Window: createImageBitmap() method

- Syntax
  - Parameters
  - Return value
- Examples
  - Creating sprites from a sprite sheet
- Specifications
- Browser compatibility
- See also

{{APIRef("Canvas API")}}

The `createImageBitmap()` method of the {{domxref("Window")}} interface creates a bitmap from a given source, optionally cropped to contain only a portion of that source. It accepts a variety of different image sources, and returns a {{jsxref("Promise")}} which resolves to an {{domxref("ImageBitmap")}}.

Syntax

```
createImageBitmap(image)
createImageBitmap(image, options)
createImageBitmap(image, sx, sy, sw, sh)
createImageBitmap(image, sx, sy, sw, sh, options)
```

Parameters

- `image`
  - : An image source, which can be any one of the following:
    - {{domxref("HTMLImageElement")}}
    - {{domxref("SVGImageElement")}}
    - {{domxref("HTMLVideoElement")}}
    - {{domxref("HTMLCanvasElement")}}
    - {{domxref("Blob")}}
    - {{domxref("ImageData")}}
    - {{domxref("ImageBitmap")}}
    - {{domxref("OffscreenCanvas")}}
    - {{domxref("VideoFrame")}}
- `sx`
  - : The x coordinate of the reference point of the rectangle from which the `ImageBitmap` will be extracted.
- `sy`
  - : The y coordinate of the reference point of the rectangle from which the `ImageBitmap` will be extracted.
- `sw`
  - : The width of the rectangle from which the `ImageBitmap` will be extracted. This value can be negative.
- `sh`
  - : The height of the rectangle from which the `ImageBitmap` will be extracted. This value can be negative.
- `options` {{optional_inline}}
  - : An object that sets options for the image's extraction. The available options are:
    - `imageOrientation`
      - : Specifies how the bitmap image should be oriented.
        - `from-image`
          - : Image oriented according to EXIF orientation metadata, if present (default).
        - `flipY`
          - : Image oriented according to EXIF orientation metadata, if present, and then flipped vertically.
        - `none`
          - : Image oriented according to image encoding, ignoring any metadata about the orientation (such as EXIF metadata, that might be added to an image to indicate that the camera was turned sideways to capture the image in portrait mode).
    - `premultiplyAlpha`
      - : Specifies whether the bitmap's color channels should be premultiplied by the alpha channel. One of `none`, `premultiply`, or `default` (default).
    - `colorSpaceConversion`

    - : Specifies whether the image should be decoded using color space conversion. Either `none` or `default` (default). The value `default` indicates that implementation-specific behavior is used.
  - `resizeWidth`
    - : A long integer that indicates the output width.
  - `resizeHeight`
    - : A long integer that indicates the output height.
  - `resizeQuality`
    - : Specifies the algorithm to be used for resizing the input to match the output dimensions. One of `pixelated`, `low` (default), `medium`, or `high`.

Return value

A {{jsxref("Promise")}} which resolves to an {{domxref("ImageBitmap")}} object containing bitmap data from the given rectangle.

Examples

Creating sprites from a sprite sheet

This example loads a sprite sheet, extracts individual sprites, and then renders each sprite to the canvas. A sprite sheet is an image containing multiple smaller images, each of which you want to be able to render separately.

```
Original image:
<img src="50x50.jpg" />
<hr />
<canvas id="myCanvas"></canvas>
```

```
canvas {
  border: 2px solid green;
}
```

```
const canvas = document.getElementById("myCanvas"),
  ctx = canvas.getContext("2d"),
  image = new Image();

// Wait for the sprite sheet to load
```

```
image.onload = () => {
  Promise.all([
    // Cut out two sprites from the sprite sheet
    createImageBitmap(image, 0, 0, 32, 32),
    createImageBitmap(image, 32, 0, 32, 32),
    createImageBitmap(image, 0, 0, 50, 50, { imageOrientation: "flipY" }),
  ]).then((sprites) => {
    // Draw each sprite onto the canvas
    ctx.drawImage(sprites[0], 0, 0);
    ctx.drawImage(sprites[1], 32, 32);
    ctx.drawImage(sprites[2], 64, 64);
  });
};

// Load the sprite sheet from an image file
image.src = "50x50.jpg";
```

{{EmbedLiveSample("Creating sprites from a sprite sheet", "100%", "250")}}

Specifications

{{Specifications}}

Browser compatibility

{{Compat}}

See also

- {{domxref("WorkerGlobalScope.createImageBitmap()")}}
- {{domxref("CanvasRenderingContext2D.drawImage()")}}
- {{domxref("ImageData")}}

□□□ #2
□ □□□ □□□ 2026-10-31 03:27:53 UTC
□ □□□ □□□ 2026-10-31 03:27:53 UTC