



# Quick Note: fetching and displaying images

In the old days you could get away with fetching an image, assigning the file to the `src` attribute of your image and then proudly display your Javascript chops to the rest of the world.

The times are changing, young padawan.

Now you have to do several things.

1. Fetch the image (using the fetch API)
2. If the response was successful convert it to a BLOB using the method on the response object. If the response was not ok throw an error and pass to the catch block
3. Create a URL from the blob using [createObjectUrl](#)
4. Create an image element
5. Attach the URL to the image's `src` attribute
6. Append the image to the body

```
fetch(
  'https://i0.wp.com/rivendellweb.net/blog/wp-content/uploads/2017/12/afs
) // 1
.then(response => {
  if (response.ok) {
    // 2
    return response.blob();
  } else {
    throw new Error('Network response was not ok.');
```

'There has been a problem: '

This is good for single images but may not lend itself to multiple images. In this case we might use the URL/path to the image in the JSON we fetch from an API. Working on a Codepen to validate this assertion.