# [HTMLCanvasElement.toDataURL()](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/toDataURL)

HTMLCanvasElement.toDataURL() 方法返回一个包含图片展示的 [data URI](https://developer.mozilla.org/en-US/docs/Web/HTTP/data_URIs) 。可以使用 type 参数其类型，默认为 [PNG](https://en.wikipedia.org/wiki/Portable_Network_Graphics) 格式。图片的分辨率为96dpi。

* 如果画布的高度或宽度是0，那么会返回字符串“data:,”。
* 如果传入的类型非“image/png”，但是返回的值以“data:image/png”开头，那么该传入的类型是不支持的。
* Chrome支持“image/webp”类型。

## **语法**

*canvas.toDataURL(type, encoderOptions);*

### 参数

**type**可选

图片格式，默认为 image/png

**encoderOptions**可选

图片质量。取值范围为0到1。如果指定图片格式为 image/jpeg 或 image/webp。如果超出取值范围 或 不填质量参数，将会使用默认值0.92。其他参数会被忽略。

### 返回值

包含 [data URI](https://developer.mozilla.org/en-US/docs/Web/HTTP/data_URIs) 的[DOMString](https://developer.mozilla.org/zh-CN/docs/Web/API/DOMString" \o "一个UTF-16字符串，JavaScript 正是使用了这种编码的字符串，所以 DOMString 直接被映射为（is mapped directly to）String。)。

## **示例**

有如下[<canvas>](https://developer.mozilla.org/zh-CN/docs/Web/HTML/Element/canvas" \o "<canvas>元素可被用来通过脚本（通常是JavaScript）绘制图形。比如,它可以被用来绘制图形,制作图片集合,甚至用来实现动画效果。你可以(也应该)在元素标签内写入可提供替代的的代码内容，这些内容将会在在旧的、不支持<canvas>元素的浏览器或是禁用了JavaScript的浏览器内渲染并展现。)元素

<canvas id="canvas" width="5" height="5"></canvas>

可以用这样的方式获取一个 data-URL

var canvas = document.getElementById("canvas");var dataURL = canvas.toDataURL();

console.log(dataURL);// "data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAAUAAAAFCAYAAACNby// blAAAADElEQVQImWNgoBMAAABpAAFEI8ARAAAAAElFTkSuQmCC"

### 设置jpegs图片的质量

var fullQuality = canvas.toDataURL("image/jpeg", 1.0);// data:image/jpeg;base64,/9j/4AAQSkZJRgABAQ...9oADAMBAAIRAxEAPwD/AD/6AP/Z"var mediumQuality = canvas.toDataURL("image/jpeg", 0.5);var lowQuality = canvas.toDataURL("image/jpeg", 0.1);

### 示例：动态更改图片

可以使用鼠标事件来动态改变图片（这个例子中改变图片灰度）。

#### HTML

<img class="grayscale" src="myPicture.png" alt="Description of my picture" />

#### JavaScript

window.addEventListener("load", removeColors);

function showColorImg() {

this.style.display = "none";

this.nextSibling.style.display = "inline";}

function showGrayImg() {

this.previousSibling.style.display = "inline";

this.style.display = "none";}

function removeColors() {

var aImages = document.getElementsByClassName("grayscale"),

nImgsLen = aImages.length,

oCanvas = document.createElement("canvas"),

oCtx = oCanvas.getContext("2d");

for (var nWidth, nHeight, oImgData, oGrayImg, nPixel, aPix, nPixLen, nImgId = 0; nImgId < nImgsLen; nImgId++) {

oColorImg = aImages[nImgId];

nWidth = oColorImg.offsetWidth;

nHeight = oColorImg.offsetHeight;

oCanvas.width = nWidth;

oCanvas.height = nHeight;

oCtx.drawImage(oColorImg, 0, 0);

oImgData = oCtx.getImageData(0, 0, nWidth, nHeight);

aPix = oImgData.data;

nPixLen = aPix.length;

for (nPixel = 0; nPixel < nPixLen; nPixel += 4) {

aPix[nPixel + 2] = aPix[nPixel + 1] = aPix[nPixel] = (aPix[nPixel] + aPix[nPixel + 1] + aPix[nPixel + 2]) / 3;

}

oCtx.putImageData(oImgData, 0, 0);

oGrayImg = new Image();

oGrayImg.src = oCanvas.toDataURL();

oGrayImg.onmouseover = showColorImg;

oColorImg.onmouseout = showGrayImg;

oCtx.clearRect(0, 0, nWidth, nHeight);

oColorImg.style.display = "none";

oColorImg.parentNode.insertBefore(oGrayImg, oColorImg);

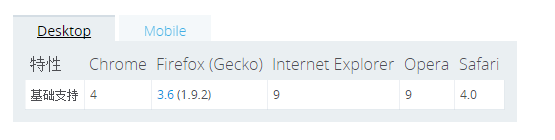
}}

## **规范**

|  |  |  |
| --- | --- | --- |
| **规范** | **状态** | **备注** |
| [WHATWG HTML Living Standard HTMLCanvasElement.toDataURL](https://html.spec.whatwg.org/multipage/scripting.html" \l "dom-canvas-todataurl) | Living Standard | No change since the latest snapshot, [HTML5](http://www.w3.org/TR/html5/" \o "HTML5) |
| [HTML5.1 HTMLCanvasElement.toDataURL](http://www.w3.org/html/wg/drafts/html/master/scripting-1.html" \l "dom-canvas-todataurl) | Working Draft |  |
| [HTML5 HTMLCanvasElement.toDataURL](http://www.w3.org/TR/html5/scripting-1.html" \l "dom-canvas-todataurl) | Recommendation | Snapshot of the [WHATWG HTML Living Standard](https://html.spec.whatwg.org/multipage/" \o "WHATWG HTML Living Standard) containing the initial definition. |

## **浏览器兼容性**

* Desktop



* Mobile

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 特性 | Chrome | Firefox (Gecko) | Internet Explorer | Opera | Safari |
| 基础支持 | 4 | [3.6](https://developer.mozilla.org/en-US/Firefox/Releases/3.6" \o "Released on 2010-01-21.) (1.9.2) | 9 | 9 | 4.0 |

## 

## **参考**

* 定义接口，[HTMLCanvasElement](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement" \o "DOM canvas元素暴露了HTMLCanvasElement接口,该接口提供了用来操作一个canvas元素布局和呈现的属性和方法.HTMLCanvasElement接口继承了element接口的属性和方法.)
* [HTTP](https://developer.mozilla.org/en-US/docs/Web/HTTP)引用中的[Data URIs](https://developer.mozilla.org/en-US/docs/Web/HTTP/data_URIs)

## **文档标签和贡献者**

**此页面的贡献者：** [fbwfbi](https://developer.mozilla.org/zh-CN/profiles/fbwfbi), [jiahui](https://developer.mozilla.org/zh-CN/profiles/jiahui)

**最后编辑者:** [fbwfbi](https://developer.mozilla.org/zh-CN/profiles/fbwfbi), Sep 20, 2016, 2:14:11 AM

**另见**

1. [Canvas API](https://developer.mozilla.org/zh-CN/docs/Web/API/Canvas_API)
2. [HTMLCanvasElement](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement)
3. [属性](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/toDataURL)
   1. [height](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/height" \o "The HTMLCanvasElement.height property is a positive integer reflecting the height HTML attribute of the <canvas> element interpreted in CSS pixels. When the attribute is not specified, or if it is set to an invalid value, like a negative, the default value of 150 is used.) [[我来译!]](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/height$translate" \o "此页面仍未被本地化, 期待您的翻译!)
   2. [mozOpaque](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/mozOpaque" \o "The HTMLCanvasElement.mozOpaque property is a Boolean reflecting the moz-opaque HTML attribute of the <canvas> element. It lets the canvas know whether or not translucency will be a factor. If the canvas knows there's no translucency, painting performance can be optimized.) [[我来译!]](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/mozOpaque$translate" \o "此页面仍未被本地化, 期待您的翻译!)
   3. [width](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/width" \o "The HTMLCanvasElement.width property is a positive integer reflecting the width HTML attribute of the <canvas> element interpreted in CSS pixels. When the attribute is not specified, or if it is set to an invalid value, like a negative, the default value of 300 is used.) [[我来译!]](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/width$translate" \o "此页面仍未被本地化, 期待您的翻译!)
4. [方法](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/toDataURL)
   1. [captureStream()](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/captureStream" \o "The HTMLCanvasElement.captureStream() method returns a CanvasCaptureMediaStream that is a real-time video capture of the surface of the canvas.) [[我来译!]](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/captureStream$translate" \o "此页面仍未被本地化, 期待您的翻译!)
   2. [getContext()](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/getContext" \o "HTMLCanvasElement.getContext() 方法返回canvas 的上下文或者返回 null 如果上下文没有定义.)
   3. [mozFetchAsStream()](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/mozFetchAsStream" \o "The HTMLCanvasElement.mozFetchAsStream() internal method used to create a new input stream that, when ready, would provide the contents of the canvas as image data. However, this non-standard and internal method has been removed.) [[我来译!]](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/mozFetchAsStream$translate" \o "此页面仍未被本地化, 期待您的翻译!)
   4. [mozGetAsFile()](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/mozGetAsFile" \o "The HTMLCanvasElement.mozGetAsFille() method returns a File object representing the image contained in the canvas; this file is a memory-based file, with the specified name. If type is not specified, the image type is image/png.) [[我来译!]](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/mozGetAsFile$translate" \o "此页面仍未被本地化, 期待您的翻译!)
   5. [toBlob()](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/toBlob" \o "HTMLCanvasElement.toBlob() 方法创造Blob对象，用以展示canvas上的图片；这个图片文件可以被缓存或保存到本地，由用户代理端自行决定。如不特别指明，图片的类型默认为 image/png，分辨率为96dpi。)
   6. toDataURL()
   7. [transferControlToOffscreen()](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/transferControlToOffscreen" \o "The HTMLCanvasElement.transferControlToOffscreen() method transfers control to an OffscreenCanvas object, either on the main thread or on a worker.) [[我来译!]](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/transferControlToOffscreen$translate" \o "此页面仍未被本地化, 期待您的翻译!)
5. [继承](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/toDataURL)
   1. [HTMLElement](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLElement)
   2. [Element](https://developer.mozilla.org/zh-CN/docs/Web/API/Element)
   3. [Node](https://developer.mozilla.org/zh-CN/docs/Web/API/Node)
   4. [EventTarget](https://developer.mozilla.org/zh-CN/docs/Web/API/EventTarget)
6. [Related pages for Canvas API](https://developer.mozilla.org/zh-CN/docs/Web/API/HTMLCanvasElement/toDataURL)
   1. [Canvas](https://developer.mozilla.org/zh-CN/docs/Web/API/Canvas)
   2. [CanvasGradient](https://developer.mozilla.org/zh-CN/docs/Web/API/CanvasGradient)
   3. [CanvasImageSource](https://developer.mozilla.org/zh-CN/docs/Web/API/CanvasImageSource)
   4. [CanvasPattern](https://developer.mozilla.org/zh-CN/docs/Web/API/CanvasPattern)
   5. [CanvasRenderingContext2D](https://developer.mozilla.org/zh-CN/docs/Web/API/CanvasRenderingContext2D)
   6. [ImageBitmap](https://developer.mozilla.org/zh-CN/docs/Web/API/ImageBitmap)
   7. [ImageBitmapFactories](https://developer.mozilla.org/zh-CN/docs/Web/API/ImageBitmapFactories)
   8. [ImageBitmapRenderingContext](https://developer.mozilla.org/zh-CN/docs/Web/API/ImageBitmapRenderingContext)
   9. [ImageData](https://developer.mozilla.org/zh-CN/docs/Web/API/ImageData)
   10. [OffscreenCanvas](https://developer.mozilla.org/zh-CN/docs/Web/API/OffscreenCanvas)
   11. [Path2D](https://developer.mozilla.org/zh-CN/docs/Web/API/Path2D)
   12. [RenderingContext](https://developer.mozilla.org/zh-CN/docs/Web/API/RenderingContext)
   13. [TextMetrics](https://developer.mozilla.org/zh-CN/docs/Web/API/TextMetrics)

© 2005-2016 Mozilla 开发者网络及各贡献者

内容可按[这些许可协议](https://developer.mozilla.org/en-US/docs/MDN/About" \l "Copyrights_and_licenses)使用。

* [关于 MDN](https://developer.mozilla.org/docs/MDN/About)
* [条款](https://www.mozilla.org/about/legal/terms/mozilla)
* [隐私](https://www.mozilla.org/privacy/websites/)
* [Cookie](https://www.mozilla.org/privacy/websites/" \l "cookies)
* [贡献代码](https://github.com/mozilla/kuma)

窗体顶端

其他语言：

窗体底端