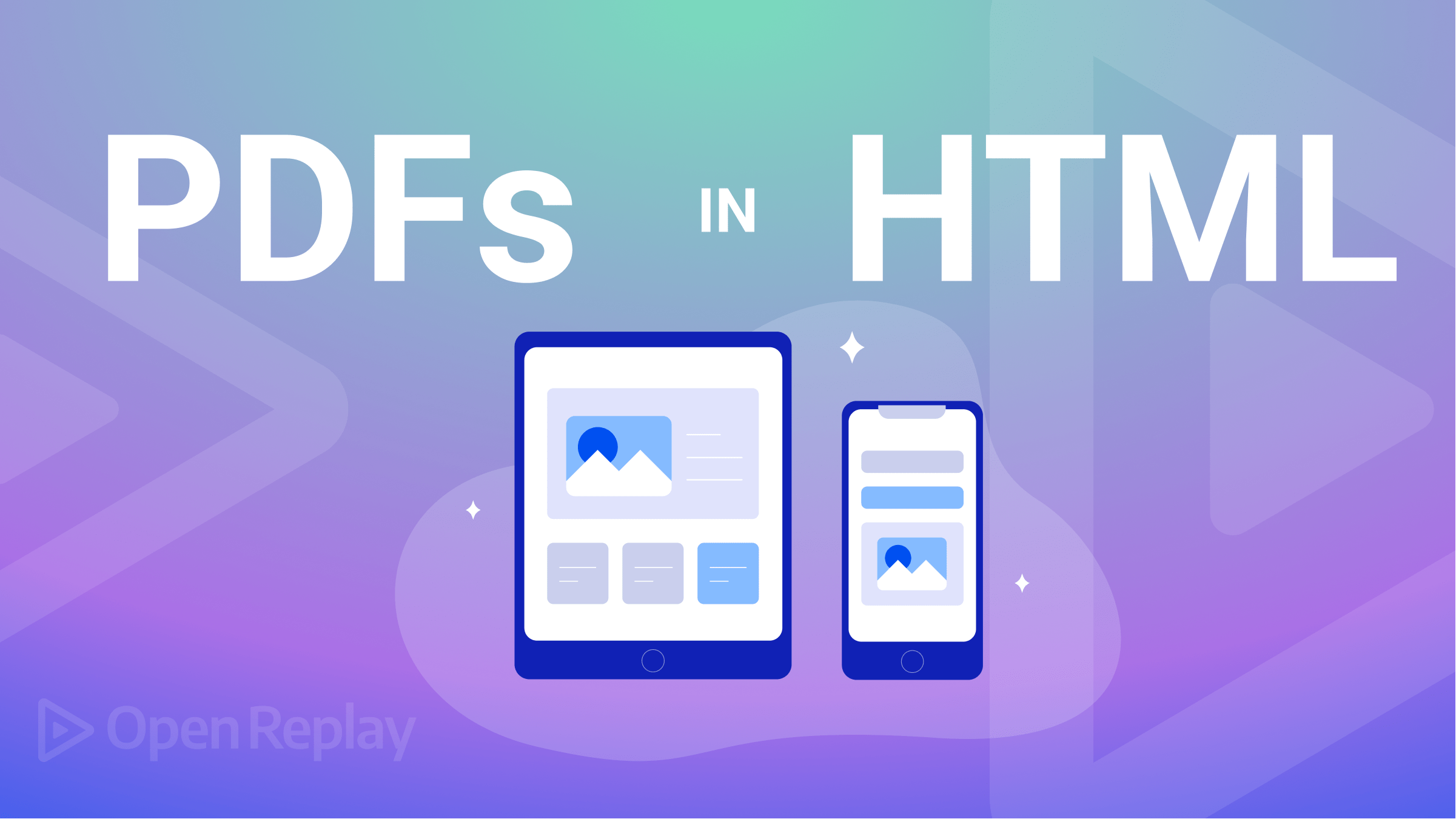
[BLOG](https://blog.openreplay.com/)

[Back](https://blog.openreplay.com/)

# Step by step: embedding PDFs in HTML pages

[Success (Cess) Eriamiantoe](https://blog.openreplay.com/authors/success-(cess)-eriamiantoe)

May 3, 2023 · 5 min read



The Portable Document Format (PDF) provides easy online access to various valuable documents and is convenient when we wish to share something that isn’t just a text, like a brochure or a form. However, displaying PDF files on a website can be difficult and time-consuming — but this article will show you how to use HTML to embed a PDF file into a website so the user can interact with it.

PDF file [Embedment](https://www.netlingo.com/word/embed.php) in HTML is a method we use to embed PDF documents into a web page. We achieve this by using HTML tags containing the PDF file’s source. We can also customize the size and appearance of the embedded PDF files using CSS. Embedding PDF files in HTML allows users to interact with PDF files from within the browser. Users can easily view the file without downloading it or using an external plugin.

There are several advantages of embedding PDF files in HTML, such as:

* **Accessibility**: One significant advantage of including PDF files in HTML is accessibility. PDF files embedded in HTML are accessible to [assistive technologies](https://equidox.co/blog/what-is-assistive-technology-screen-readers-and-other-tools) like Screen Readers.
* **Consistent formatting**: When we embed PDF files in HTML, they appear as static images on the web page. As a result, the composition and layout remain consistent across different devices.
* **Search engine optimization (SEO)**: By embedding PDF files in HTML, we can improve the [SEO](NULL) of our websites. Like any other page, search engines treat PDF files embedded in HTML as part of the website. This way, search engines can index the content of the PDF file so it ranks higher on the search engine results page.
* **Improved user experience**: Users can interact with the embedded files without downloading, and it helps save time and creates a better user experience.

In this article, we’ll show three ways of embedding PDF files in your HTML pages:

* Method 1: Using <iframe> tag.
* Method 2: Using <Object> tag.
* Method 3: Using <embed> tag.

**Method 1: Using <iframe> tag**

In [HTML web pages](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/iframe), the <iframe> tag is one of the most widely used tags to embed PDF files. It is also known as an inline frame because it displays the embedded content within the HTML page. The embedded content is now viewed from the same web page, not in a separate window or tab. It allows us to embed content from other sources like videos and maps.

Example of using the <iframe> tag to embed a PDF file:

<iframe

src="path/to/Clean-Code.pdf"

width="100%"

height="600px"

loading="lazy"

title="PDF-file"

></iframe>

In the example above:

* **Src**: The src attribute specifies our PDF file’s location, which can be on the server or the web. If it’s on the web, we use the PDF file’s URL; if it’s on our local computer or server, we use the file path.

PDF file’s URL:

<iframe

src="https://fileurl.com/clean-code.pdf"

width="100%"

height="600px"

></iframe>

PDF file’s path:

<iframe

src="path/to/Clean-Code.pdf"

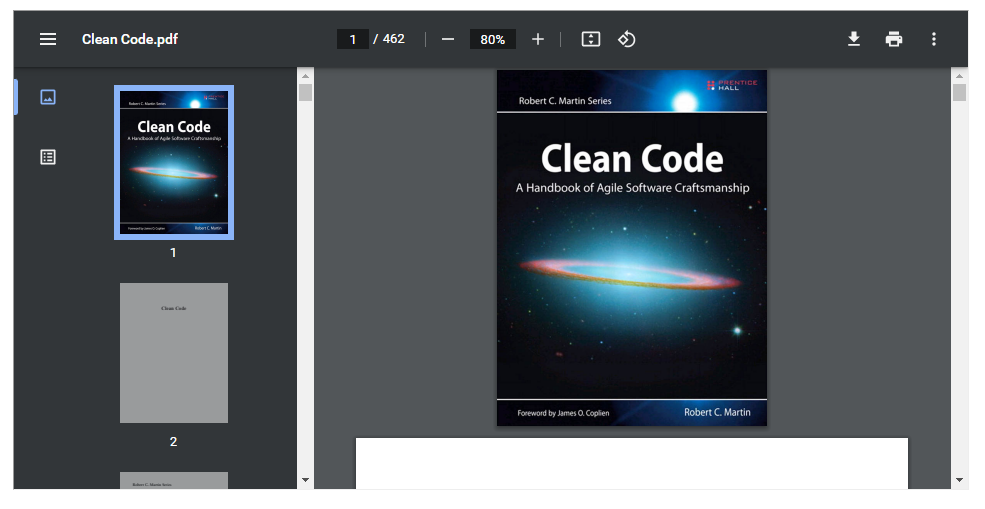
width="100%"

height="600px"

></iframe>

* **Width and Height**: We use these attributes to set the sizes of the embedded PDF file within the HTML page. The unit of the dimensions can either be in pixels or percentages.
* **Title**: The title attribute improves accessibility. It helps users using assistive technologies like [Screen Readers](https://abilitynet.org.uk/factsheets/introduction-screen-readers) understand the content better.
* **Loading**: This specifies how the iframe should load in the browser.

The <iframe> tag is a block-level element that takes up the entire width of the parent container. Here is the result of our embedded PDF file:



By default, the <iframe> tag has a border around it, which we can remove or style further with the CSS border [property](https://www.w3schools.com/css/css_border.asp). We can also use CSS to style the content within the <iframe> differently from the rest of the document. For interactions with the main HTML document and other <iframe> tags, we use JavaScript.

**Advantages of using the <iframe> tag**

Using the <iframe> tag has the following benefits:

* It is easy to use.
* It [supports](https://caniuse.com/?search=%3Ciframe%3E) all modern browsers like Google Chrome.
* Search engines can index PDF files embedded with the <iframe> tag. As a result, users can find and access the content more easily.

**Disadvantages of using the <iframe> tag**

Using the <iframe> tag has the following disadvantages:

* Older web browsers like Internet Explorer have [limited support](https://caniuse.com/?search=%3Ciframe%3E).
* Embedding PDF files from untrusted sources can expose the website to [security vulnerabilities](https://blog.bitsrc.io/4-security-concerns-with-iframes-every-web-developer-should-know-24c73e6a33e4).

**Method 2: Using <object> tag**

The <object> tag is like the <iframe>, as we use both tags to embed external content in an HTML document. We use the <object> tag to embed a wide range of external content like multimedia, Java Applets, etc.

Example of using the <object> tag to embed a PDF file:

<object

data="path/to/Clean-Code.pdf"

type="application/pdf"

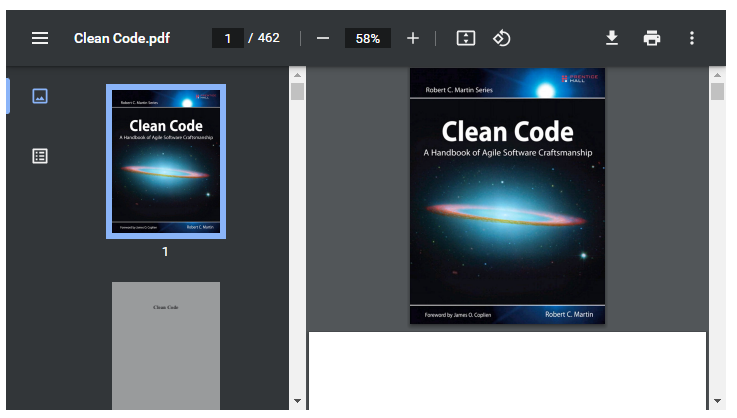
width="100%"

height="600px"

></object>

In the example above:

* **Data**: The data attribute contains the path to the PDF file we want to embed.
* **Type**: The type attribute identifies the file or media we are about to embed.
* **Width and height**: We use these attributes to specify the size of the embedded files.

Here is the outcome of our embedded PDF file: 

**Advantages of using the <object> tag**

Using the <object> tag has the following benefits:

* It is easy to use.
* It is easy to customize.

**Disadvantages of using the <object> tag**

Using the <object> tag has the following disadvantages:

* Embedding a PDF file with the <object> tag can pose a security risk if the file contains malicious content.
* Different [devices](https://caniuse.com/?search=%3Cobject%3E) may display the embedded PDF files differently.

**Method 3: Using <embed> tag**

HTML <embed> tags allow us to embed external content directly into our HTML documents. We mainly use the <embed> tag to embed external multimedia files such as audio, video, PDF files, and much more. It is a self-closing tag, meaning it doesn’t have a closing tag like the <iframe> and <object> elements.

Example of using the <embed> tag to embed a PDF file:

<embed

src="CleanCode.pdf"

type="application/pdf"

width="100%"

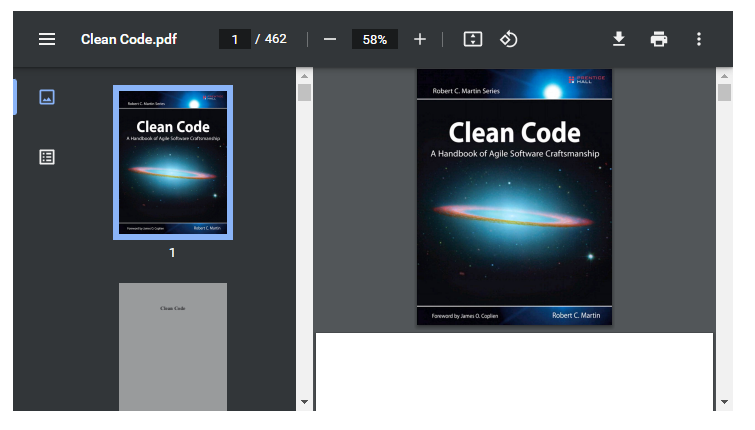
height="600px"

/>

The following attributes of the <embed> tag describe its requirements:

* **Src**: The src attribute contains the path to the PDF file we want to embed.
* **Type**: The type attribute identifies the file we are about to embed.
* **Width and Height**: We use these attributes to specify the size of the embedded files.

**Note**: When we run our HTML, the embedded PDF file’s appearance may vary based on the browser or mobile device.

Here is the outcome of our embedded PDF file: 

**Advantages of using the <embed> tag**

* It is easy to use.
* It allows us to embed multimedia content like audio and video files.

**Disadvantages of using the <embed> tag**

* It does not [support all browsers](https://caniuse.com/?search=%3Cembed%3E), especially older browsers.
* It does not have a fallback option.
* It has limited controls, and we cannot control embedded media like audio and videos. It is better to use the <video>, <img>, and <audio> HTML tags for videos, images, and audio files.

**Session Replay for Developers**

Uncover frustrations, understand bugs and fix slowdowns like never before with [OpenReplay](https://github.com/openreplay/openreplay) — an open-source session replay tool for developers. Self-host it in minutes, and have complete control over your customer data.

[Check our GitHub repo](https://github.com/openreplay/openreplay) and join the thousands of developers in our community.

**Fallback content**

When a browser does not support the feature, we use Fallback Content as a backup to the main content. So we use the Fallback Content for browsers that don’t support <iframe>, <object>, and <embed> tags. It can be anything from a simple blog post to a video, audio, or PDF file.

The <iframe> tag does not support [Fallback Content](https://www.w3.org/TR/2010/WD-html5-20101019/the-iframe-element.html#:~:text=An%20iframe%20element%20never%20has,of%20iframe%20elements%20represent%20nothing.). If the content within the <iframe> tag fails to load, there is no way to display alternative content.

The <object> tag supports [Fallback Content](https://www.w3.org/TR/WCAG20-TECHS/H53.html#:~:text=Fallback%20content%20for%20the%20object,not%20to%20render%20that%20technology.). Here is a code example of how to provide fallback content:

<object

data="path/to/Clean-Code.pdf"

type="application/pdf"

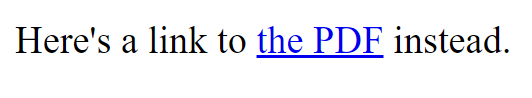
width="100%"

height="600px"

>

<p>Here's a link to <a href="CleanCode.pdf">the PDF</a>instead.</p>

</object>

The content between the opening and closing <object> tag will display as Fallback Content for browsers that do not support the <object> tag. Here is an example of how it will display on the browser: 

The <embed> element does not support [Fallback Content](https://www.w3.org/TR/2010/WD-html5-20101019/the-iframe-element.html#:~:text=The%20embed%20element%20has%20no%20fallback%20content.), just like the <iframe>. It will be blank if the content within the <embed> tag fails to load.

**Best practices for embedding PDFs in HTML**

* We must test the embedded PDF file in several browsers to ensure it will function and appear as expected. Testing the embedded PDFs will provide an optimal user experience for all web users.
* Use small PDF files instead of large ones. Large PDF files make our website’s page load slowly. Also, it increases the amount of [bandwidth](https://support.yondo.com/hc/en-us/articles/234706707-What-is-Bandwidth-) required to load our web page. It will impact users with low data plans or web hosts with limited bandwidth allotments. Before embedding our PDF files in HTML web pages, it is a great practice to compress them as much as possible. Doing this will guarantee that they load quickly on any device or browser.
* Always include a Fallback option in case of any glitches with the embedded PDF file.

**Conclusion**

Embedding PDFs in HTML is a great way to make our website more interactive and engaging for our visitors. It provides a smooth experience for users who can view the PDF content without leaving the web page. Following the steps outlined in this article, we can easily embed PDF files into our website.

**Resources**

We will find the following resources helpful:

* [Why You Should Embed PDFs in HTML for a Better Viewing Experience](https://www.accusoft.com/resources/blog/embed-pdf-in-html/).
* [How to Use PDF.js in 2 Easy Steps](https://pdfjs.express/blog/how-to-use-pdf-js).

**More articles from OpenReplay Blog**

[Aug 11, 2023, 6 min read](https://blog.openreplay.com/a-guide-to-2d-and-3d-transforms/)

**[A Guide to 2D and 3D transforms](https://blog.openreplay.com/a-guide-to-2d-and-3d-transforms/)**

[Apr 3, 2023, 4 min read](https://blog.openreplay.com/a-guide-to-http-post-requests-in-javascript/)

**[A Guide to HTTP POST requests in JavaScript](https://blog.openreplay.com/a-guide-to-http-post-requests-in-javascript/)**

OpenReplay relies on [cookies](https://openreplay.com/privacy.html) to make its website easier to use.

© 2023 [OpenReplay](https://openreplay.com) Blog