

# PHP - Download File

Most modern browsers allow files of certain types to be downloaded automatically, without any server-side code such as a PHP script. For example, a zip file, or an EXE file.

If an HTML hyperlink points to a ZIP or EXE file, the browser downloads it and pops up a save dialog. However, text files, image files, etc., are not downloaded but opened in the browser, which you can save to your local filesystem.

## The readfile() Function

To download such files (instead of the browser automatically opening them), we can use the **readfile()** function in PHP's built-in function library.

```
readfile(string $filename,  
bool $use_include_path = false,  
?resource $context = null)  
: int|false
```

This function reads a file and writes it to the output buffer.

The second parameter **\$use\_include\_path** is false by default, hence the file in the current directory will be downloaded. If set to **true**, the directories added to the **include\_path** setting of **php.ini** configuration will be searched to locate the file to be downloaded.

The readfile() function returns the number of bytes read or false even it is successfully completed or not.

## Example

The following PHP script shows the usage of readfile() function.

To download a file, the **Content-Type** response header should be set to **application/octet-stream**. This MIME type is the default for binary files. Browsers usually don't execute it, or even ask if it should be executed.

Additionally, setting the Content-Disposition header to attachment prompts the "Save As" dialog to pop up.

```
<?php  
$filePath = 'welcome.png';
```

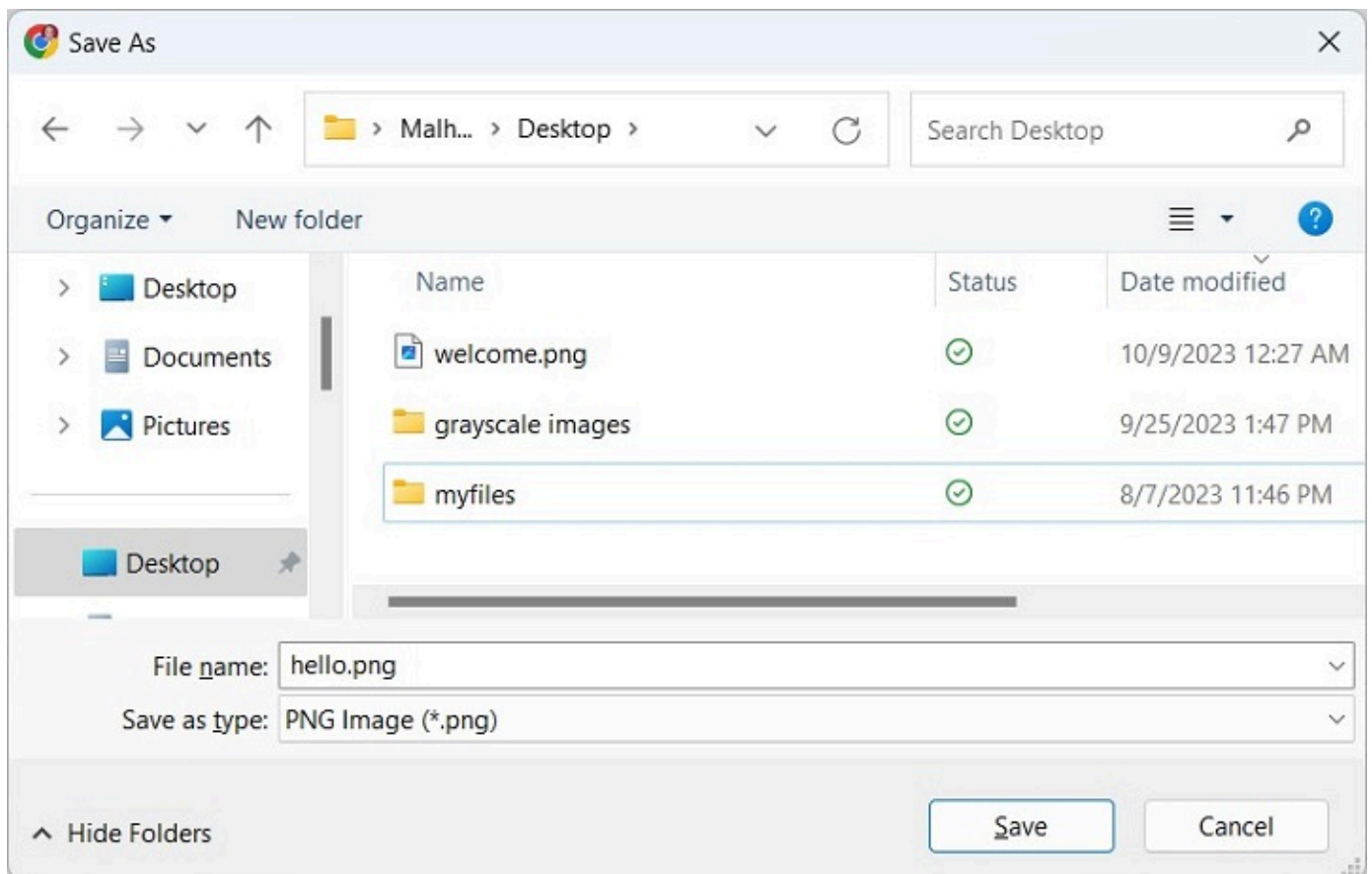
```
// Set the Content-Type header to application/octet-stream
header('Content-Type: application/octet-stream');

// Set the Content-Disposition header to the filename of the downloaded file
header('Content-Disposition: attachment; filename="'. basename($filePath) .'"');

// Read the contents of the file and output it to the browser.
readfile($filePath);
```

Save the above script as "download.php" in the document root folder. Make sure that the file to be downloaded is present in the same folder.

Start the server and visit **<http://localhost/download.php>** in the browser. You will get a "Save As" dialog as below –



You can select a name and download the file.

For a large file, you can read it from the file stream in the chunk of a certain predefined size. The browser offers to save it in the local filesystem, if the Content-Disposition head is set to "attachment", as in the previous example.

```
<?php
$filename = 'welcome.png';
```

```
header('Content-Type: application/octet-stream');
header('Content-Disposition: attachment; filename="' . basename($filename) . '"');

$handle = fopen($filename, 'rb');
$buffer = '';
$chunkSize = 1024 * 1024;

ob_start();
while (!feof($handle)) {
    $buffer = fread($handle, $chunkSize);
    echo $buffer;
    ob_flush();
    flush();
}
fclose($handle);
```