

學習要點:

1. 目錄操作
2. 磁盤、目錄和文件計算
3. 文件處理

三、文件處理

資源(resource)這個詞常常與可以發起輸入或輸出流的實體聯繫起來。標準輸入或輸出、文件和網絡套接字都是資源的例子。因此你會經常看到本節所介紹的很多函數都是在資源處理的上下文中討論的，而不是文件處理，這本身是因為所有這些函數都能夠與前面所述的資源結合使用。但是，由於這些函數與文件結合使用是應用中最常見的。

將數據寫入一個文件，有 3 個步驟:

1. 打開這個文件。如果文件不在，需要先創建它
2. 將數據寫入這個文件。
3. 關閉這個文件

同樣，從一個文件中讀出數據，也有 3 步驟:

1. 打開這個文件。如果這個文件不能打開，就應該意識到這一點而正確退出
2. 從文件讀取數據
3. 關閉這個文件

打開文件

要在 PHP 中打開一個文件，可以使用 `fopen()` 函數。當打開一個文件的時候，還需要指定如何使用它。也就是，文件模式。

選擇文件模式: 當打開一個文件的時候，有 3 種選擇:

1. 打開文件為了只讀、只寫或者讀和寫。
2. 如果要寫一個文件，你可能希望覆蓋所有已有的文件內容，或者僅僅將新數據追加到文件末尾。如果該文件已經存在，也可以終止程序的執行而不是覆蓋該文件。
3. 如果希望一個區分了二進制方式和純文本方式的系統上寫一個文件，還必須制定採用的方式。

函數 `fopen()` 支持以上 3 種方式的組合。

`$fp=fopen(文件路徑, 文件模式, [是否在 include_path 中搜索一個文件], [允許文件名已協議名稱開始(如 http://)])`

Fopen()函數的文件模式總結

模式	模式名稱	意義
r	只讀	文件指針置於文件開頭
r+	讀寫	文件指針置於文件開頭
w	只寫	在寫入前，刪除文件內容，將指針返回到文件開頭。如果文件不存在，則嘗試創建
w+	讀寫	在讀取或寫入前，刪除文件內容，將指針返回到文件開頭。若文件不存在則嘗試創建
a	只寫	文件指針置於文件末尾。若文件不存在，則嘗試創建。此模式成為追加 (append)
a+	讀寫	文件指針置於文件末尾。若文件不存在，則嘗試創建。此過程稱為追加到文件
b	二進制	二進制模式---用於與其他模式進行鏈接。如果文件系統能夠區分二進制文件和文本文件，你可能會使用它。 Windows 系統可以區分：而 UNIX 則不區分。推薦一直使用這個選項，以便獲得最大程度的可移植性。二進制模式是默認的模式。
t	文本	用於與其他模式結合。這個模式只是 Windows 系統下一個選項。它不是推薦選項。除非你曾經在代碼中使用 b 選項。

如果 `fopen()` 函數調用失敗，函數將返回 `false`，可以用一種對於用戶有好的方式來處理這個錯誤，可以通過抑制 PHP 的錯誤信息並且根據自己的方式給出錯誤信息。

```
@$fp=fopen("file.txt",'ab'); //ab 追加並且二進制方式
```

寫文件

在 PHP 中寫文件相對比較簡單。可以使用 `fwrite()` 或者 `fputs()` 函數。我們可以使用如下方式調用 `fwrite()`;

```
Fwrite($fp,$outputstring,[int length]); //第三個可選參數為最大字符數
```

這個函數告訴 PHP 將保存在 `$outputstring` 中的字符串寫入到 `$fp` 只向的文件

中。

`Fwrite()`函數的一個新的替換函數是 `file_put_contents()`

可以通過 PHP 的內置 `strlen()`函數獲得字符串的長度:

```
Fwrite($fp,$outputstring,strlen($outputstring));
```

當然還有一種不需要資源句柄的寫入方法: `file_put_contents()`。

關閉文件

當文件使用完畢後，應該將其關閉。應該調用 `fclose()`函數:

```
Fclose($fp);
```