

# Gravity Wiz

## SNIPPETS

# Customizing the Multi-File Merge Tag

JANUARY 18, 2014 | DAVID SMITH | LEAVE A COMMENT

A simple way to create custom templates for displaying files uploaded via a Multi-File Upload field. Image files can be displayed as images, video files can be loaded as playable videos, PDFs (and other text file types) can be stylized to indicate the file type. The possibilities are endless.

[Show Code](#) [Download Code](#)

Gravity Forms v1.8 adds support for the much desired multi-file uploads field. Just add a File Upload field to your form, check the “Enable Multi-File Upload” setting on the field, and just like that, you’ll find yourself in multi-file upload heaven.

Like every other Gravity Forms field, the Multi-File Upload field has a merge tag that can be used in Notifications, Confirmations, [Post Content Templates](#), and more. It looks something like `{Files:2}` (where “Files” is the field label and “2” is the ID of the field).

By default, this merge tag will output a simple list of the file URLs that were uploaded.

```
http://myurl.com/.../Tulips8.jpg  
http://myurl.com/.../Penguins11.jpg  
http://myurl.com/.../Lighthouse11.jpg
```

This isn’t a very pretty (or useful) way to display these files for the end user.

This snippet provides a simple way to provide custom markup (HTML) templates for displaying these file URLs. Image URLs can be displayed as images, video URLs can be loaded as playable videos, PDFs (and other text file types) can be stylized more attractively to indicate the file type, and so on. The possibilities are endless.

## How do I get started?

1. Make sure you're running Gravity Forms v1.8 or later.
  - Already have a license? [Download Latest Gravity Forms](#)
  - Need a license? [Buy Gravity Forms](#)
2. Copy and paste the snippet into your theme's *functions.php* file.
3. Review the usage instructions below to configure the snippet for your needs.

## Basic Usage

Applies **default markup** to all forms.

```
gw_multi_file_merge_tag()->register_settings();
```

## Exclude Form(s)

Applies **default markup** to all forms excluding the specified forms.

```
gw_multi_file_merge_tag()->register_settings( array(
    'exclude_forms' => 2 // multiple forms: array( 2, 4 )
) );
```

## Specific Form

Applies **default markup** to a specific form.

```
gw_multi_file_merge_tag()->register_settings( array(
    'form_id' => 402
) );
```

## Specific Form w/ Custom Markup

Applies custom markup to a specific form. If the snippet has also been configured to apply the default markup to all forms, the markup provided here will override the default markup for this form.

```
gw_multi_file_merge_tag()->register_settings( array(
    'form_id' => 402,
    'markup' => array(
        array(
```

```

        'file_types' => array( 'jpg', 'jpeg', 'png', 'gif' ),
        'markup' => '<div class="gw-image"><a href="{url}" class="gw-image-link"><img
    ),
    array(
        'file_types' => array( 'mp4', 'ogg', 'webm' ),
        'markup' => '<video width="320" height="240" controls>
                    <source src="{url}" type="video/{ext}">
                    Your browser does not support the video tag.
                    </video>'
    )
)
);

```

## Parameters

- **form\_id** (int) The ID of the form to which this snippet should be applied. Defaults to **false**. If you want to apply this snippet to all forms, do not pass this parameter.
- **exclude\_forms** (int|array) A single form ID or array of form IDs that should be excluded from this snippet. Defaults to **array()**.
- **markup** (array) An array of arrays, with each child array having a 'file\_types' and 'markup' property. See the [Default Markup](#) section for default values.
  - *file\_types* (array) An array of file types for which this markup should apply.
  - *markup* (string) An HTML-based template for how the file type should be displayed. See [Markup Merge Tags](#) for a list of available merge tags that can be used in this template.

## Default Markup

Here is a list of the default markup that comes packaged with the snippet. Default support is limited to images ("jpg", "jpeg", "png" and "gif") and videos ("mp4", "ogg", "ogv", "webm"). If you'd think there are more file types that should be supported, leave a comment with the file type and suggested markup.

```

'markup' => array(
    array(
        'file_types' => array( 'jpg', 'png', 'gif' ),
        'markup' => ''
    ),
    array(
        'file_types' => array( 'mp4', 'ogg', 'webm' ),
        'markup' => '<video width="320" height="240" controls>
                    <source src="{url}" type="video/{ext}">
                    Your browser does not support the video tag.
                    </video>'
    )
),

```

```
array(  
    'file_types' => array( 'ogv' ),  
    'markup' => '<video width="320" height="240" controls>  
                <source src="{url}" type="video/ogg">  
                Your browser does not support the video tag.  
            </video>'  
    )  
)
```

## Markup Merge Tags

The `markup` parameter supports the usage of the following merge tags which provide access to information about the uploaded file. See the [Default Markup](#) section and [custom markup](#) usage example for examples of usage.

```
http://myurl.com/.../Tulips8.jpg
```

- `{url}` The URL of the file (i.e. “http://myurl.com/.../Tulips8.jpg”).
- `{filename}` The name of the file (i.e. “Tulips8”).
- `{basename}` The name of the file including the extension (i.e. “Tuplips8.jpg”).
- `{ext}` The extension of the file (i.e. “jpg”).

## Summary

If there are more file types that should be supported, leave a comment with the file type and suggested markup. And if you use it and like it, let me know!