

webpack v4.33.0

EDIT DOCUMENT PRINT DOCUMENT  Print Section Concepts printable.md

- ... Concepts
- ... Entry
- ... Output
- ... Loaders
- ... Plugins
- ... Mode
- ... Browser Compatibility
- ... Entry Points
- ... Single Entry (Shorthand) Sy...
- ... Object Syntax
- ... Scenarios
- ... Separate App and Vendor En...
- ... Multi Page Application
- ... Output
- ... Usage
- ... Multiple Entry Points
- ... Advanced
- ... Loaders
- ... Example
- ... Using Loaders
- ... Configuration
- ... Inline
- ... CLI
- ... Loader Features
- ... Resolving Loaders
- ... Plugins
- ... Anatomy
- ... Usage
- ... Configuration
- ... Node API
- ... Configuration

Concepts

At its core, **webpack** is a *static module bundler* for modern JavaScript applications. When webpack processes your application, it internally builds a [dependency graph](#) which maps every module your project needs and generates one or more *bundles*.

Learn more about JavaScript modules and webpack modules [here](#).

Since version 4.0.0, **webpack does not require a configuration file** to bundle your project, nevertheless it is [incredibly configurable](#) to better fit your needs.

To get started you only need to understand its **Core Concepts**:

- [Entry](#)
- [Output](#)
- [Loaders](#)
- [Plugins](#)
- [Mode](#)
- [Browser Compatibility](#)

This document is intended to give a **high-level** overview of these concepts, while providing links to detailed concept-specific use cases.

For a better understanding of the ideas behind module bundlers and how they work under the hood, consult these resources:

- [Manually Bundling an Application](#)
- [Live Coding a Simple Module Bundler](#)
- [Detailed Explanation of a Simple Module Bundler](#)