fread() 11](#_Toc521255077)

[5、读取一行内容fgets() 12](#_Toc521255078)

[6、读取文件内容到数组中file() 12](#_Toc521255079)

[7、读取文件内容到字符串中file\_get\_contents() 13](#_Toc521255080)

[8、实例：读取记事本中的内容，并用表格展示出来 13](#_Toc521255081)

[9、写入文件 14](#_Toc521255082)

[10、将一个字符串写入文件 14](#_Toc521255083)

[11、拷贝文件 15](#_Toc521255084)

[12、删除文件 15](#_Toc521255085)

[13、其它的文件操作函数 15](#_Toc521255086)

[综合实例：递归删除phpMyAdmin目录 16](#_Toc521255087)

# 学习目标

* PHP操作目录
* PHP操作文件

# PHP操作目录

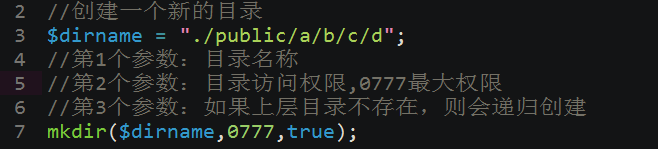
## 1、目录操作概述

例如，百度网盘(http://pan.baidu.com) ，就是经典的网页版文件管理系统。



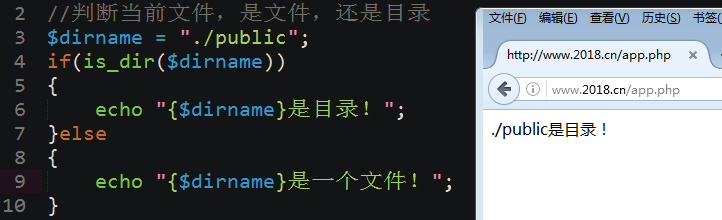
## 2、创建新目录mkdir()

* 描述：尝试新建一个由 pathname 指定的目录。
* 语法：**bool mkdir ( string $pathname [, int $mode = 0777 [, bool $recursive = false ]] )**
* 参数：
  + $pathname：指定目录的路径；
  + $mode：默认的 mode 是 0777，意味着最大可能的访问权。$mode在windows下被忽略。 注意$mode的值为八进制。
  + $recursive：如果指定的路径的上级目录不存在，则也会递归创建。
* 返回值：成功时返回TRUE，失败时返回FALSE。



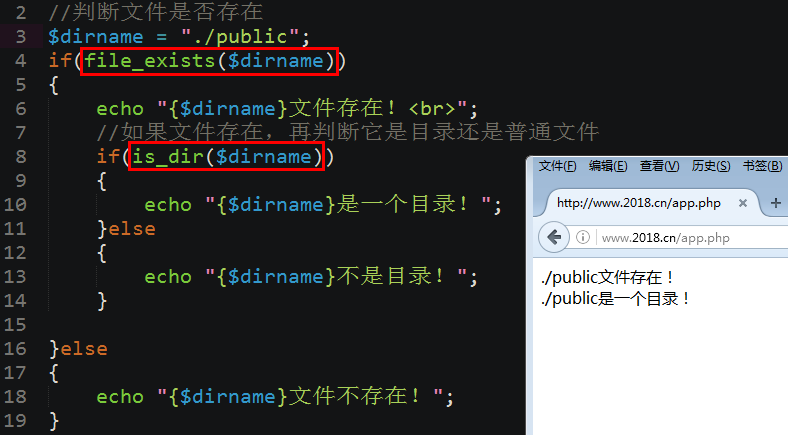
## 3、判断是否是一个目录is\_dir()

* 描述：判断给定文件名是否是一个目录。
* 语法：**bool is\_dir ( string $filename )**
* 返回值：如果文件名存在，并且是个目录，返回 TRUE，否则返回FALSE。



## 4、判断目录或文件是否存在file\_exists()

* 语法：**bool file\_exists ( string $filename )**
* 提示：可以判断文件，也可以判断目录，是否存在。
* 返回：如果由 filename 指定的文件或目录存在则返回 TRUE，否则返回 FALSE。



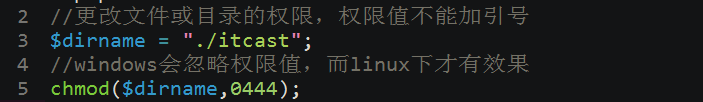
## 5、删除目录rmdir()

* 语法：**bool rmdir ( string $dirname )**
* 参数：$dirname目录路径。
* 返回值：成功时返回 TRUE， 或者在失败时返回 FALSE。
* 提示：尝试删除 dirname 所指定的目录。 该目录必须是空的，且要有相应的权限。



## 6、更改目录的访问权限chmod()

* 描述：改变文件的访问权限
* 语法：**bool chmod ( string $filename , int $mode )**
* 参数：
  + $filename，指定文件的路径；
  + $mode，mode 参数包含三个八进制数按顺序分别指定了所有者、所有者所在的组以及所有人的访问限制。每一部分都可以通过加入所需的权限来计算出所要的权限。数字 1 表示使文件可执行，数字 2 表示使文件可写，数字 4 表示使文件可读。加入这些数字来制定所需要的权限。
  + 提示：在window下文件的只读权限是 0444。文件夹只读权限0555。
* 注意：**此函数不能作用于远程文件**。



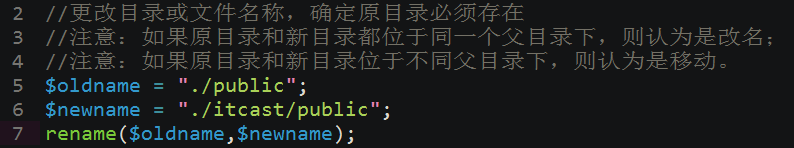
## 7、取得目录或文件访问权限fileperms()

* 语法：**int fileperms ( string $filename )**
* 返回值：以十进制数字返回文件的访问权限。



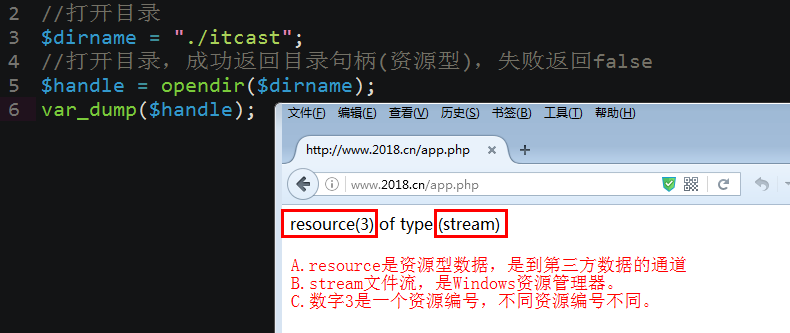
## 8、重命名或移动文件或目录rename()

* 描述：重命名或移动一个文件或目录。如果文件不存在同一目录下，就是移动。
* 语法：**bool rename ( string $oldname , string $newname )**
* 返回值：成功时返回 TRUE， 或者在失败时返回 FALSE。



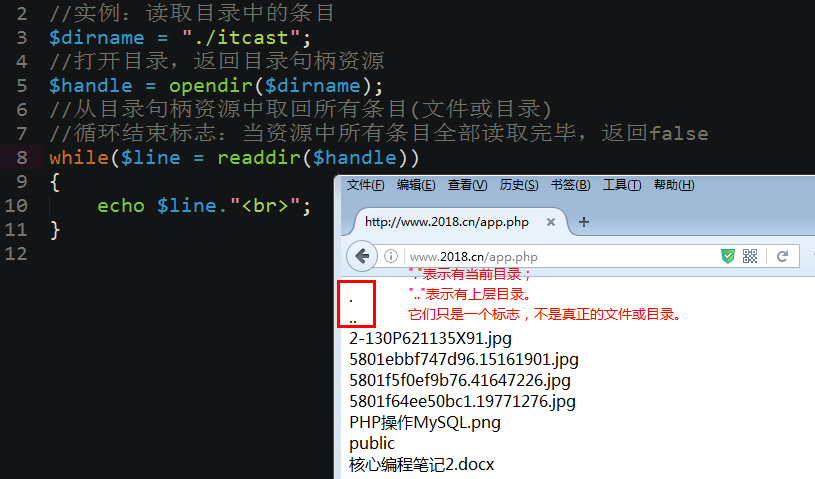
## 9、打开目录opendir()

* 描述：打开目录句柄，可用于之后的 closedir()，readdir() 和 rewinddir() 调用中。
* 语法：**resource opendir ( string $path )**
* 返回值：如果成功则返回目录句柄的 resource，失败则返回 FALSE。



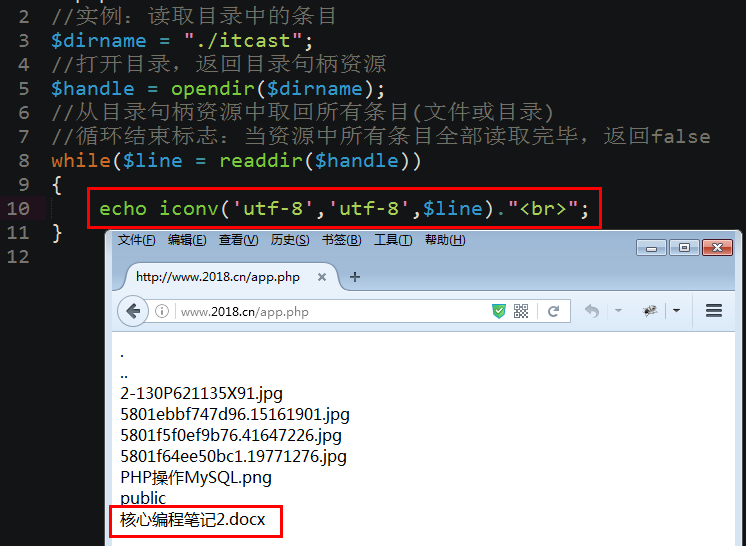
## 10、读取目录中条目readdir()

* 描述：从目录句柄中读取条目 ;
* 语法：**string readdir ([ resource $dir\_handle ] )**
* 说明：返回目录中下一个文件的文件名。文件名以在文件系统中的排序返回。
* 返回：成功则返回文件名 或者在失败时返回 FALSE 。



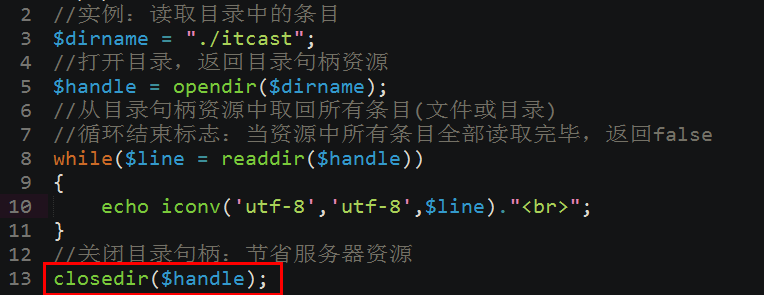
## 11、显示中文目录或文件iconv()

* 描述：字符串按要求的字符编码来转换
* 语法：**string iconv ( string $in\_charset , string $out\_charset , string $str )**
* 参数：
  + $in\_charset，输入的字符集；
  + $out\_charset，输出的字符集；
  + $str，要转换的字符串；
* 返回：返回转换后的字符串， 或者在失败时返回 FALSE。



## 12、关闭目录句柄closedir()

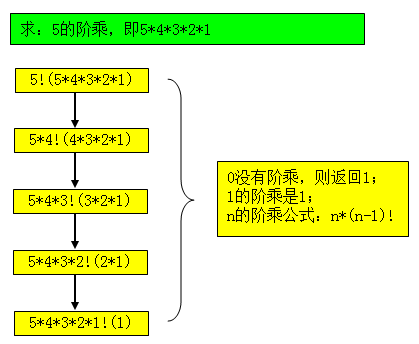
* 描述：关闭由 dir\_handle 指定的目录流。流必须之前被 opendir() 所打开。
* 语法：**void closedir ( [resource $dir\_handle] )**
* 提示：如果省略$dir\_handle，则默认为最后由opendir()打开的目录句柄。



# 综合实例：递归遍历phpMyAdmin下的所有条目

## 1、递归思想

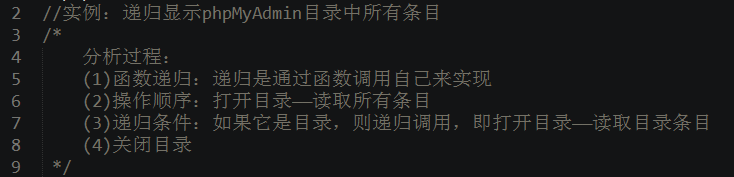
* 递归思想：把一个相对复杂的问题转化为一个与原问题相似的规模较小的问题来求解。比如：求n的阶乘n!=n\*(n-1)!。
* 递归方法只需少量的程序就可描述出解题过程所需要的多次重复计算，大大地减少了程序的代码量。但在带来便捷的同时，也会有一些缺点，即：通常用递归方法的运行效率不高。



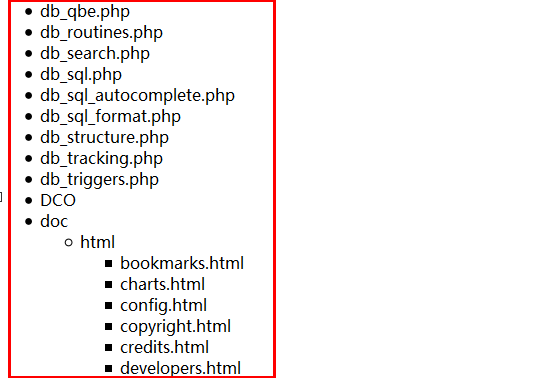
## 2、递归实现的条件

* 确定递归公式；
* 确定递归边界条件(**递归出口**)，否则会出现死循环。

## 3、递归遍历phpMyAdmin目录下所有条目



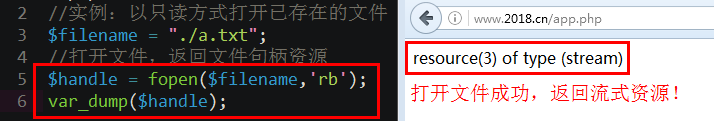




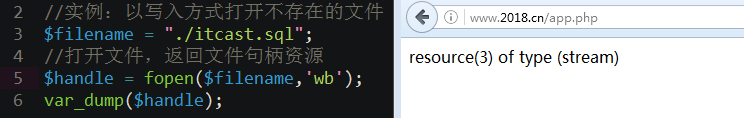
# PHP操作文件

## 1、打开文件fopen()

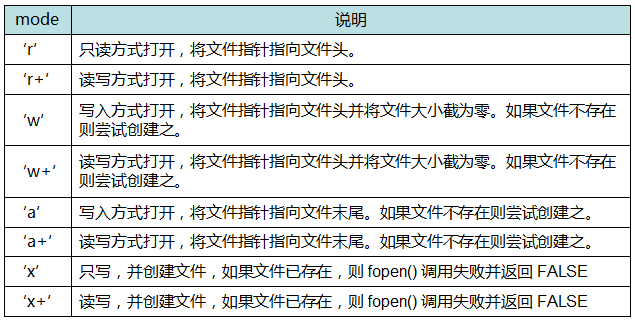
* 描述：打开文件或者 URL
* 语法：**resource fopen ( string $filename , string $mode )**
* 参数：$filename，指定要打开的目标文件或URL；$mode，指定文件的打开方式。
* 提示：为移植性考虑，在用 fopen()打开文件时总是使用 'b' 标记。binary
* 返回值：成功时返回文件指针资源，如果打开失败，本函数返回 FALSE。



以写入方式打开，它自动创建文件

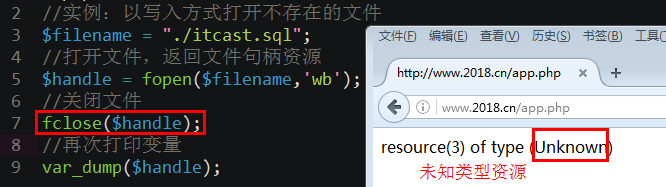


## 2、打开文件的方式



## 3、关闭文件fclose()

* 描述：关闭一个已打开的文件指针
* 语法：**bool fclose ( resource $handle )**
* 参数：$handle必须是由fopen()打开的文件指针。



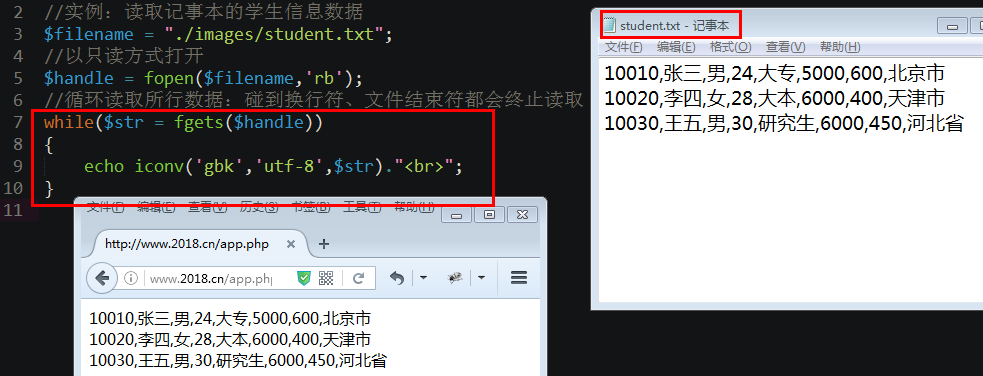
## 4、读取指定大小文件内容fread()

* 描述：读取文件（可安全用于二进制文件）
* 语法：**string fread ( resource $handle , int $length )**
* 参数：$handle是由fopen()创建的handle；$length最多读取的字节数。
* 提示：在区分二进制文件和文本文件的系统上（如 Windows）打开文件时，fopen() 函数的 mode 参数要加上 ‘b’。
* 返回值：返回所读取的字符串， 或者在失败时返回 FALSE。



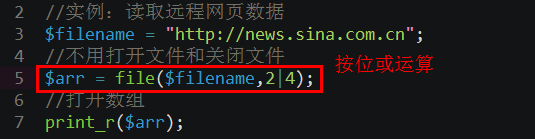
## 5、读取一行内容fgets()

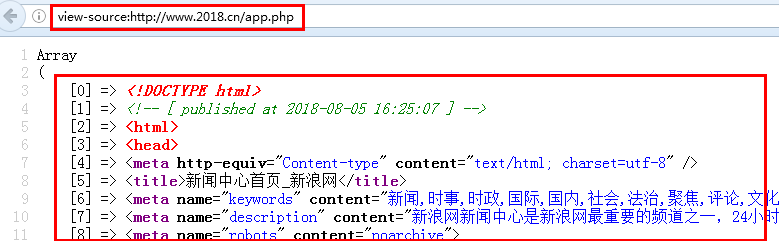
* 描述：从文件指针中读取一行
* 语法：**string fgets ( resource $handle [, int $length ] )**
* 参数：
  + $handle：文件指针必须是有效的
  + $length：从 handle指向的文件中读取一行并返回长度最多为 length - 1 字节的字符串。碰到换行符（包括在返回值中）、EOF 或者已经读取了 length - 1 字节后停止（看先碰到那一种情况）。如果没有指定 length，则默认为 1K，或者说 1024 字节。
* 返回值：从指针 handle 指向的文件中读取了 length - 1 字节后返回字符串。 如果文件指针中没有更多的数据了则返回 FALSE。



## 6、读取文件内容到数组中file()

* 描述：把整个文件读入一个数组中
* 语法：**array file ( string $filename [, int $flags = 0 ] )**
* 参数：
  + $filename，文件的路径；
  + $flags，附加选项；
  + FILE\_USE\_INCLUDE\_PATH(1)：在include\_path 中查找文件；
  + FILE\_IGNORE\_NEW\_LINES(2)：在数组的每个元素末尾不添加换行符；
  + FILE\_SKIP\_EMPTY\_LINES(4)：跳过空行。
* 注意：不需要打开文件和关闭文件。



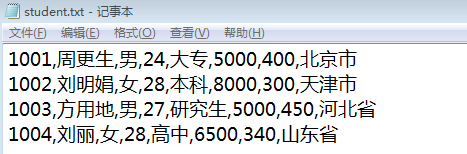


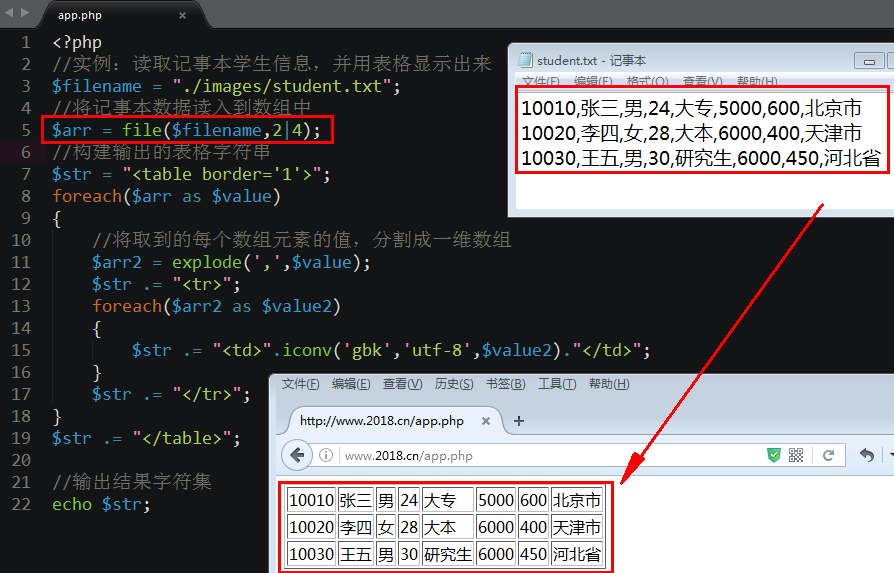
## 7、读取文件内容到字符串中file\_get\_contents()

* 描述：将整个文件读入一个字符串 ，不用打开文件，也不用关闭文件
* 语法：**string file\_get\_contents ( string $filename )**
* 返回：返回读取的数据，或在失败时返回false。



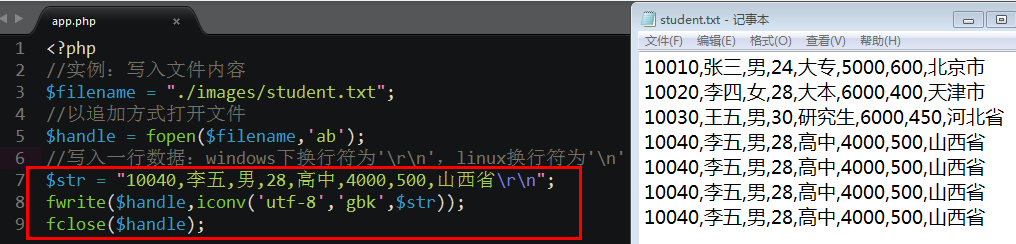
## 8、实例：读取记事本中的内容，并用表格展示出来





## 9、写入文件

* 描述：写入文件（可安全用于二进制文件）
* 语法：**int fwrite ( resource $handle , string $string )**
* 返回：返回写入的字符数，出现错误时则返回 FALSE 。



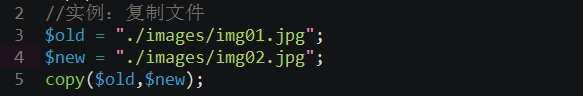
## 10、将一个字符串写入文件

* 描述：和依次调用 fopen()，fwrite()以及 fclose() 功能一样。
* 语法：**int file\_put\_contents ( string $filename , mixed $data [, int $flags = 0] )**
* 参数：
  + $filename要被写入数据的文件名。
  + $data要写入的数据。类型可以是 string，array(一维数组)。
  + $flags附加选项。
  + FILE\_USE\_INCLUDE\_PATH(1) 在 include 目录里搜索 filename
  + FILE\_APPEND(8)如果文件 filename已经存在，**追加数据而不是覆盖**。
  + LOCK\_EX(2)在写入时获得一个独占锁。
* 提示：**如果文件不存在，则会自动创建。如果文件过大，不能一次性处理**。
* 返回：该函数将返回写入到文件内数据的字节数，失败时返回FALSE。



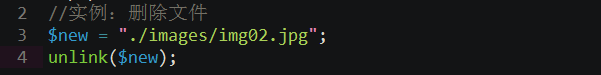
## 11、拷贝文件

* 描述：将文件从 source 拷贝到 dest。
* 语法：**bool copy ( string $source , string $dest )**
* 提示：如果目标文件存在，则会覆盖。



## 12、删除文件

* 语法：**bool unlink ( string $filename )**
* 返回：成功时返回 TRUE， 或者在失败时返回 FALSE。
* 提示：可以删除虚拟空间之外的文件，但必须使用相对路径(../)，但必须有权限。
* 注意：**删除的文件不会进入回收站**。



## 13、其它的文件操作函数

* filesize()，可以获取文件大小，单位为字节。
* is\_writable()，判断文件是否可写。
* is\_readable()，判断文件是否可读。
* feof()，判断文件指针是否到达文件结尾。
* filectime()，获取创建文件的时间。
* fileatime()，获取文件最新访问时间。
* filemtime()，获取文件最后修改时间。

# 综合实例：递归删除phpMyAdmin目录

