

Image Constructor For Custom Options for Magento 2

Installation	1
Usage	2
Import & Export	5
Import	5
Export	6
Known issues	7
No layers in custom theme with image hover zoom	7
Bug Reporting/Feature Request	7

Installation and User Guide

Installation

1. First of all, back up your store database and web directory. It is wise to perform this procedure before any installation.
2. Copy extension zip archive to the **magento root directory**.
3. **The next steps should be done using SSH console.** Connect to your hosting server using SSH client and go to the **magento root directory**.

```
cd ~/public_html
```

The "~/public_html" path is an example of **magento root directory**, which can be different on your hosting server.

4. Create the **app/code/Orange35/ImageConstructor** folder and extract archive files there:

```
mkdir -p app/code/Orange35/ImageConstructor  
unzip orange35-image-constructor.zip -d app/code/Orange35/ImageConstructor
```

5. **Install depended modules:**

```
composer require orange35/magento2-catalog-import-export  
composer require orange35/magento2-core
```

6. Enable the modules:

```
php bin/magento module:enable Orange35_CatalogImportExport
php bin/magento module:enable Orange35_Core
php bin/magento module:enable Orange35_ImageConstructor
```

7. Run the following commands:

```
php bin/magento setup:upgrade
php bin/magento setup:di:compile
```

8. If your store is in production mode then deploy static content

```
php bin/magento setup:static-content:deploy
```

9. Congratulations! You are now ready to use Image Constructor For Custom Options extension.

Usage

1. Go to the Edit Product page on the back end
2. Add an Image in the **Images And Videos** accordion tab
3. Add a custom option with Option Type = Drop-down.
4. Add multiple values to the custom option. You should see the **Image Layer** column.
5. Upload an image in the **Image Layer** column for each custom option value.

The screenshot shows the 'Edit Product' page for a product named 'Tank Top'. At the top, there are fields for 'Option Title' (Tank Top), 'Option Type' (Drop-down), and a 'Required' checkbox (checked). Below this is a table with columns: Title, Price, Price Type, SKU, and Image Layer.

Title *	Price *	Price Type	SKU	Image Layer
Blue	\$ 1.0000	Fixed		Upload
Green	\$ 2.0000	Fixed		 /t/a/tank-top-green.png 373x560
Maroon	\$ 3.0000	Fixed		

Image 1. Edit Product Page

6. Save the product and find it on the front end.
7. You should see the main image.



Image Constructor Test #1

[Be the first to review this product](#)

\$50.00

IN STOCK
SKU#: RTP

Tank Top *

-- Please Select --

Skirt *

-- Please Select --

Qty

1

Add to Cart

[ADD TO WISH LIST](#) [ADD TO COMPARE](#) [EMAIL](#)

Image 2. Product Detail Page. Main image.

8. Select the option value which has an Image Layer and you will see that the layer applied to the main image.



Image Constructor Test #1

[Be the first to review this product](#)

\$52.00

IN STOCK
SKU#: RTP

Tank Top *

Green +\$2.00

Skirt *

-- Please Select --

Qty

1

Add to Cart

[ADD TO WISH LIST](#) [ADD TO COMPARE](#) [EMAIL](#)

Image 3. Product Detail Page. Main Image with Layer applied.



RIGHT SOLUTIONS FOR YOUR BUSINESS

Website: <http://orange35.com>

Email: info@orange35.com

Import & Export

Module supports “Entity Type = Products” import & export.

Import

1. Create the following **import directory** and upload layer images there
~/public_html/pub/media/import/catalog/orange35/image-constructor/
where ~/public_html is a project root directory which may be different in your case.
For example you have the following files there:
 - skirt-light-blue.png
 - abc/skirt-light-green.png - where abc is a subfolder
2. Go to the **Magento Admin > System > Data Transfer > Import** section
3. Choose **Entity Type = Products**
4. Prepare a CSV file for import. You can download a sample file and use it. Keep in mind that a **custom_options** column may have multiple rows split by “[” separator. For each row you can add a **layer** field. Image name/path must be relative to the **import directory** (see point #1 above). E.g.:
 - layer=skirt-light-blue.png
 - layer=abc/skirt-light-green.png
5. When CSV file is ready, choose it in the **Select File to Import** field and press a **Check Data** button at the top right corner of the page.
6. If the file is valid, start import by pressing an **Import** button at the bottom of the page.

As a result, the layer images will be imported along with the products.

If a wrong layer file name is specified in a CSV file, then the product will be imported but without the layer. Also, there will be a warning in the **var/log/system.log** file, e.g.

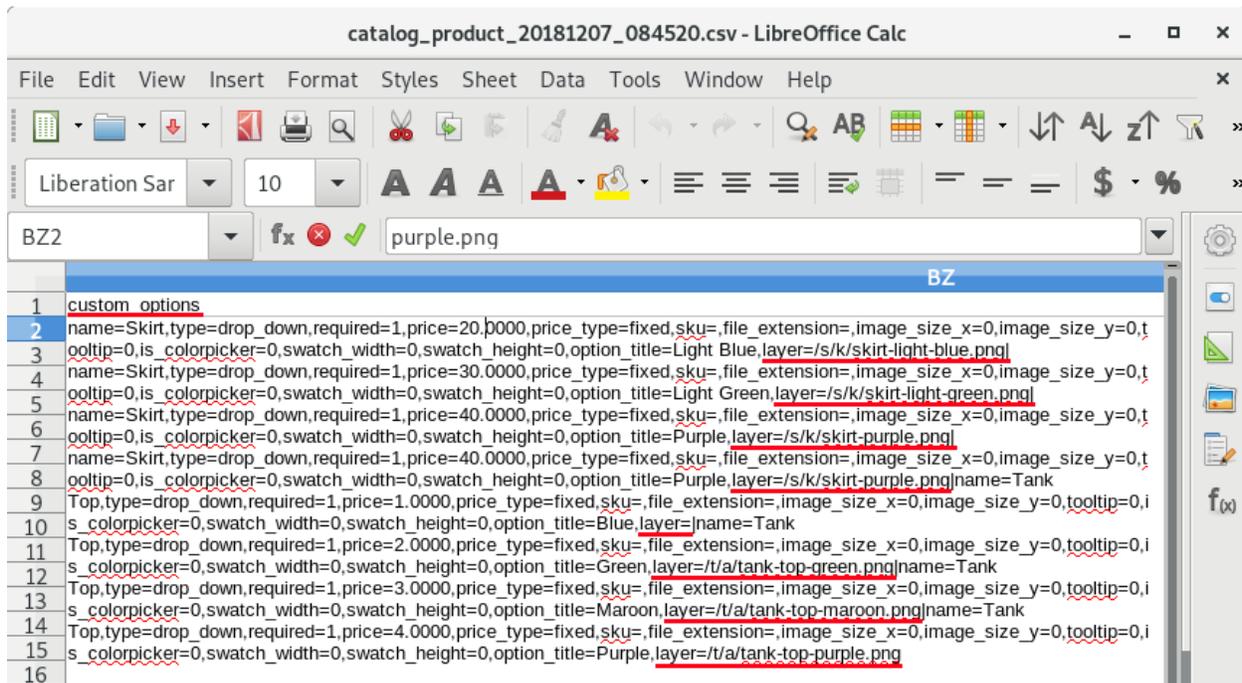
```
[2018-10-19 11:59:29] main.WARNING: Import [SKU=RTP-6;OPTION=Skirt;VALUE=Light Blue]. The file "/home/user/public_html/pub/media/import/catalog/orange35/image-constructor/skirt-light-blue-123.png" doesn't exist or not a file
```

The **import directory** (see point #1 above) can be removed to save disk space since images were copied to another folders during import.

Export

1. Go to the **Magento Admin > System > Data Transfer > Export** section
2. Choose **Entity Type = Products**
3. Fill in additional product filters if needed
4. Press a **Continue** button at the bottom of the page

As a result, a csv file will be downloaded with a **layer** field in **custom_options** column. See the screenshot below.



Known issues

No layers in custom theme with image hover zoom

Custom theme with the mouse hover zoom on image may conflict with the Image Constructor. As a result the layers may not be visible on the main product image. To resolve this issue follow the following steps:

1. Please make sure the **imagick** php extension is installed on your hosting. Log in via SSH to your hosting server and run the following command. If it outputs “imagick” in the next line then imagick php extension is installed and you can go to the next step. Otherwise contact your hosting provider and ask to install “imagick php extension”.

```
$ php -m | grep imagick  
imagick
```

2. Setup Magento to use the Image Magick adapter instead of GD2

```
php bin/magento config:set dev/image/default_adapter IMAGEMAGICK
```

3. Enable the **Single Image Strategy** (merge image and layers to a single image)

```
php bin/magento config:set orange35_image_constructor/general/single_image_strategy 1
```

4. Clear Magento cache and check the results on the frontend

```
php bin/magento cache:clean config
```

Bug Reporting/Feature Request

Would like to report a bug, request a new feature or simply ask a question? Feel free to email us at info@orange35.com and we'll gladly assist you.