

مدیریت فایل‌ها و پوشه‌ها

واحد کار هشتم

پس از مطالعه این واحد کار از فراگیر انتظار می‌رود که:

- 1 - اصول کار با فایل‌ها را بشناسد.
- 2 - بتواند یک فایل را باز کند.
- 3 - بتواند یک فایل را ببندد.
- 4 - بتواند یک فایل را نمایش دهد.
- 5 - بتواند اطلاعات را از فایل بخواند یا در آن بنویسد.
- 6 - اصول کار با پوشه‌ها را بداند.
- 7 - بتواند یک پوشه جدید ایجاد کند.
- 8 - بتواند یک پوشه را حذف کند.
- 9 - بتواند فایل‌ها را از طرف سرویس گیرنده بارگذاری نماید.

زمان (ساعت)	
عملی	نظری
	2

مقدمه

در اغلب زبان‌های برنامه‌نویسی امکان انجام مدیریت روی فایل‌ها هنگام اجرا وجود دارد، در زبان PHP نیز امکانات مختلفی برای دسترسی به فایل‌ها وجود دارد. مهم‌ترین این عملیات عبارتند از: ایجاد یک فایل، به‌روزرسانی یک فایل، حذف یک فایل و ...

1-8 کار با فایل‌ها

1-1-8 نوشتن محتویات فایل در مرورگر

یکی از پرکاربردترین اعمالی که روی فایل انجام می‌شود، عمل نوشتن اطلاعات فایل در مرورگر است.

برای این که قادر باشیم اطلاعات درون فایل را دریافت کنیم، باید ابتدا فایل را باز کنیم. ساختار دستور باز کردن فایل به صورت زیر است:

؛ ("r" و "پسوند فایل . نام فایل") = fopen (متغیر نام فایل \$

برای خواندن اطلاعات، پس از باز کردن فایل باید از دستور (متغیر نام فایل \$) fgets استفاده کنیم.

در هر فایل یک شناسه برای مشخص شدن انتهای آن وجود دارد، در PHP، رسیدن به انتهای فایل توسط دستور (متغیر نام فایل \$) feof¹ تعیین می‌شود.

حال با استفاده از سه دستور فوق می‌خواهیم فایل "Newest.txt" را باز کرده و محتویات آن را در مرورگر نمایش دهیم.

رسیدن به انتهای فایل توسط مرورگر قابل تشخیص نیست، لذا باید از شرط (نام فایل \$) feof استفاده شود. به این ترتیب که چک کنیم تا زمانی که به انتهای فایل نرسیده است عمل خواندن اطلاعات انجام شود و اطلاعات خوانده شده را توسط دستور echo در خروجی چاپ کنیم.

تابع (نام فایل \$) feof یک مقدار boolean را برمی‌گرداند یعنی دارای دو مقدار True یا False است، اگر به انتهای فایل رسیده باشیم، مقدار آن True و در غیر این صورت مقدار


آن False خواهد بود، ما قصد داریم برنامه را طوری بنویسیم که تا زمانی که به انتهای فایل نرسیده است عملیات را انجام دهد.

برای پیاده کردن این عمل باید از حلقه‌ها استفاده کنیم و چون حلقه‌ها زمانی اجرا می‌شوند که شرط مقابل آن‌ها True باشد و شرط بررسی و ادامه این برنامه توسط تابع feof False است، لذا باید از یک علامت نقیض (!) در ابتدای نام تابع استفاده کنیم، کد این برنامه به این صورت خواهد بود:

```
<?php
$file=fopen("Newest.txt","r");
while(!feof($file))
{
    echo (fgets($file));
}
?>
```

دقت کنید که در مسیر سرویس دهنده محلی شما فایل Newest.txt وجود داشته باشد، اگر این فایل در مسیر فوق موجود نیست، آن را ایجاد کرده و چند جمله داخل آن بنویسید، فایل را ذخیره کرده، از آن خارج شوید.

پس از اجرای برنامه فوق در مرورگر خواهید دید که محتویات فایل متنی Newest.txt در مرورگر ظاهر شده است.

 **نکته** در دستور fopen () اگر فایلی که به عنوان پارامتر تعیین می‌شود از قبل وجود نداشته باشد، با استفاده از این دستور ساخته خواهد شد.


یکی دیگر از دستورات خواندن از فایل، دستور fread است. تفاوت دستور fread با دستور fgets در این است که دستور fgets یک پارامتر دارد که مشخص‌کننده نام فایل است و تمام محتویات فایل مشخص شده را می‌خواند اما توسط دستور fread می‌توان تعیین کرد که تعداد مشخصی کاراکتر از فایل خوانده شود.

```
<?php
$file=fopen("Newest.txt","r");
```

```
while(!feof($file))
{
    echo (fread( $file, 10 ));
}
?>
```

2-1-8 نوشتن در فایل

به منظور نوشتن یک عبارت در فایل از دستور () fputs استفاده می‌شود. این دستور عبارت موردنظر را به محتوای فایل اضافه می‌کند.

 **نکته** قبل از اقدام به هرگونه عملیات خواندن یا نوشتن فایل باید با استفاده از دستور fopen، فایل موردنظر باز شده و در انتها نیز با استفاده از دستور fclose بسته شود.

```
<?php
$file=fopen("Newest.txt","w");
$str="This is same file with different text";
fputs($file,$str);
$file=fopen("Newest.txt","r");
while(!feof($file))
{
    echo fgets($file)."<br/>";
}
?>
```

دستور دیگری نیز برای نوشتن در فایل وجود دارد. این دستور fwrite است. دستور fwrite و fputs دو پارامتر ورودی دارند: پارامتر اول متغیری است که نام فایل را مشخص می‌کند و پارامتر دوم، عبارتی است که باید در فایل نوشته شود.

2-8 کار با پوشه‌ها

1-2-8 ایجاد پوشه جدید

دستور () mkdir یک پوشه جدید را در مسیر ریشه سرویس دهنده محلی ایجاد می‌کند.

مثال زیر در مسیر ریشه سرویس دهنده یک پوشه با نام dir1 را ایجاد می‌کند.
مثال:

```
<?php  
mkdir (dir1);  
?>
```

2-2-8 حذف پوشه

دستور () rmdir پوشه مشخص شده در پارامتر ورودی تابع را حذف می‌کند.
مثال زیر پوشه ایجاد شده در مثال قبل را حذف می‌کند.
مثال:

```
<?php  
rmdir (dir1);  
?>
```

واژه‌نامه

Make
Remove

ساختن
حذف

خلاصه مطالب

- در زبان PHP نیز مانند سایر زبان‌های برنامه‌نویسی امکان انجام اعمال مدیریتی توسط کد برنامه وجود دارد، این اعمال مدیریتی عبارتند از: ایجاد فایل یا پوشه جدید، به‌روز رسانی یک فایل، حذف یک فایل یا پوشه و ...
- از دستور `fputs()` برای نوشتن محتویات فایل در مرورگر استفاده می‌شود (خواندن از فایل و نوشتن در مرورگر).
- تابع `feof` یک مقدار Boolean را برای اینکه بدانیم به انتهای فایل رسیده‌ایم یا خیر برمی‌گرداند.
- برای نوشتن در فایل از دستورات `fwrite` و `fputs` استفاده می‌شود.
- برای ایجاد پوشه جدید از دستور `mkdir` و برای حذف آن از دستور `rmdir` استفاده می‌شود.