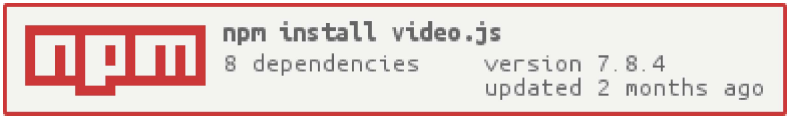
Video.js - HTML5 Video Player



> Video.js is a web video player built from the ground up for an HTML5 world. It supports HTML5 and Flash video, as well as YouTube and Vimeo (through plugins). It supports video playback on desktops and mobile devices. This project was started mid 2010, and the player is now used on over ~~50,000~~ ~~100,000200,000~~ 400,000 websites.

Table of Contents

Quick Start

Thanks to the awesome folks over at Fastly, there's a free, CDN hosted version of Video.js that anyone can use. Add these tags to your document's `<head>`:

```
<link href="//vjs.zencdn.net/7.8.2/video-js.min.css" rel="sty
<script src="//vjs.zencdn.net/7.8.2/video.min.js"></script>
```

> For the latest version of video.js and URLs to use, check out the Getting Started page on our website.

Video.js version 7 (and newer) CDN builds do not send any data to Google Analytics.

In older versions of Video.js (6 and earlier), in the `vjs.zencdn.net` CDN-hosted versions we include a stripped down Google Analytics pixel that tracks a random sampling (currently 1%) of players loaded from the CDN. This allows us to see (roughly) what browsers are in use in the wild, along with other useful metrics such as OS and device. If you'd like to disable analytics, you can simply include the following global before including Video.js via the free CDN:

```
<script>window.HELP_IMPROVE_VIDEOJS = false;</script>
```

Alternatively, you can include Video.js by getting it from npm, downloading from GitHub releases or by including it via unpkg or another JavaScript CDN like CDNjs. These releases *do not* include Google Analytics tracking at all.

```
<!-- unpkg : use the latest version of Video.js -->
<link href="https://unpkg.com/video.js/dist/video-js.min.css'
<script src="https://unpkg.com/video.js/dist/video.min.js"></

<!-- unpkg : use a specific version of Video.js (change the v
<link href="https://unpkg.com/video.js@7.8.2/dist/video-js.mi
<script src="https://unpkg.com/video.js@7.8.2/dist/video.min.

<!-- cdnjs : use a specific version of Video.js (change the v
<link href="https://cdnjs.cloudflare.com/ajax/libs/video.js/7
<script src="https://cdnjs.cloudflare.com/ajax/libs/video.js/
```

Next, using Video.js is as simple as creating a `<video>` element, but with an additional `data-setup` attribute. At a minimum, this attribute must have a value of `'{}'`, but it can include any Video.js options - just make sure it contains valid JSON!

```
<video
    id="my-player"
    class="video-js"
    controls
    preload="auto"
    poster="//vjs.zencdn.net/v/oceans.png"
    data-setup='{}'>
  <source src="//vjs.zencdn.net/v/oceans.mp4" type="video/mp4
  <source src="//vjs.zencdn.net/v/oceans.webm" type="video/we
  <source src="//vjs.zencdn.net/v/oceans.ogv" type="video/ogg
  <p class="vjs-no-js">
    To view this video please enable JavaScript, and consider
    web browser that
    <a href="https://videojs.com/html5-video-support/" target
      supports HTML5 video
    </a>
  </p>
</video>
```

When the page loads, Video.js will find this element and automatically setup a player in its place.

If you don't want to use automatic setup, you can leave off the `data-setup` attribute and initialize a `<video>` element manually using the `videojs` function:

```
var player = videojs('my-player');
```

The `videojs` function also accepts an `options` object and a callback to be invoked when the player is ready:

```
var options = {};

var player = videojs('my-player', options, function onPlayerR
  videojs.log('Your player is ready!');

  // In this context, `this` is the player that was created b
  this.play();

  // How about an event listener?
  this.on('ended', function() {
    videojs.log('Awww...over so soon?!');
  });
});
```

If you're ready to dive in, the Getting Started page and documentation are the best places to go for more information. If you get stuck, head over to our Slack channel!