

How to Download Images Without "Save As" and Store as Base64 String

Download Images When "Save As" Is Disabled

If a website disables right-click or the "Save image as" option, you can still download images using browser developer tools:

Steps Using Inspect Element

1. Open Developer Tools:

- Press F12 or Ctrl+Shift+I (Windows/Linux) or Cmd+Option+I (Mac) in your browser to open developer tools.

2. Select the Image:

- Click the *Element Inspector* icon and then click on the image.

3. Find the Image URL:

- In the HTML panel, find the `` tag and look for the `src` attribute.
- The image URL may appear in the side panel or directly in the code [\[1\]](#) [\[2\]](#) [\[3\]](#) [\[4\]](#).

4. Open the Image in a New Tab:

- Right-click the URL and select "Open in new tab."
- If right-click is disabled, copy the URL and paste it into a new browser tab.

5. Save the Image:

- In the new tab displaying only the image, right-click and choose "Save image as..."
- If right-click is still disabled, use the browser menu or take a screenshot as a last resort (note: this may not give you the original quality).

Tip:

You can also press Ctrl+S to save the entire web page. This places all image assets into a folder on your computer, from where you can pick the files you need [\[5\]](#).

How to Store an Image as a Base64 String

Base64 encoding allows you to store image data as a string, useful for embedding directly in HTML, CSS, or databases.

Using Online Tools

- Upload or paste the image URL into an online Base64 encoder.
- The tool generates a Base64 string for you to copy and use^[6].

Using JavaScript in the Browser

Convert an Image URL to Base64:

```
fetch('https://example.com/image.jpg')
  .then(res => res.arrayBuffer())
  .then(buffer => {
    let binary = '';
    new Uint8Array(buffer).forEach(b => binary += String.fromCharCode(b));
    let base64 = btoa(binary);
    console.log(base64); // The base64 string
  });
```

- This code fetches the image, reads its binary data, and converts it to a Base64 string^[6] ^[7].

Convert an Image Tag to Base64 via Canvas:

```
let img = document.querySelector('img');
let canvas = document.createElement('canvas');
canvas.width = img.width;
canvas.height = img.height;
let ctx = canvas.getContext('2d');
ctx.drawImage(img, 0, 0);
let base64string = canvas.toDataURL(); // Data URL with base64-encoded content
```

- For small images, you can now use this base64string directly in `` or elsewhere^[7].

Practical Notes

- Base64-encoded images are larger in size than the original files and can bloat your HTML or database^[8].
- Use this technique when you need to embed images directly or transfer images as text data.

Summary Table

Task	Solution
Download image with no "Save"	Use Inspect Element, find URL, open/save
Store image as Base64 string	Use online tools, JavaScript, or Node.js code

These methods allow you to extract and convert almost any visible image from a webpage, regardless of download restrictions, and store or use them as base64-encoded data^[1] ^[6] ^[7].

1. <https://phancybox.com.au/how-to-save-an-image-from-a-website/>
2. <https://www.adobe.com/acrobat/hub/extract-images-from-a-web-page.html>
3. <https://www.alphr.com/save-images-when-right-click-is-disabled/>
4. <https://www.instagram.com/reel/DBjplYmRz4P/>
5. https://www.reddit.com/r/TechnologyProTips/comments/mnwwdv/tpt_if_a_web_site_doesnt_allow_you_to_save/
6. <https://formspring.io/blog/image-to-base64/>
7. <https://pqina.nl/blog/convert-an-image-to-a-base64-string-with-javascript/>
8. <https://stackoverflow.com/questions/9722603/storing-image-in-database-directly-or-as-base64-data>