

File



- [Description](#)
 - [Screenshots](#)
 - [Changelog](#)
 - [Settings](#)
 - [Template usage](#)
 - [Basic display \(array\)](#)
 - [Advanced display \(array\)](#)
 - [Basic display \(ID\)](#)
 - [Basic display \(URL\)](#)
-

Description

The File field allows a file to be uploaded and selected by using the native WP media popup.

Screenshots

A file field that allows you to upload and select a file

The File field interface

The list of File field settings shown when setting up a File field

The File field settings

Changelog

- Added `minimum` setting in version 5.1.9.

- Added `maximum` setting in version 5.1.9.
- Added `allowed file types` setting in version 5.1.9.

Settings

- **Return value**
Specifies the format of the returned data. Choose from File Array (array), File URL (string), or File ID (integer).
- **Library**
Limits file selection to only those that have been uploaded to this post, or the entire library.
- **Minimum**
Adds upload validation for minimum filesize in MB (integer). The filesize may also be entered as a string containing the unit. eg. '400 KB'.
- **Maximum**
Adds upload validation for maximum filesize.
- **Allowed file types**
Adds upload validation for specific file types. Enter a comma separated list to specify which file types are allowed or leave blank for all types.

Template usage

The File field will return either an array, a string or an integer value depending on the *Return Value* set. Below are some examples of how you can use this data.

Basic display (array)

This example demonstrates how to display the selected file when using the `array` return type. This return type allows us to easily access data such as `url` and `filename`.

```
<?php
$file = get_field('file');
if( $file ): ?>
    <a href="<?php echo $file['url']; ?>"><?php echo $file['filename']; ?></a>
<?php endif; ?>
```

Advanced display (array)

This example demonstrates how to display a custom link when using the `array` return type. This return type allows us to easily access data such as `url`, `title`, `type` and more.

```
<?php
$file = get_field('file');
if( $file ):

    // Extract variables.
    $url = $file['url'];
    $title = $file['title'];
    $caption = $file['caption'];
    $icon = $file['icon'];

    // Display image thumbnail when possible.
    if( $file['type'] == 'image' ) {
        $icon = $file['sizes']['thumbnail'];
    }

    // Begin caption wrap.
    if( $caption ): ?>
        <div class="wp-caption">
<?php endif; ?>

    <a href="<?php echo esc_attr($url); ?>" title="<?php echo esc_attr($title); ?>">
        
        <span><?php echo esc_html($title); ?></span>
    </a>

<?php
// End caption wrap.
if( $caption ): ?>
    <p class="wp-caption-text"><?php echo esc_html($caption); ?></p>
    </div>

<?php endif; ?>
<?php endif; ?>
```

Basic display (ID)

This example demonstrates how to display the selected file when using the `ID` return type.

```
<?php
$file = get_field('file');
if( $file ):
    $url = wp_get_attachment_url( $file ); ?>
    <a href="<?php echo esc_html($url); ?>" >Download File</a>
<?php endif; ?>
```

Basic display (URL)

This example demonstrates how to display the selected file when using the `URL` return type.

```
<?php if( get_field('file') ): ?>
    <a href="<?php the_field('file'); ?>" >Download File</a>
<?php endif; ?>
```

```
### #1
[] [] [] 2026-08-29 15:44:03 UTC
[] [] [] 2026-08-29 15:44:03 UTC
```