

學習要點:

1. Header()函數
2. 接收及驗證數據

我們對 Web 感興趣，認為他有用的原因是其主要通過基於 HTML 的表單發布和收集信息的能力。這些表單用來鼓勵網站的反饋、進行論壇會話、收集在線訂購的郵件地址，等等。但是對 HTML 表單進行編碼只是有效接收用戶輸入的必要操作的一部分，必須由服務器端組件來處理輸入。

一、Header()函數

標頭(header)是服務器以 HTTP 協議傳 HTML 資料到瀏覽器前所送出的字符串，在標頭與 HTML 文件之間尚需空一行分隔。

1. 用於重新導向指定的 URL

```
<?php
    Header('Location:http://www.baidu.com');
?>
```

2. 用於設置頁面字符編碼

```
<?php
    Header('Content-Type: text/html; charset=utf-8');
    Echo '嘿嘿，我是中文！ 頁面編碼是 utf-8，文件也是 utf-8';
?>
```

注意: 除非啟用了輸出緩衝，否則這些命令必須在返回任何輸出之前執行。 啟用輸出緩衝: ob_start()

```
<?php
    Ob_start();
?>
```

二、接受及驗證數據

HTML 表單元素

表單元素	描述
Text input	文本框
Password input	密碼框
Hidden input	隱藏框

select	下拉式列表
checkbox	複選框
radio	單選按鈕
textarea	區域框
file	上傳
submit	提交按鈕
reset	重置按鈕

GET 與 POST

處理表單時，必須指定輸入到表單的信息以何種方式傳輸到其目的地 (method="")對此，Web 開發人員可以採用 GET 和 POST。使用 GET 方法發送數據時，所有域都追加到瀏覽器的 URL 後面，並且為數據隨 URL 地址發送。採用 POST 方法時，值會做為標準值發送。

PHP 分別使用\$_GET 和\$_POST 超全局變量來處理 GET 和 POST 變量。通過使用這兩個超全局變量，可以準確地指定信息應當來自哪裡，並以你希望的方式處理數據。

使用\$_GET 或\$_POST 來接收數據

1. \$_GET['username'], 發送的表單 method 必須是 get;
2. \$_POST['username'], 發送的表單 method 必須是 post;
3. 採用 isset()來驗證\$_GET['username']超級全局變量是否定義
4. 使用 htmlspecialchars()函數將 HTML 特殊字符進行過濾

對數據有效性進行驗證

1. 使用函數 trim()去除數據的前後空格
2. 使用函數 strlen()判斷數據長度
3. 使用函數 is_numeric()判斷數據是否為數字
4. 使用正則表達式驗證郵箱是否合法。