

Media field with library



This section shows how to add a media field to your entity, using the media library, and to only show media items you have uploaded.



Enable the core **media** module as well as the core **media library** module:

```
bash$ drush en media media_library
The following module(s) will be enabled: media, media_library,
image

Do you want to continue? (yes/no) [yes]:
> yes

[success] Successfully enabled: media, media_library, image
```

A real advantage of drupal is that time-consuming development practices like file uploads and managing a file library are easily installed with some clicks.



Define a media type at [Structure > Media types > Add media Type](#) and choose this:

Name: Image

Media source: Image

After saving the media type, you'll discover 'Image' is a bundle of entity media. It has fields which are preconfigured at [admin/structure/media/manage/image/fields](#). In this case, the entity has an image field that is shipped with the media source: image.

Home > Administration > Structure > Media types > Edit Image

← 🔑 👤

Manage fields

+ Add field

Edit Manage fields Manage form display Manage display Devel

Label	Machine name	Field type	Operations
Image	field_media_image	Image	<div>Edit ▼</div>

We tweak the image field at /admin/structure/media/manage/image/fields/media.image.field_media_image with some settings, so resizing is automatically done when a file is too large.

Image settings for *Image*

Allowed file extensions *

png, gif, jpg, jpeg

Separate extensions with a space or comma and do not include the leading dot.

File directory

media/[media:uid]

Optional subdirectory within the upload destination where files will be stored. Do not include preceding or trailing slashes.

Maximum image resolution

2000

×

3000

pixels

The maximum allowed image size expressed as WIDTH×HEIGHT (e.g. 640×480). Leave blank for no restriction. If a larger image is uploaded, it will be resized to reflect the given width and height. Resizing images on upload will cause the loss of [EXIF data](#) in the image.

Minimum image resolution

×

pixels

The minimum allowed image size expressed as WIDTH×HEIGHT (e.g. 640×480). Leave blank for no restriction. If a smaller image is uploaded, it will be rejected.

Maximum upload size

5 MB

Enter a value like "512" (bytes), "80 KB" (kilobytes) or "50 MB" (megabytes) in order to restrict the allowed file size. If left empty the file sizes will be limited only by PHP's maximum post and file upload sizes (current limit **64 MB**).

☒ Enable Alt field

Short description of the image used by screen readers and displayed when the image is not loaded. Enabling this field is recommended.



We'll add a media field for our offer at </admin/structure/offer/fields> > *add field*

Add a new field (Media)

Add a label (Image)

Help text (Market your offer with a tremendous image!)

Reference type (Default)

Media Type (Image)

(save and continue)

Allowed number of values (1)

Required field (yes)

An additional step is giving our authenticated users the permissions to add these media items. Because the core media module works like our offer entity, it has its own permissions. Set them to make sure users can create and edit media entities:

Permissions		
Permission	Anonymous user	Authenticated user
Create media	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Delete any media	<input type="checkbox"/>	<input type="checkbox"/>
Delete own media	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Update any media	<input type="checkbox"/>	<input type="checkbox"/>
Update own media	<input type="checkbox"/>	<input checked="" type="checkbox"/>
View all media revisions <small>To view a revision, you also need permission to view the media item.</small>	<input type="checkbox"/>	<input type="checkbox"/>
View media	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
View own unpublished media	<input type="checkbox"/>	<input checked="" type="checkbox"/>

Save and head back to our offer create form (</offers/create>) we now have a widget with a media library that works for us out-of-the box!

Add offer

Image

No media items are selected.

Add media

Market your offer with a tremendous image!
One media item remaining.

Save



You'll notice that when adding media items to your library, also media items of others are visible. This is because the media library is using a view to display the content. At /admin/structure/views/view/media_library **add the same contextual filter as we did with the offers view**: contextual filters > media:authored by > provide default value > User ID from logged in user.

Important! Do this for every display mode:

/admin/structure/views/view/media_library/edit/page

/admin/structure/views/view/media_library/edit/widget

/admin/structure/views/view/media_library/edit/widget_table

This will *only* make sure we don't see media items from other users inside our library.

Media entities are not secured further in this course. If you would like more security, you could overwrite the Media module `AccessControlHandler()` by adding a `offer_entity_access()` hook to your **modules/custom/offer/offer.module**. You could also use private files for uploads.

But we'll leave this public, as this is not a feature that is needed.

