Krasbit

Layouter^2

Examples

Table of contents

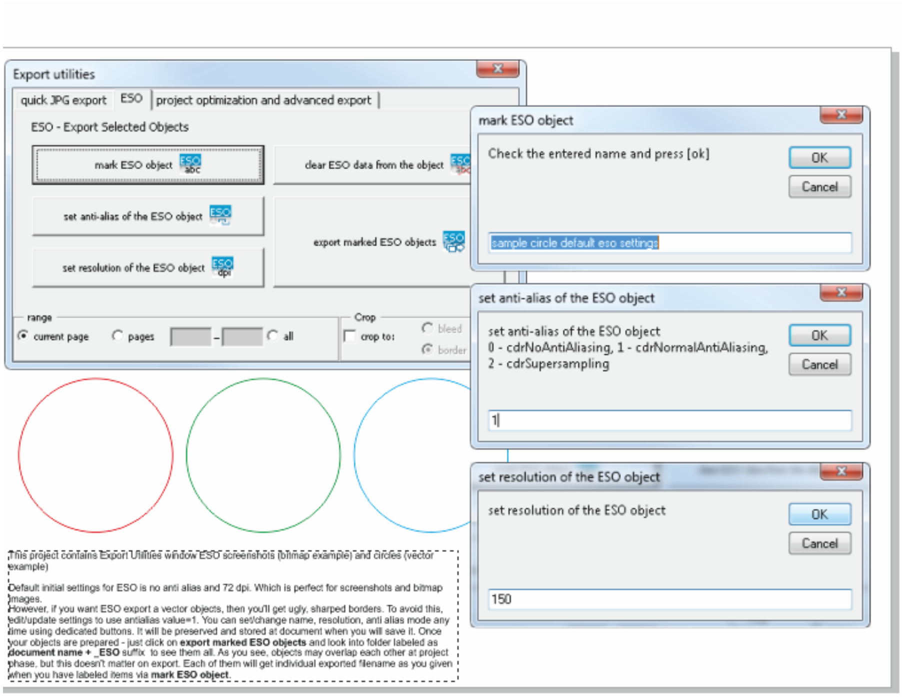
ESO

Export Slice Objects.cdr

This project contains Export Utilities window ESO screenshots (bitmap example) and circles (vector example).

Default initial settings for ESO is no anti alias and 72 dpi. Which is perfect for screenshots and bitmap images.
However, if you want ESO export a vector objects, then you'll get ugly, sharped borders. To avoid this, edit/update settings to use antialias value=1.

You can set/change name, resolution, anti alias mode any time using dedicated buttons. It will be preserved and stored at document when you will save it. Once your objects are prepared - just click on **export marked ESO objects** and look into folder labeled as **document name** + **_ESO** suffix. In case of example it will be *Export Slice Objects.cdr_ESO* folder. As you see, this folder will have individually exported file for each of marked ESO objects.
In contrast to traditional *"cut project by slices"* technique, when using ESO - objects may overlap each other inside project, but this doesn't matter on export. Each of them will get individual exported filename as you given when you have labeled items via **mark ESO object**.
If you do not want specific object to continue exporting as ESO object - select it and click **clear ESO data from object**.


Export utilities
quick JPG export
ESO
project optimization and advanced export
ESO - Export Selected Objects
mark ESO object
clear ESO data from the object
set anti-alias of the ESO object
export marked ESO objects
set resolution of the ESO object
range
● current page
○ pages
○ all
Crop
□ crop to:
○ bleed
● border
mark ESO object
OK
Cancel
set anti-alias of the ESO object
OK
Cancel
set resolution of the ESO object
OK
Cancel

Check the entered name and press [ok]

sample circle default eso settings

set anti-alias of the ESO object
0 - cdrNoAntiAliasing, 1 - cdrNormalAntiAliasing,
2 - cdrSupersampling

1

set resolution of the ESO object

150

This project contains Export Utilities window ESO screenshots (bitmap example) and circles (vector example)

Default initial settings for ESO is no anti alias and 72 dpi. Which is perfect for screenshots and bitmap images.
However, if you want ESO export a vector objects, then you'll get ugly, sharped borders. To avoid this, edit/update settings to use antialias value=1. You can set/change name, resolution, anti alias mode any time using dedicated buttons. It will be preserved and stored at document when you will save it. Once your objects are prepared - just click on **export marked ESO objects** and look into folder labeled as **document name + _ESO** suffix to see them all. As you see, objects may overlap each other at project phase, but this doesn't matter on export. Each of them will get individual exported filename as you given when you have labeled items via **mark ESO object**.

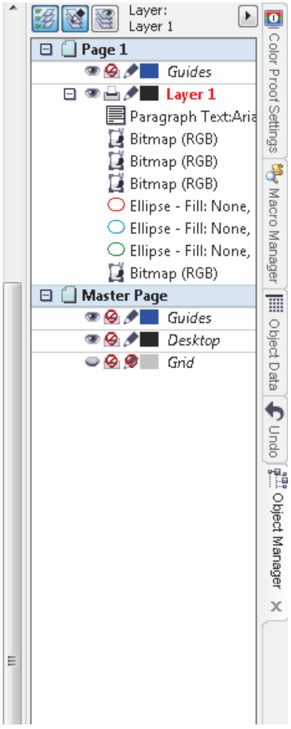
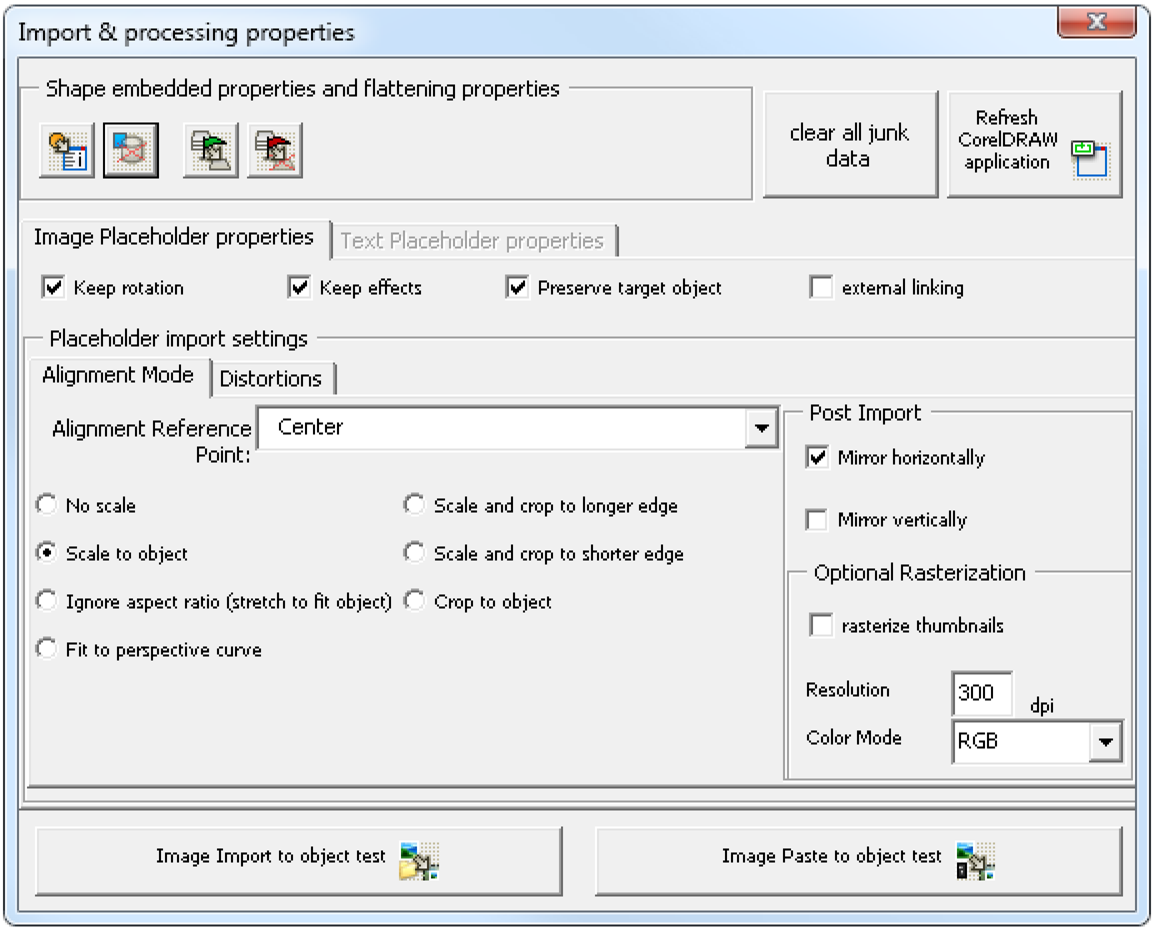


Image placeholder properties


Import & processing properties
Shape embedded properties and flattening properties
clear all junk data
Refresh CorelDRAW application
Image Placeholder properties
Text Placeholder properties
Placeholder import settings
Alignment Mode
Distortions
Image Import to object test
Image Paste to object test

☑ Keep rotation ☑ Keep effects ☑ Preserve target object □ external linking

Alignment Reference Point: Center

○ No scale
● Scale to object
○ Ignore aspect ratio (stretch to fit object)
○ Fit to perspective curve
○ Scale and crop to longer edge
○ Scale and crop to shorter edge
○ Crop to object

Post Import

☑ Mirror horizontally
□ Mirror vertically

Optional Rasterization

□ rasterize thumbnails

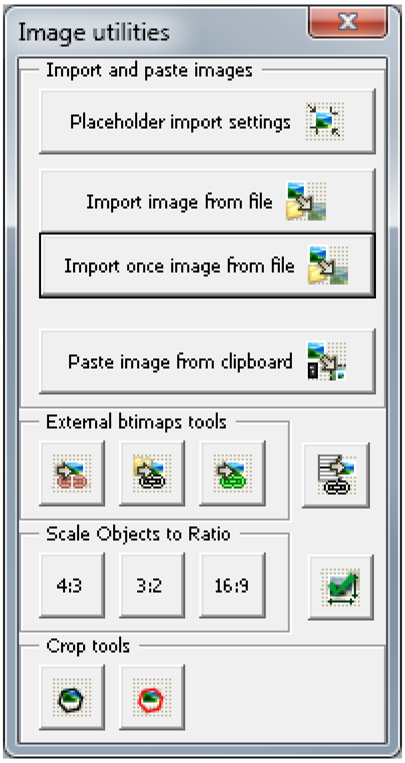
Resolution 300 dpi

Color Mode RGB

You should already read **Krasbit Layouter 2 User's Manual** at **Placeholder import settings** and **Image Utilities** chapter to find out details about all available options and buttons at this windows.

Placeholders may have or don't an outline. It's up to your design whatever you want to have a visible outline boundary over the placeholder object (area where images will be placed) or not. Enter into View - Wireframe mode to see all the placeholders. It's also up to you design whatever you would to keep the original placeholder object after image is placed (**Preserve target object option**) If you expect frequent updates of imported images in same template, you should consider to check this option. That options matters mostly for scale-like alignment modes, because scrop-like modes must persist original placeholder as it is used as it will be used as powerclip parent after placement (preserve target option will make a copy of placeholder frame in that case).

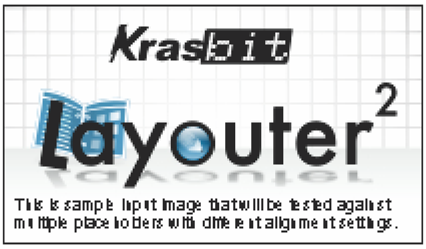
To configure placeholders open **Placeholder import settings** window and edit/update individually on each shape.



Alignment mode scale.cdr

This is an introductory example file with examples how different **Alignment Reference point** and **Alignment Mode** scales placed object to placeholder due to given settings. This project focus on scale **-** like modes. This is one of: **No scale**, **Scale to object**, **Ignore aspect ratio** (stretch to fit object)

Example steps:

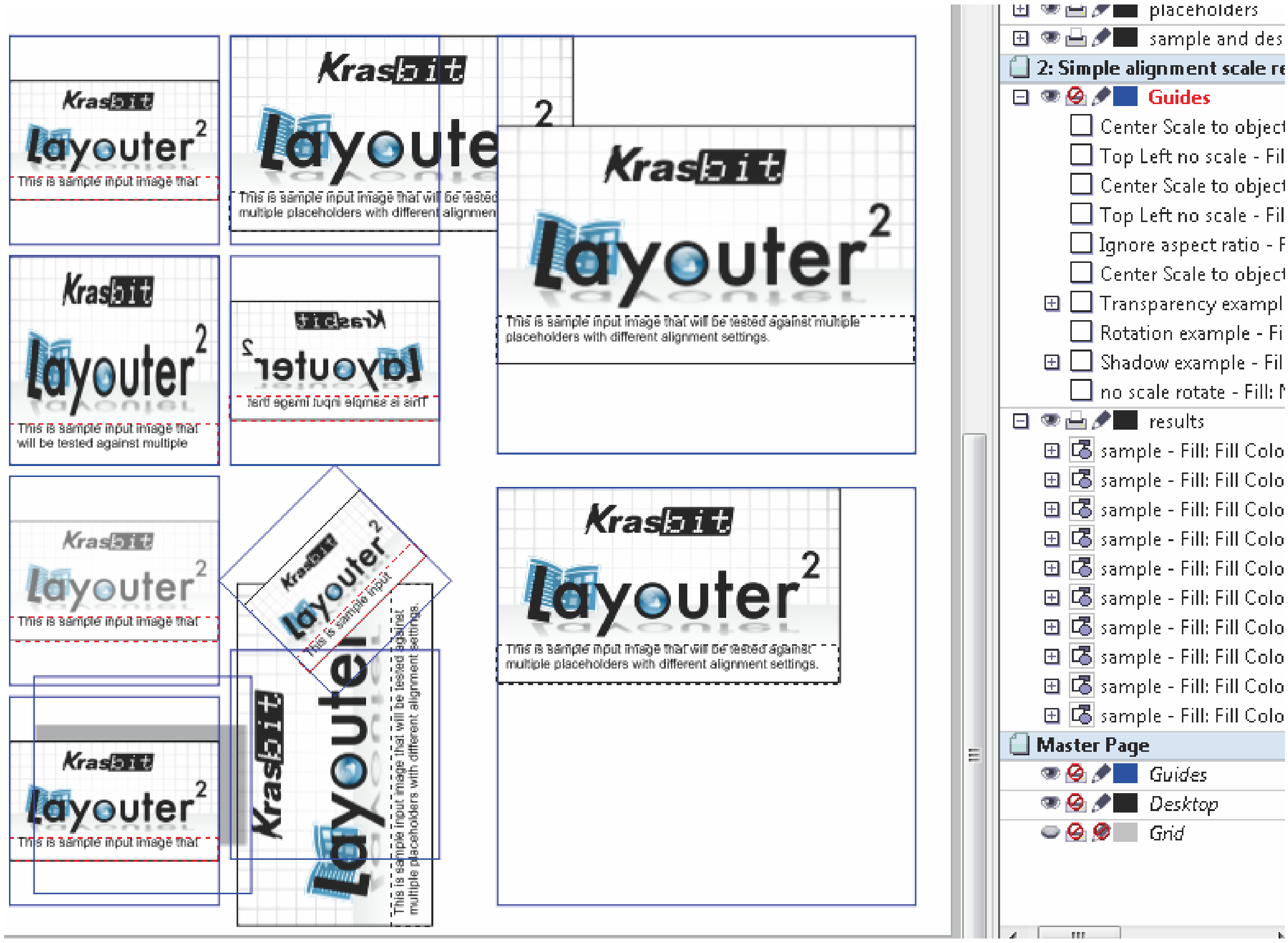

Krasbit
Layouter 2
This is sample input image that will be tested against multiple placeholders with different alignment settings.

1. Copy **sample** object to clipboard
2. Next, select all the objects on **placeholders** layer, then depends if you are working with
    - ○ **Image Utilities** window
        - ■ click **paste image from clipboard**
    - ○ **Import and image processing properties** window
        - ■ Click **Image Paste to object test**

Alternatively, you may try the image **importing** to placeholders.

1. Select all the objects on **placeholders** layer then depends if you are working with
    - ○ **Image Utilities** window
        - ■ click **Import image from file**
    - ○ **Import and image processing properties** window
        - ■ Click **Image Import to object test**
2. For each selected placeholder, application will prompt you to file browse to file you would to place inside placeholder. You may select *KrasbitLayouter2 sample logo.cdr* that exists in this folder or any other image that you would to test.

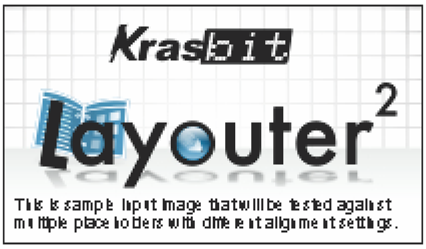
Simple alignment scale results



Alignment mode crop.cdr

This is an introductory example file with examples how different **Alignment Reference point** and **Alignment Mode** crops placed object to placeholder due to given settings. This project focus on crop - like modes. This is one of: **Scale and crop to longer edge**, **Scale and crop to shorter edge**, **Crop to object**.

Example steps:


Krasbit
layouter 2
This is sample input image that will be tested against multiple placeholders with different alignment settings.

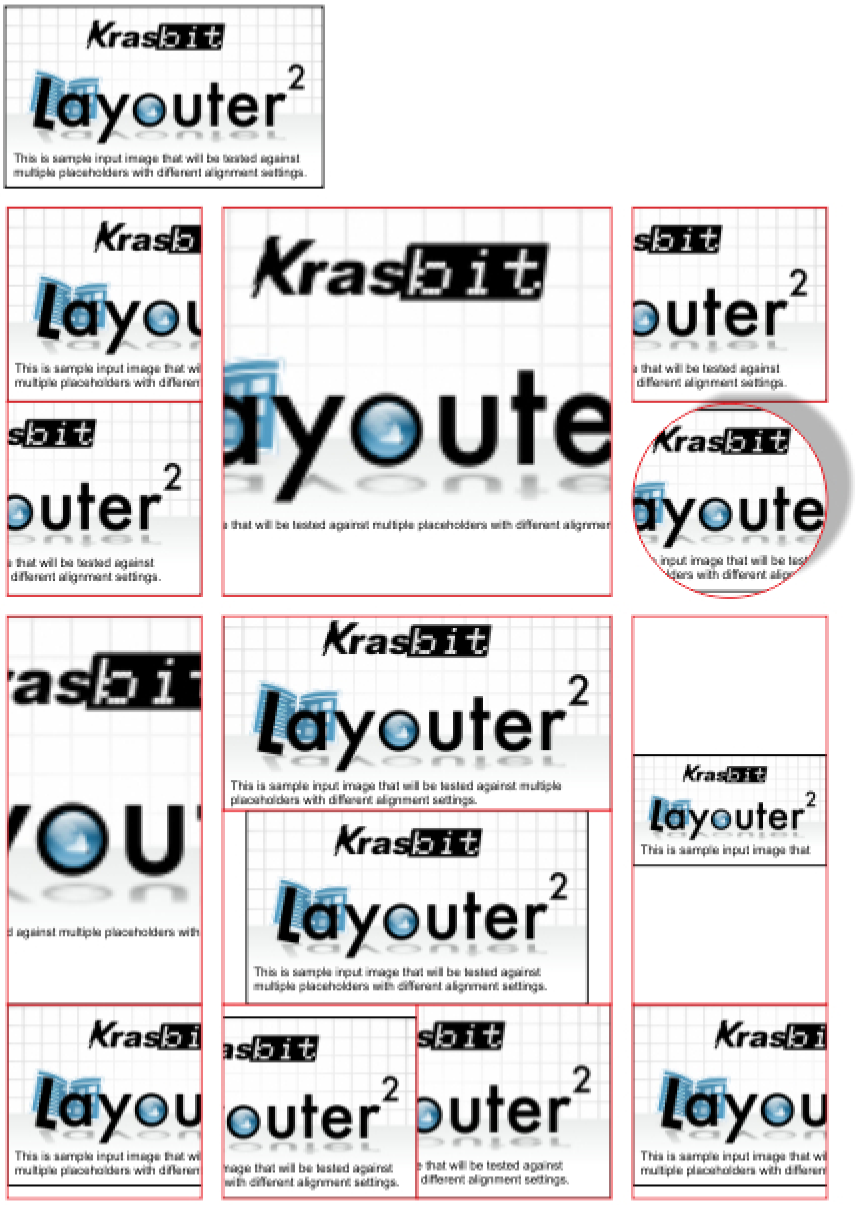
3. Copy **sample** object to clipboard
4. Next, select all the objects on **placeholders** layer, then depends if you are working with
    - **Image Utilities** window
        - click **paste image from clipboard**
    - **Import and image processing properties** window
        - Click **Image Paste to object test**

Alternatively, you may try the image **importing** to placeholders.

3. Select all the objects on **placeholders** layer then depends if you are working with
    - **Image Utilities** window

        - click **Import image from file**
    - **Import and image processing properties** window
        - Click **Image Import to object test**
4. For each selected placeholder, application will prompt you to file browse to file you would to place inside placeholder. You may select *KrasbitLayouter2 sample logo.cdr* that exists in this folder or any other image that you would to test.

Crop results


Krasbit
layouter 2
This is sample input image that will be tested against multiple placeholders with different alignment settings.

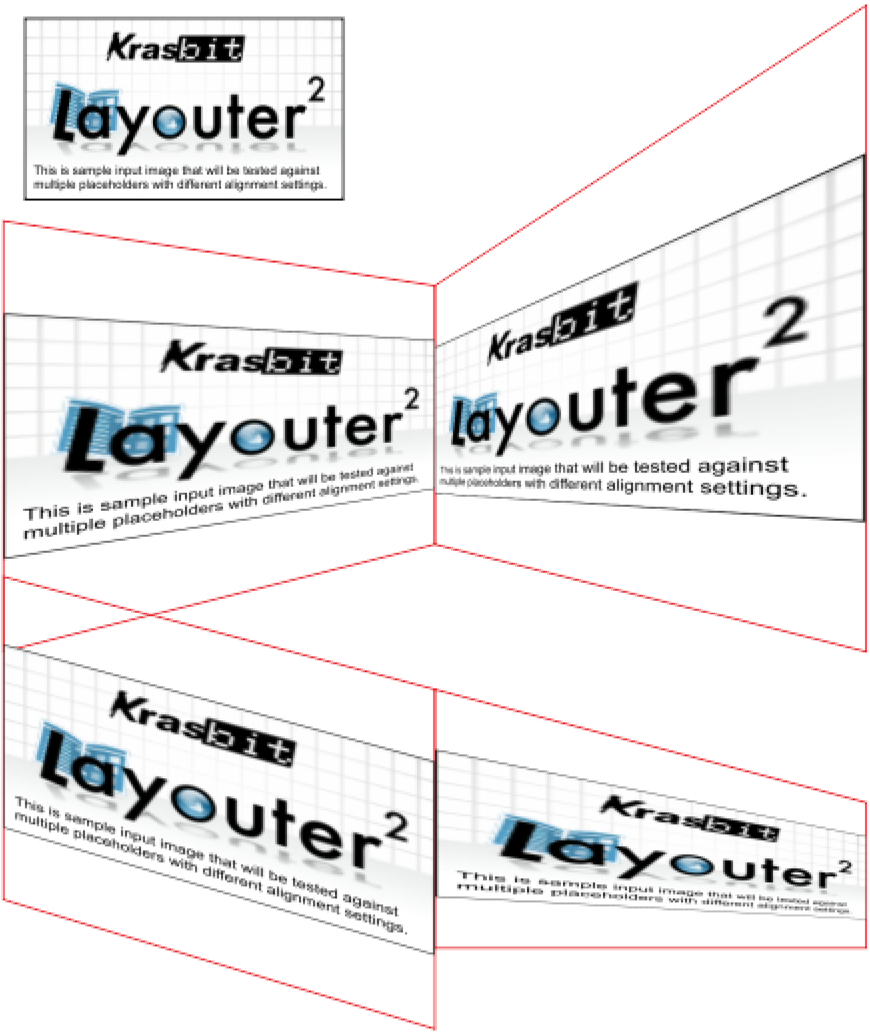
Alignment mode perspective.cdr

If you want to get a perspective effect when image will be placed.

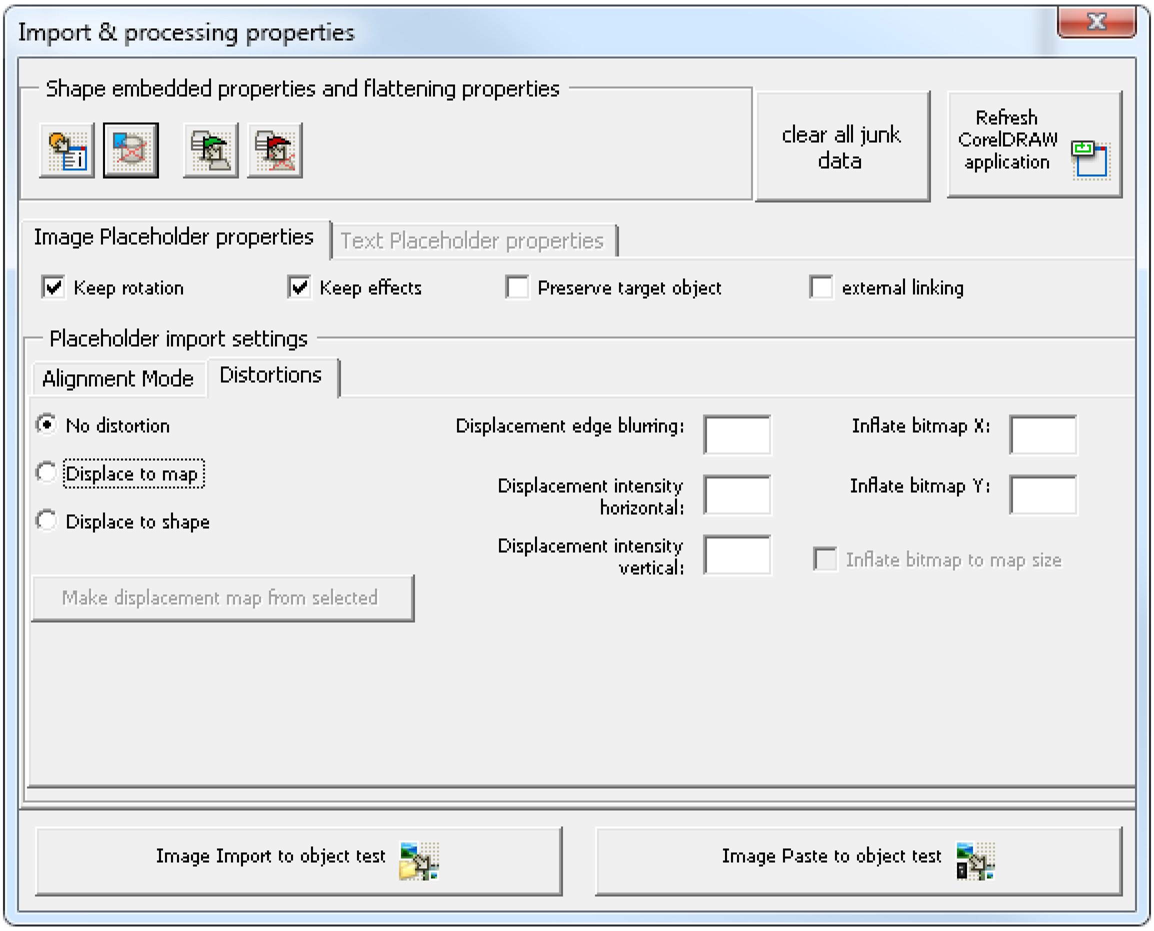
Steps

1. Start with simple rectangle shape
2. Convert rectangle to curve shape (CTRL +Q)
3. Use shape tool (F10) to drag and move any shape curve's corner to a position that will form your desired perspective
4. Select **Fit to perspective curve** as alignment mode
5. Specify desired rasterization settings. When using perspective alignment mode, rasterization will always occur.
6. Your placeholder in template is ready to place image and give it a perspective look. In contrast to CorelDRAW's native perspective effect that works only with vector shapes, **Krasbit Layouter perspective effect will work both with vector and bitmap shapes**.

Results


Krasbit
layouter 2
This is sample input image that will be tested against multiple placeholders with different alignment settings.

Displacement.cdr



Import & processing properties

Shape embedded properties and flattening properties

Image Placeholder properties | Text Placeholder properties

☑ Keep rotation ☑ Keep effects □ Preserve target object □ external linking

Placeholder import settings

Alignment Mode | Distortions

● No distortion

○ Displace to map

○ Displace to shape

Displacement edge blurring:

Displacement intensity horizontal:

Displacement intensity vertical:

Inflate bitmap X:

Inflate bitmap Y:

□ Inflate bitmap to map size

clear all junk data
Refresh CorelDRAW application
Make displacement map from selected
Image Import to object test
Image Paste to object test

Please read theory behind displacement at Krasbit Layouter 2- User’s Manual. Basically there are 2 kinds of displacement maps: grayscale and colorful. Both of them are based on pixel movement due to lightness or color. Additionally, with proper use of displacement settings you may control the factor and offset in desired X / Y axis. The rest is on example images. Open project in CorelDRAW and look into displacement settings of each placeholder. Play with them and have fun.

No distortion
Default option. Disables displace effect. No distortion made to target import image placeholder

Displace to map
This option will displace imported image in second step after it is scaled or cropped due to alignment mode. Placeholder must be a color or grayscale bitmap that defines how to push pixels in custom direction. Use this option if you are familiar with displacement theory and can prepare a map in advance which will do desired effect perfectly.

Displace to shape
This mode will make displacement map automatically and on the fly for a given input vector shape like circle, ellipse, curve. Image is imported into rendered displacement map in second step. Displacement map IN will be based on colorful displace map covered with neutral gray mask made from the shape with custom amount of blur. Finally pixels at non - masked areas should be pushed into the center and overall picture get the look like shape's mask.

Make displacement map
This button will generate a displacement map from selected vector shape in advance so it can be adjusted or reused somewhere else in project.

Displacement suboptions

Displacement edge blurring
This suboption available only in Displace to shape mode. Specifies how many of blur to apply when neutral gray mask is applied on shape.

Displacement intensity horizontal
This is multiplier/ factor Controls how much intensive will be the displace effect in X axis.
Specify 0 to ignore pushing pixels in that axis regardless of displace map pixel color

Displacement intensity vertical
This is multiplier/ factor Controls how much intensive will be the displace effect in Y axis.
Specify 0 to ignore pushing pixels in that axis regardless of displace map pixel color.

Inflate bitmap X
A number of transparent pixels to add to edges on x axis before applying displace effect.

Inflate bitmap Y
A number of transparent pixels to add to edges on y axis before applying displace effect.

Inflate matters in case when displace effect is pushing pixels out of original image boundary.
Without inflating bitmaps- final effect makes impression like image is cut. If inflate is used - pixels moved outside image are still visible on added space.

Inflate bitmap to map size
This checkbox toggles inflating as described above.

Displacement theory
There are **2** types of displacement maps that differs on how they are pushing pixels:

Grayscale map
Works if your input displacement map is given as grayscale image. Here are pixel offset directions
that this map does:
50% black is neutral for pixel. Makes no movement to a pixel. Use this color like a mask. Region covered with neutral 50% gray will not change.

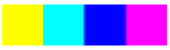
← ↑

White - moves pixel horizontally to the left and vertically upside. Pure white makes the maximum offset. That offset reduces smoothly as you use shades of gray between 0% of black up to neutral 50% gray.

→ ↓

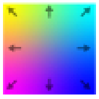
Black - moves pixel horizontally to the right and vertically downside. Pure black makes the maximum offset. That offset reduces smoothly as you use shades of gray between 100% of black down to neutral 50% gray

Color map
Works if your input displacement map is given as RGB image. It gives you more control for each
possible axis. Instead of black and white, you have this colors to control offset:

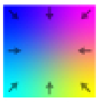


Yellow, Cyan, Blue and Magenta.
The more intense is the color the higher distance pixel is moved. Gray color makes no movement.

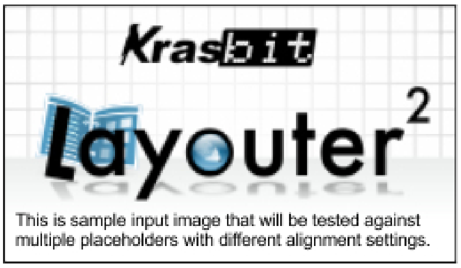
Pixel movement directions for certain colors works as pictures shows:



Displace map OUT.
* this displace map will be pushing pixels out of image bounds.
Created using Mesh Fill tool.



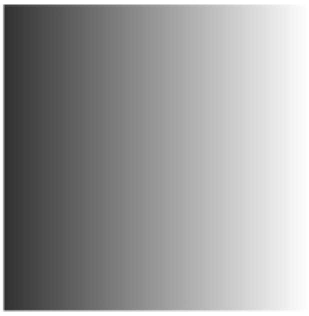
Displace map IN.
* this displace map will be pushing pixels into the center of image.
Created using Mesh Fill tool

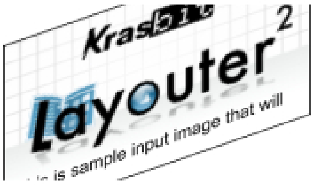

Krasbit
layouter 2
This is sample input image that will be tested against multiple placeholders with different alignment settings.

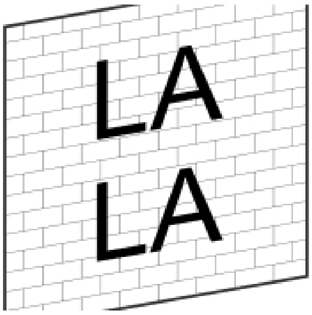
To see the displacemenet settings - select gradient based map placeholders and look at values at **Placeholder import settings / Distortions** window

displacement map | result on sample 1 | result on sample 2

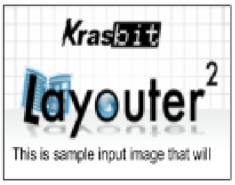
Grayscale map. Left to right gradient.
Disabled horizontal pixel movement via intensity settings: **Displacement intensity horizontal**=0, **Displacement intensity vertical** =100
Without inflate bitmap: **Inflate bitmap X, Y**=0

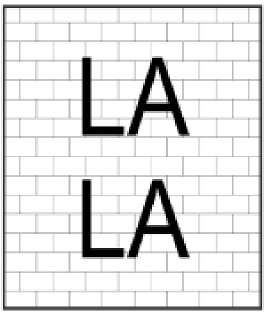




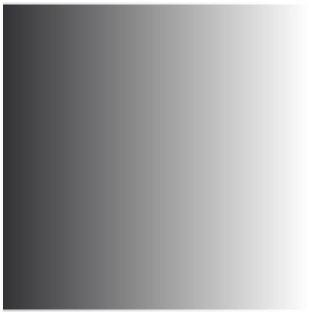


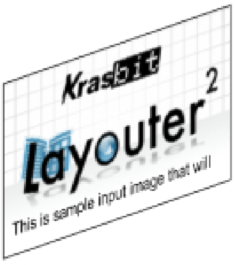
Grayscale map. Left to right gradient.
Disabled vertical pixel movement via intensity settings: **Displacement intensity horizontal**=100, **Displacement intensity vertical** =0
Without inflate bitmap: **Inflate bitmap X, Y**=0

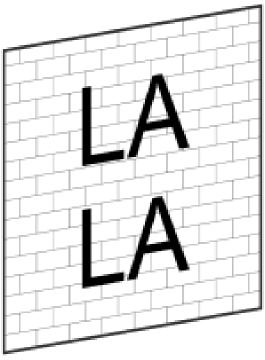




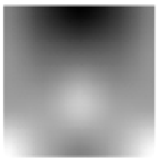
Grayscale map. Left to right gradient.
Enabled pixel movement at both vertical (Y) and horizontal (X) axis via intensity settings: **Displacement intensity horizontal**=100, **Displacement intensity vertical** =100
Iinflate bitmap at Y axis: **Inflate bitmap X**=0, **Y**=100 to let pixels flow out of original boundary

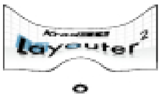


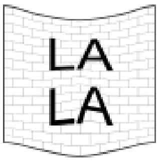


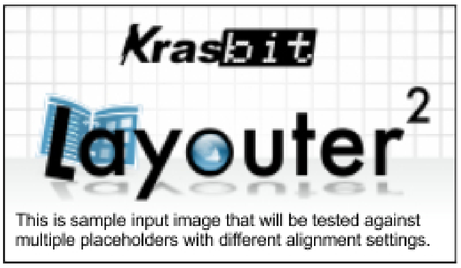


Grayscale map. Custom mesh fill.
Disabled horizontal pixel movement via intensity settings: **Displacement intensity horizontal**=0, **Displacement intensity vertical** =100
Without inflate bitmap: **Inflate bitmap X, Y**=0





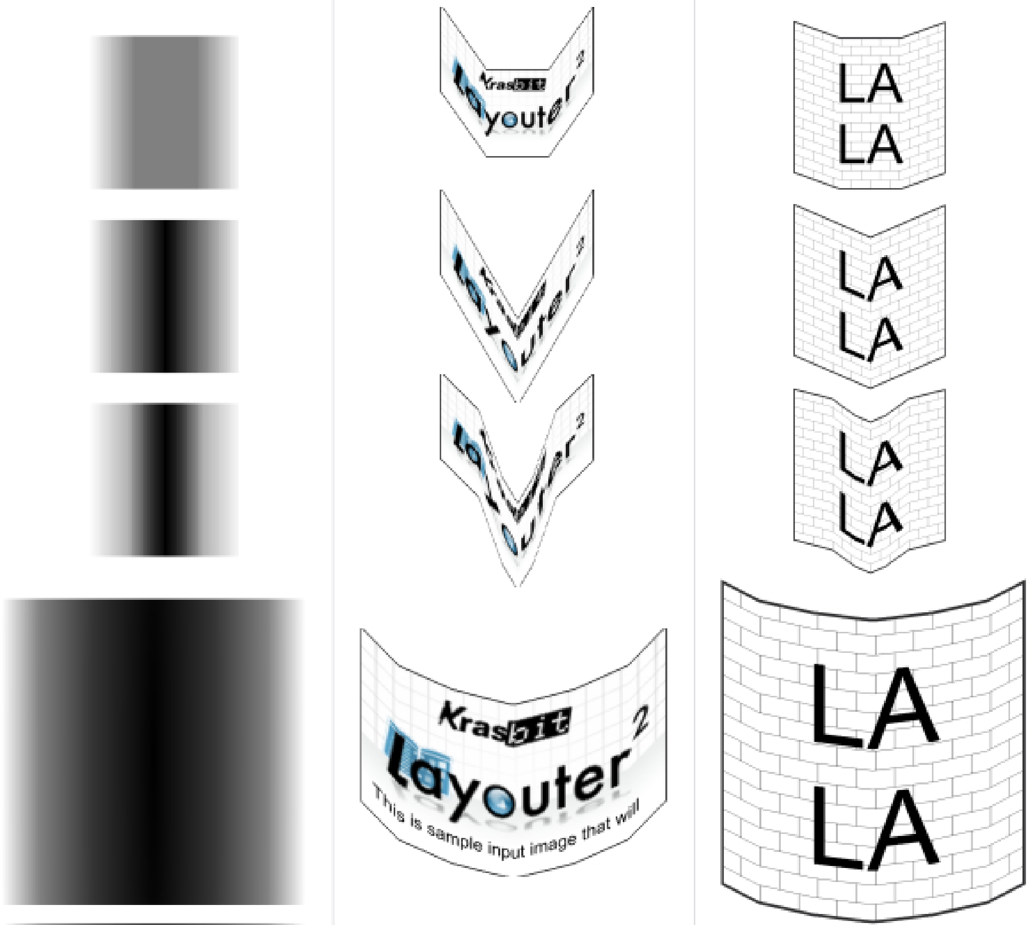




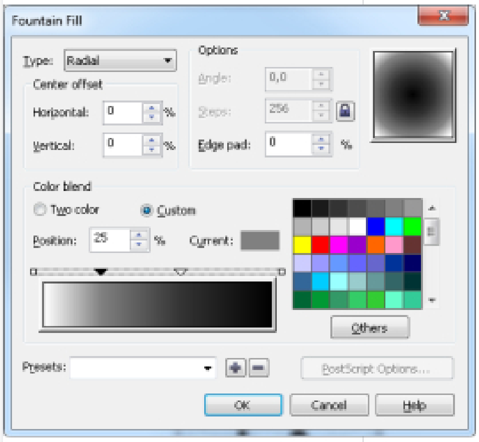
This is sample input image that will be tested against multiple placeholders with different alignment settings.

To see the displacemenet settings - select gradient based map placeholders and look at values at **Placeholder import settings / Distortions** window

displacement map | result on sample 1 | result on sample 2



Intensity horizontally disabled (intensity X=0).
Intensity Y=60,
inflate Y=70
Radial Gradient with custom settings for non linear settings. Then convered to bitmap slice. which is later stretched to square. Finally map produces something closer to rounded mug. You can still research for better roundness via adding more custom control points in gradient (fountain fill advanced settings) settings



Fountain Fill

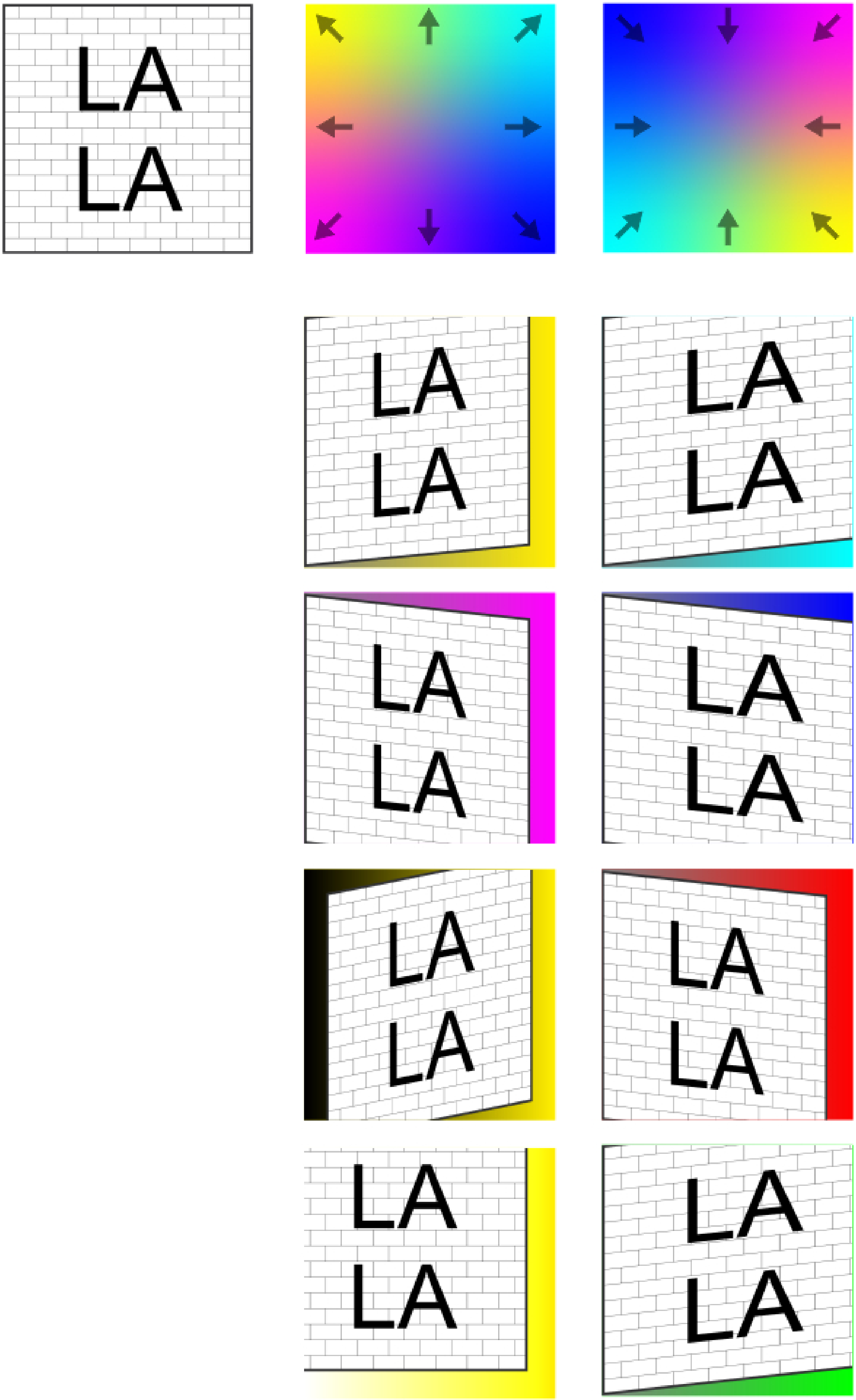
Type: Radial

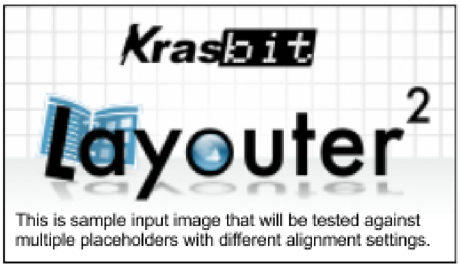
Center offset
Horizontal: 0 %
Vertical: 0 %

Options
Angle: 0,0
Steps: 256
Edge pad: 0 %

Color blend
○ Two color ● Custom
Position: 25 %
Current:

Presets:

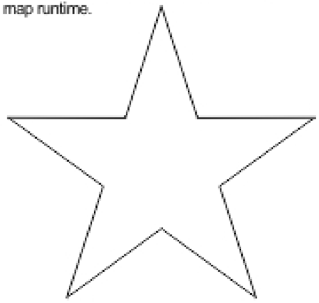

LA
LA

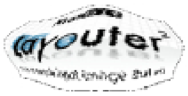

Krasbit
Layouter 2
This is sample input image that will be tested against multiple placeholders with different alignment settings.

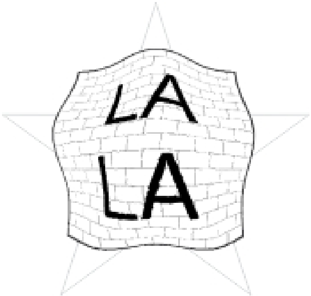
To see the displacemenet settings - select gradient based map placeholders and look at values at **Placeholder import settings / Distortions** window

displacement shape / map | result on sample 1 | result on sample 2

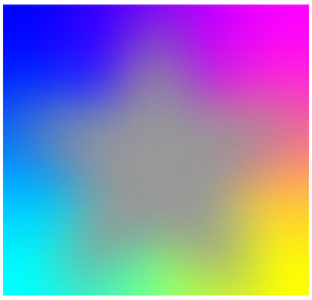
*TIP: You can displace to a custom shape via **"Displace to shape"** option that will create a map runtime.

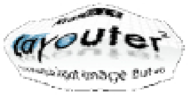


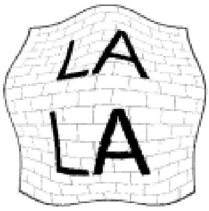


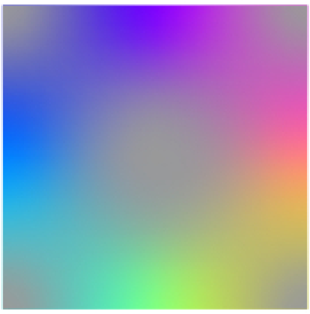


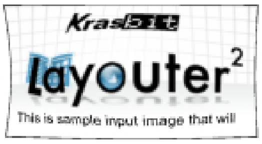
*TIP: You can easily produce displacement maps from selected shape via **"Make displacement map from selected"** button.

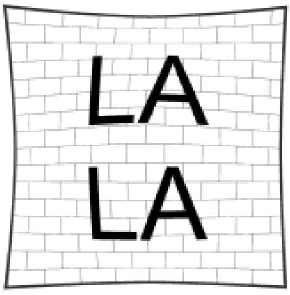


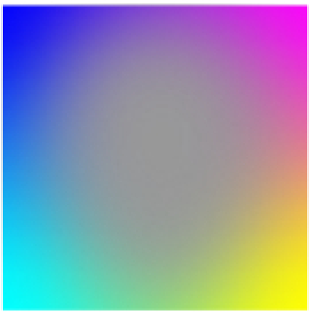


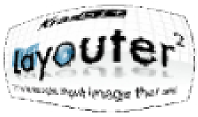


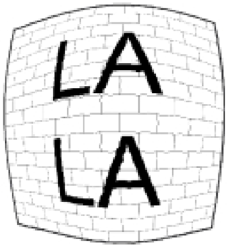


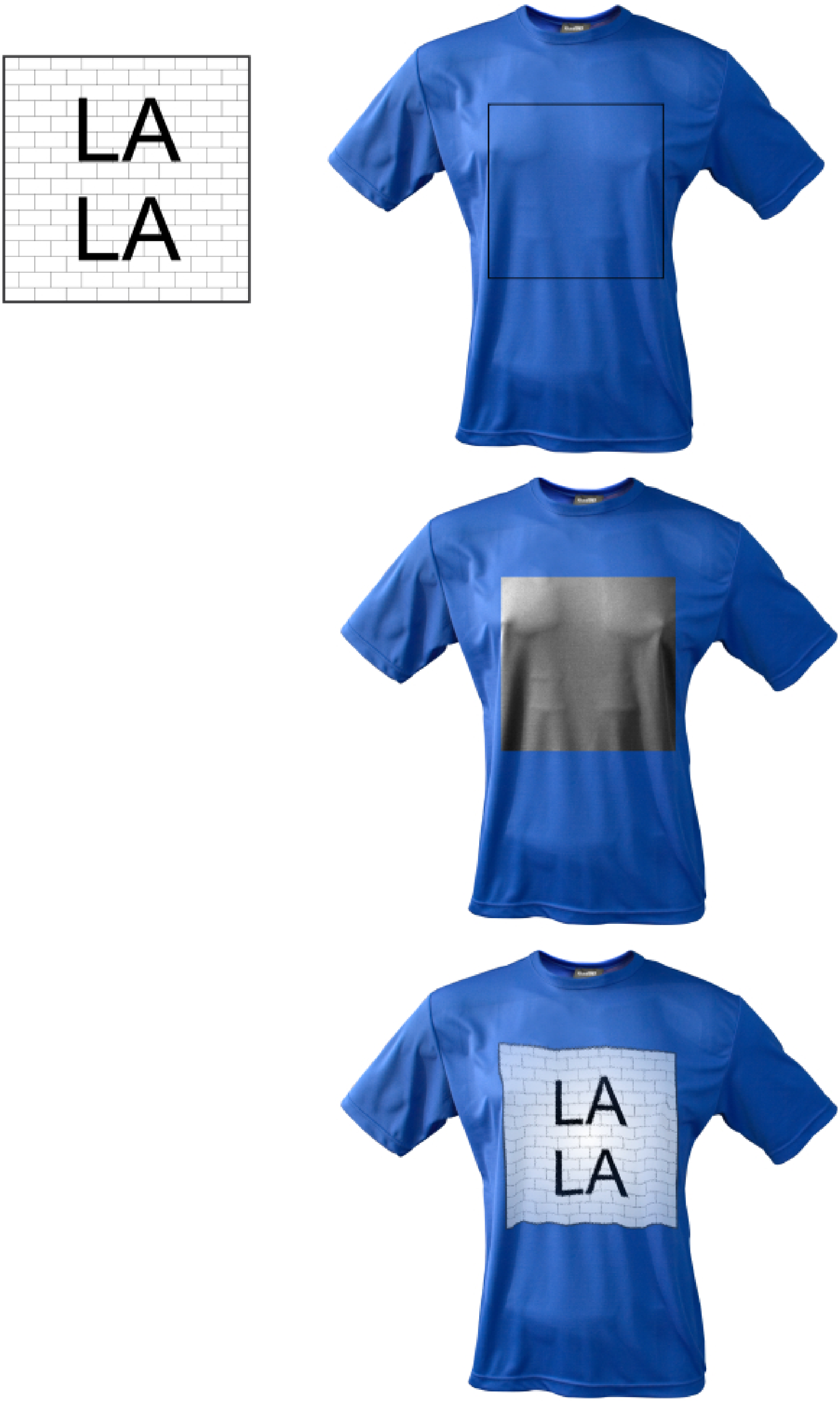




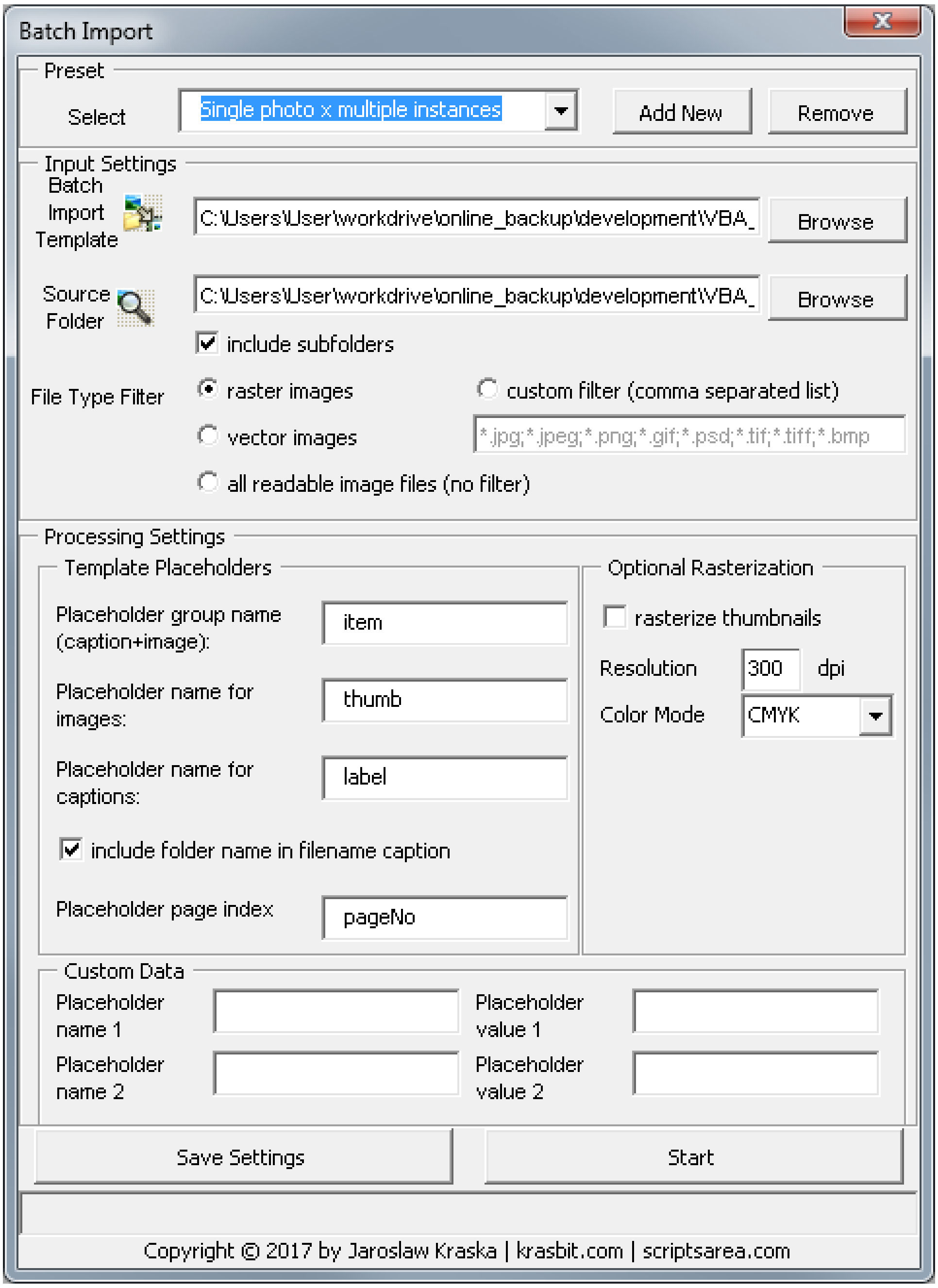







LA
LA
LA
LA

Batch Import



Batch Import

Preset

Select: Single photo x multiple instances

Input Settings

Batch Import Template: C:\Users\User\workdrive\online_backup\development\VBA_

Source Folder: C:\Users\User\workdrive\online_backup\development\VBA_

☑ include subfolders

File Type Filter:
● raster images
○ vector images
○ all readable image files (no filter)
○ custom filter (comma separated list): *.jpg;*.jpeg;*.png;*.gif;*.psd;*.tif;*.tiff;*.bmp

Processing Settings

Template Placeholders

Placeholder group name (caption+image): item

Placeholder name for images: thumb

Placeholder name for captions: label

☑ include folder name in filename caption

Placeholder page index: pageNo

Optional Rasterization

☐ rasterize thumbnails

Resolution: 300 dpi

Color Mode: CMYK

Custom Data

Placeholder name 1:
Placeholder value 1:
Placeholder name 2:
Placeholder value 2:

Copyright © 2017 by Jaroslaw Kraska | krasbit.com | scriptsarea.com

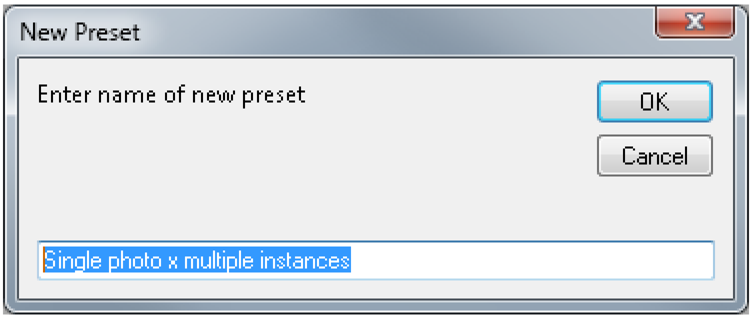
Look into **Krasbit Layouter data** folder **\examples\Batch Import** there will example Batch Import templates. Under **\examples\Batch Import\photos** you will find an example source folder and images. You may use your own folder of images when continue with this tutorial.

Same photo into multiple size items.cdr

You can use Batch Import utility to assembly multiple instances of same photo on one page in order to minimize paper waste when you will print them. You may also prepare a preview mockup of same image at different size or cropping or fit algorithm to let client see difference and choose between white strip Vs cropped borders option.

Steps

1. If you haven't created yet your personalized preset for that use case - start from **Add New** Preset. Application will prompt you to give a name for that preset. I go with *Single photo x multiple instances*. You can easily switch between prepared presets later and do a Batch Import job which is perfectly adjusted to your desired use case.

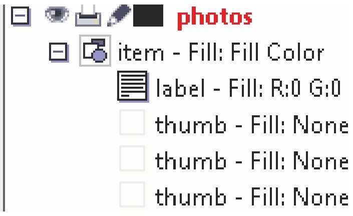


New Preset

Enter name of new preset

Single photo x multiple instances

2. Select **Batch Import template**. Browse to an example file:
   Same photo into multiple size items.cdr
   To make this template proper to you needs - you can adjust this template, work on a copy, or create a new one from scratch later.
   Take a note that this **template** has got a single **item** group that gots multiple **thumb** objects. **Item** and **thumb** are the names of objects under Object manager docker.



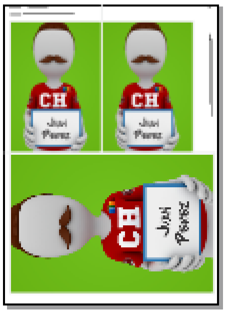
photos
item - Fill: Fill Color
label - Fill: R:0 G:0
thumb - Fill: None
thumb - Fill: None
thumb - Fill: None

   However, they all share a single **label** -paragraph text placeholder, as all the copies of same image will have same origin - file, and there is no reason to repeat that object.

3. Specify your **Source Folder**. This will be a working dir, where you expect to read files from. There are most popular scenarios:
   a. You could have a one working/source folder for all tasks you are going to do. Before Batch Import you could drag/drop or copy paste the amount of images and folders that are required, then delete the content after Batch Import. In this scenario you can specify source folder only once for all your Batch Import jobs that will run at your preset.
   b. You could neatly prepare separate folders with images in advance and specify different source folders more frequently
   c. For the sake of this example I'll pick the *\Batch Import\photos\Coach* folder
4. Specify desired input filters like **File Type Filter** and **include subfolders** option. In this use case raster images are the way to go.
5. Because template we are going to use supports both images (**thumb**) and captions (**label**) that are grouped together as an **item** within CorelDRAW project of this template - we should specify this information in **Template Placeholders** frame. Ensure that you made no typos at settings.
6. If you expect that file will be printed, you should not enable **Optional Rasterization** and let images be placed with original resolution color mode then downscale and specify color mode when PDF for print will be produced. However if you would have a lot of files in folder - better to turn on this option and specify desired resolution like 300 dpi for print or lower value for email preview purpose . Regarding color mode - in case of print - specify CMYK, while for preview purpose of ganged document you may leave as RGB.

7. Preserve your preset settings for later use via **Save Settings** button.
8. Click on start and wait a while. Progress of Batch Import will be shown at bottom of Batch Import utility window.
9. You should see a CorelDRAW generated a document having as many pages as input files in source folder (3). And each page gots a copies of the file as many time as thumb placeholders available on page(3). All the thumb placeholders respects and supports all the features of **Placeholder import settings**. Refer to Krasbit Layouter 2 - User's manual to see more details.
10. Save and/or print your document.

Results



Page 1

Page 2

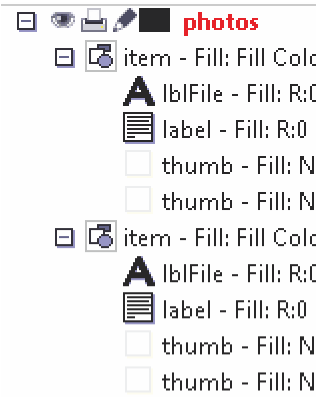
Page 3

Different photos into multiple size items.cdr

You can use Batch Import utility to assembly multiple instances of different or same photos on one page in order to minimize paper waste when you will print them.

Steps

1. If you haven't created yet your personalized preset for that use case - start from **Add New** Preset. Application will prompt you to give a name for that preset. I go with *Multiple photo x multiple instances*. You can easily switch between prepared presets later and do a Batch Import job which is perfectly adjusted to your desired use case.
2. Select **Batch Import template**. Browse to an example file:
   Different photos into multiple size items.cdr
   Take a note that this **template** has got multiple **item** groups that gots multiple **thumb** objects each. **Item** and **thumb** are the names of objects under Object manager docker. Each of **item** group got a single **label** -paragraph text placeholder that is shared by same **thumb** placeholders that will have same origin - file.

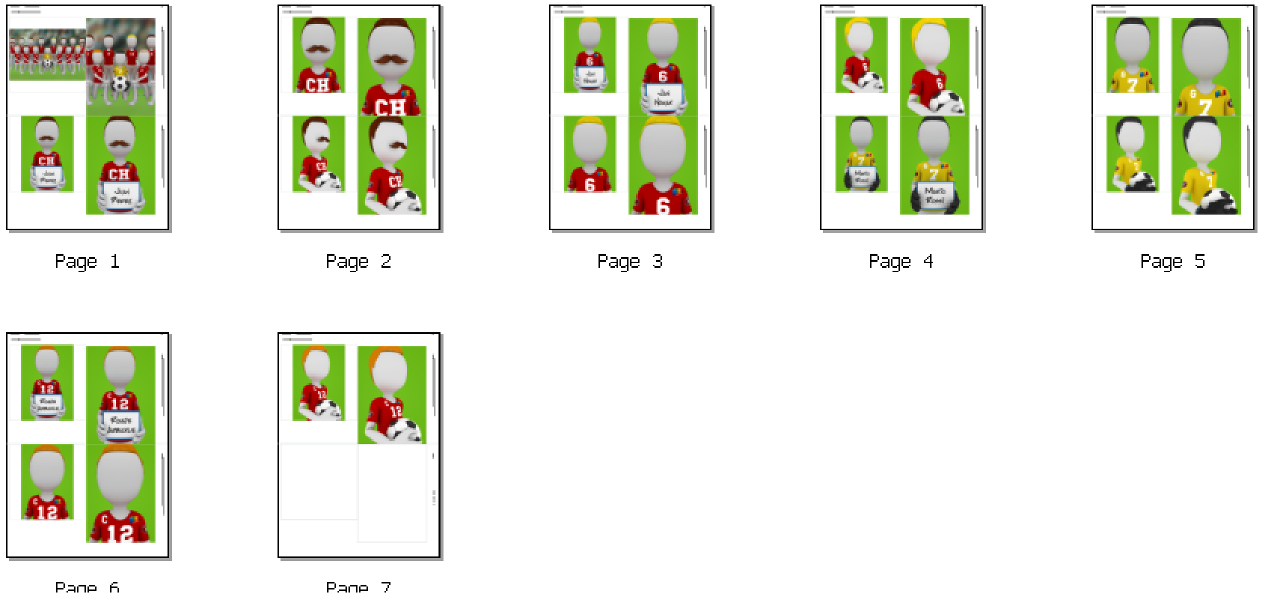


photos
item - Fill: Fill Colc
lblFile - Fill: R:(
label - Fill: R:0
thumb - Fill: N
thumb - Fill: N
item - Fill: Fill Colc
lblFile - Fill: R:(
label - Fill: R:0
thumb - Fill: N
thumb - Fill: N

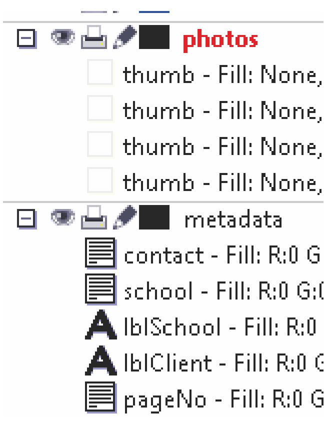
3. Specify your **Source Folder**. This will be a working dir, where you expect to read files from. Let's use *\Batch Import\photos\* folder
4. Specify desired input filters like **File Type Filter** and **include subfolders** option. In this use case raster images are the way to go and include subfolders should be selected to read images from subdirectories.

5. Because template we are going to use supports both images (**thumb**) and captions (**label**) that are grouped together as an **item** within CorelDRAW project of this template - we should specify this information in **Template Placeholders** frame. Ensure that you made no typos at settings.
6. Enable **Optional Rasterization** and use 96 dpi RGB for a preview purpose.
7. In this step I'll show you how to add some additional personalized data to rendered content which may help you later manage orders. Fill **Custom Data** form with this placeholder name-value pairs:
    a. Placeholder name 1: **school** (this is the name given to variable 1 placeholder inside template) and Placeholder Value 1= *Krasbit Layouter Class* (this is the value that is going to be injected into personalized custom data placeholder)
    b. Placeholder name 2: **client** (this is the name given to variable 2 placeholder inside template) and Placeholder Value 2= *My example client* (this is the value that is going to be injected into personalized custom data placeholder)
8. Click on start and wait a while. Progress of Batch Import will be shown at bottom of Batch Import utility window.
9. You should see a CorelDRAW generated a document having as many pages as input files in source folder and subdirectories. And each page gots a copies multiple files (2) per page and each file has copies as many time as **thumb** placeholders available within **item** group.
10. Save and/or print your document.

Results



Thumbnailer for all images in folder.cdr


photos
thumb - Fill: None,
thumb - Fill: None,
thumb - Fill: None,
thumb - Fill: None,
metadata
contact - Fill: R:0 G
school - Fill: R:0 G:(
lblSchool - Fill: R:0
lblClient - Fill: R:0 G
pageNo - Fill: R:0 G

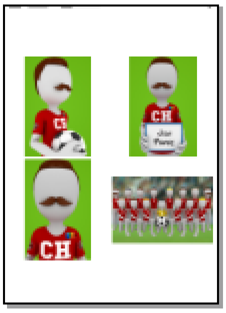
You can use Batch Import utility to create a thumbnail like overview from all images in your folder (and optionally subfolders). You can do this with or without presenting file captions. In such scenario it is recommended to enable optional rasterization in order to reduce resolutions of imported and scaled down images and save file size effectively.

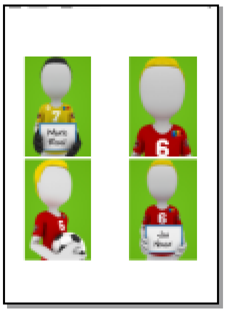
This example project does not have group item as previous examples nor label text placeholder for file caption, only multiple thumb placeholders for importing source folder files. However you still can render that simplified version of template.

Steps

Adjust *Multiple photo x multiple instances* preset settings or create another preset that expects *Thumbnailer for all images in folder.cdr* as **Batch Import Template** and clear fields **placeholder group name** and **placeholder name for captions** (so **item** group or **label** will no longer be required during processing). Leave only **placeholder name for images** with value of field = **thumb** as this is the object name in template.

Results





Products preview mockup.cdr

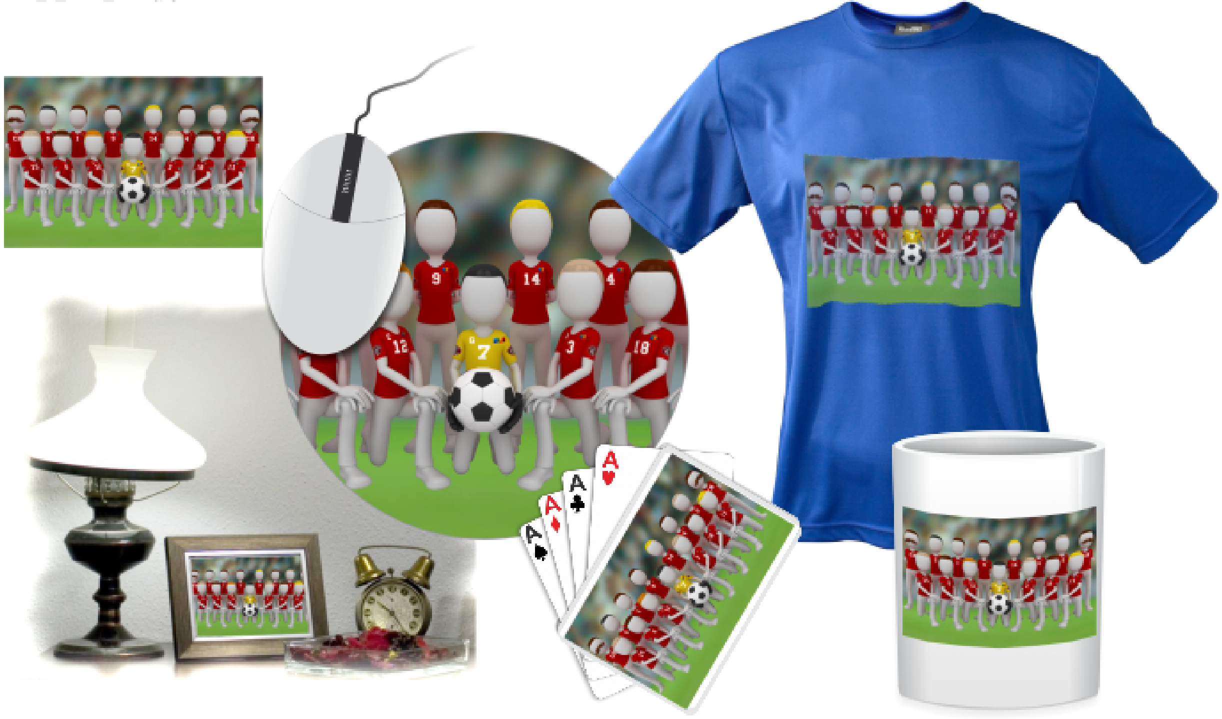
You can use Batch Import to show beautiful previews of your products that are based on printed images.

Steps

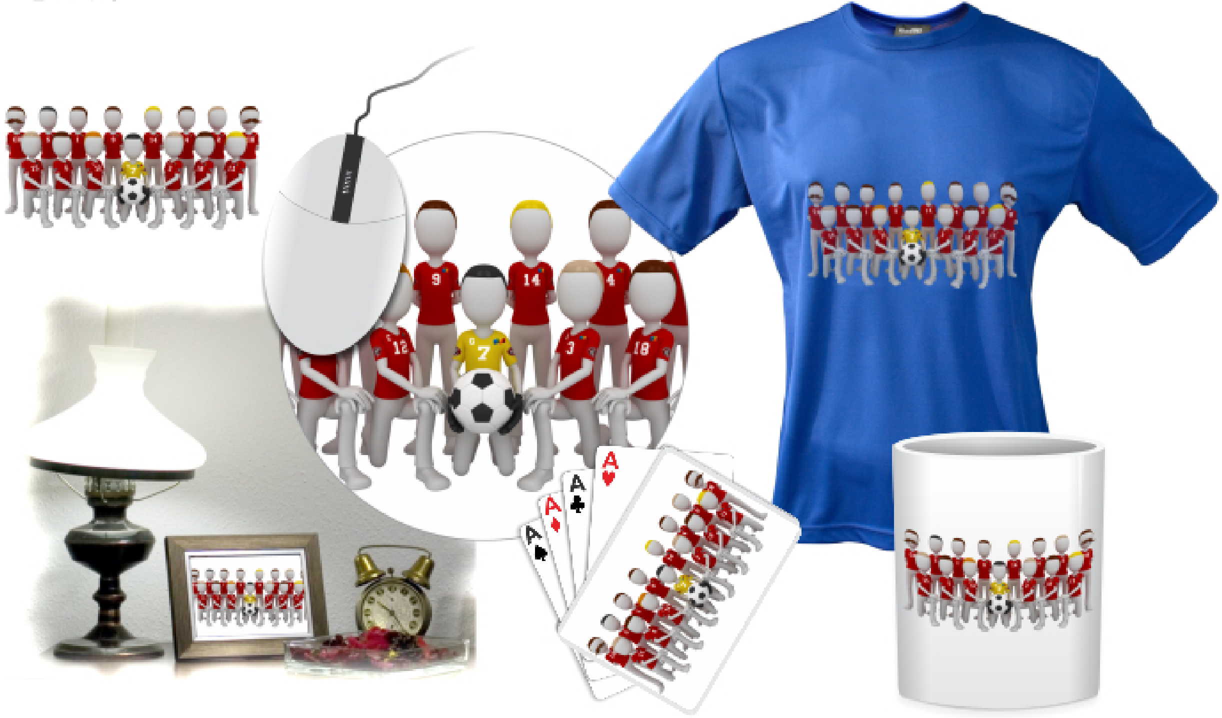
Specify *\examples\Batch Import\ Product preview mockup.cdr* as **Batch Import Template** and set fields for placeholders the object name in template. This is:
placeholder name for images with value of field = **thumb, placeholder group name** with value of field =**item**, **placeholder name for captions** with value field=**label**.
Select **source folder** to *\examples\Batch Import\photos* and unselect **include subfolders** as we want only import 2 photos that exist there and skip subfolders in example.

Results

Lions_0_team_NGS.jpg



team_shoot.png


9
14
4
12
7
3
18
A
A
A
A