

PHP File Download

In this tutorial you will learn how to force download a file using PHP.

Downloading Files with PHP

Normally, you don't necessarily need to use any server side scripting language like PHP to download images, zip files, pdf documents, exe files, etc. If such kind of file is stored in a public accessible folder, you can just create a hyperlink pointing to that file, and whenever a user click on the link, browser will automatically downloads that file.

Example	Try this code »
<pre>Download Zip file Download PDF file Download Image file Download EXE file</pre>	

Clicking a link that points to a PDF or an Image file will not cause it to download to your hard drive directly. It will only open the file in your browser. Further you can save it to your hard drive. However, zip and exe files are downloaded automatically to the hard drive by default.

Forcing a Download Using PHP

You can force images or other kind of files to download directly to the user's hard drive using the PHP `readfile()` function. Here we're going to create a simple image gallery that allows users to download the image files from the browser with a single mouse click.

Let's create a file named "image-gallery.php" and place the following code inside it.

Example	Run this code »
<pre><!DOCTYPE html> <html lang="en"> <head> <meta charset="UTF-8"> <title>Simple Image Gallery</title> <style type="text/css"> .img-box{ display: inline-block; text-align: center;</pre>	

```

        margin: 0 15px;
    }
</style>
</head>
<body>
    <?php
    // Array containing sample image file names
    $images = array("kites.jpg", "balloons.jpg");

    // Loop through array to create image gallery
    foreach($images as $image){
        echo '<div class="img-box">';
        echo '';
        echo '<p><a href="download.php?file=' . urlencode($image) .
'">Download</a></p>';
        echo '</div>';
    }
    ?>
</body>
</html>

```

If you see the above example code carefully, you'll find the download link points to a "download.php" file, the URL also contains image file name as a query string. Also, we've used PHP `urlencode()` function to encode the image file names so that it can be safely passed as URL parameter, because file names may contain URL unsafe characters.

Here's the complete code of "download.php" file, which force image download.

Example

Run this code »

```

<?php
if(isset($_REQUEST["file"])){
    // Get parameters
    $file = urldecode($_REQUEST["file"]); // Decode URL-encoded string

    /* Test whether the file name contains illegal characters
    such as "../" using the regular expression */
    if(preg_match('/^[^.] [-a-z0-9_]+[a-z]$/i', $file)){
        $filepath = "images/" . $file;

        // Process download
        if(file_exists($filepath)) {
            header('Content-Description: File Transfer');

```

```
        header('Content-Type: application/octet-stream');
        header('Content-Disposition: attachment;
filename="'.basename($filepath).'"');
        header('Expires: 0');
        header('Cache-Control: must-revalidate');
        header('Pragma: public');
        header('Content-Length: ' . filesize($filepath));
        flush(); // Flush system output buffer
        readfile($filepath);
        die();
    } else {
        http_response_code(404);
        die();
    }
} else {
    die("Invalid file name!");
}
}
```

?>

Similarly, you can force download other files formats like word doc, pdf files, etc.

The regular expression in the above example (*line no-8*) will simply not allow those files whose name starts or ends with a dot character (.), for example, it allows the file names such as kites.jpg or Kites.jpg, myscript.min.js but do not allow kites.jpg. or .kites.jpg.

Please check out the tutorial on [regular expressions](#) to learn the regular expressions in details.