

Apply asset renditions

August 16, 2022 • David Klement • 3 min read

Asset renditions are customizations of images in your content items. In this tutorial, you'll learn how to obtain a definition of such a customization using the [Delivery API](#), so you can use it in your app.

Key points

- An asset rendition is a customization of an image used to make the image fit the particular content needs.
- To apply a rendition in your app, get an item with an image and append the rendition parameters to the URL of the image.
- Combine renditions with the image transformation options to optimize the images for each channel your app serves.

What is an asset rendition

Generally, a rendition crops and resizes the image it's applied to. Using a rendition on an image never modifies the image itself. It's only additional data telling your app which part of the image to display.

Combine renditions with image transformation

Content creators use asset renditions when they add an image to content. When you develop your app, you use [image transformation](#) to optimize the images for each channel where your app displays the content.

Asset renditions use a subset of the image transformation URL parameters. Combine asset renditions with image transformation to optimize the customized images by adjusting their size, format, or compression.

1. Retrieve an asset rendition

To retrieve an asset rendition programmatically, you need to [get the content item with the image](#) using the Delivery API:

cURL

```
1 | curl --request GET \  
2 |   --url 'https://deliver.kontent.ai/<YOUR_PROJECT_ID>/items/my_article' \  
   --header 'content-type: application/json'
```

The response contains a reference to the image together with the rendition information at the end:

JSON

```

1 | "teaser_image": {
2 |   "type": "asset",
3 |   "name": "Teaser image",
4 |   "value": [
5 |     {
6 |       "name": "construction-insurance-header.jpg",
7 |       "description": null,
8 |       "type": "image/jpeg",
9 |       "size": 402916,
10 |      "url": "https://assets-us-01.kc-usercontent.com:443/8d20758c-d74c-4f59-ae04-
11 | ee928c0816b7/7bf69996-30a0-49b0-ad8c-09f6e0a04e44/construction-insurance-header.jpg",
12 |       "width": 2124,
13 |       "height": 767,
14 |       "renditions": {
15 |         "default": {
16 |           "rendition_id": "b6ed90d0-e544-4b93-9212-fc041279fa90",
17 |           "preset_id": "a6d98cd5-8b2c-4e50-99c9-15192bce2490",
18 |           "width": 600,
19 |           "height": 300,
20 |           "query": "w=600&h=300&fit=clip&rect=1247,183,600,300"
21 |         }
22 |       }
23 |     }
24 |   ]
25 | }

```

If the [image with a rendition is in a component](#), look for the `modular_content` section in the response.

2. Apply the rendition to the image

In the image object above, you see the key `query` at the bottom. Its value holds the URL query string parameters with the rendition specification:

JSON

```
1 | "query": "w=600&h=300&fit=clip&rect=1247,183,600,300"
```

You need to append this query string to the image's URL to get the customized version of the image. The URL is in the value of the `url` key:

JSON

```
1 | "url": "https://assets-us-01.kc-usercontent.com:443/8d20758c-d74c-4f59-ae04-
2 | ee928c0816b7/7bf69996-30a0-49b0-ad8c-09f6e0a04e44/construction-insurance-header.jpg"
```



The source image with no renditions applied

When combined with the query string parameters, you get the final URL to retrieve the customized image:

cURL

```
1 | https://assets-us-01.kc-usercontent.com/8d20758c-d74c-4f59-ae04-ee928c0816b7/7bf69996-30a0-49b0-ad8c-09f6e0a04e44/construction-insurance-header.jpg?  
   | w=600&h=300&fit=clip&rect=1247,183,600,300
```



The image customized by the URL parameters

After you get the image URL and the rendition query string parameters like this, you can display the image in your app. But before you display it, you can also use the [image transformation](#) to optimize the image specifically for each channel your app is serving.

What's next?

You've learned how to get asset renditions from Kontent.ai to your app using the Deliver API. We also recommend you to combine the rendition parameters with the image transformation options to serve each channel with what suits it the best.

Here are some tips on related resources:

- [A guide on image optimization](#)
- [An image transformation reference](#) – a useful resource when you're optimizing images
- [Optimize your content for search engines](#)