

# 面试题

## 1. 什么是 Webpack

1. webpack是一个javascript的静态模块打包工具
2. webpack里一切文件皆模块，通过loader转换文件，通过plugin注入钩子
3. 最后输出由多个模块组合成的文件，webpack专注构建模块化项目

## 2. Webpack 的优点是什么

1. 专注于处理模块化的项目，能做到开箱即用，一步到位
2. 通过plugin扩展，完整好用又不失灵活
3. 通过loaders扩展，可以让webpack把所有类型的文件都解析打包
4. 社区庞大活跃，经常引入紧跟时代发展的新特性，能为大多数场景找到已有的开源扩展

## 3. Webpack 的构建流程是什么

webpack 的运行流程是一个串行的过程，从启动到结束会依次执行以下流程：

1. 初始化参数：从配置文件读取与合并参数，得出最终的参数
2. 开始编译：用上一步得到的参数初始化 Compiler 对象，加载所有配置的插件，开始执行编译
3. 确定入口：根据配置中的 entry 找出所有的入口文件
4. 编译模块：从入口文件出发，调用所有配置的 Loader 对模块进行翻译，再找出该模块依赖的模块，再递归本步骤直到所有入口依赖的文件都经过了本步骤的处理
5. 完成模块编译：在经过第4步使用 Loader 翻译完所有模块后，得到了每个模块被翻译后的最终内容以及它们之间的依赖关系
6. 输出资源：根据入口和模块之间的依赖关系，组装成一个个包含多个模块的 Chunk，再把每个 Chunk 转换成一个单独的文件加入到输出列表，这步是可以修改输出内容的最后机会
7. 输出完成：在确定好输出内容后，根据配置确定输出的路径和文件名，把文件内容写入到文件系统。

在以上过程中，webpack 会在特定的时间点广播出特定的事件，插件在监听到感兴趣的事件后会执行特定的逻辑，并且插件可以调用 webpack 提供的 API 改变 webpack 的运行结果

## 4. 说一下 Webpack 的热更新原理

webpack 的热更新又称热替换（Hot Module Replacement），缩写为 HMR。这个机制可以做到不用刷新浏览器而将新变更的模块替换掉旧的模块。

HMR的核心就是客户端从服务端拉去更新后的文件，准确的说是 chunk diff (chunk 需要更新的部分)，实际上 WDS(webpack-dev-server) 与浏览器之间维护了一个 Websocket，当本地资源发生变化时，WDS 会向浏览器推送更新，并带上构建时的 hash，让客户端与上一次资源进行对比。客户端对比出差异后会向 WDS 发请求来获取更改内容(文件列表、hash)，这样客户端就可以再借助这些信息继续向 WDS 发起 jsonp 请求获取该chunk的增量更新。

后续的部分(拿到增量更新之后如何处理？哪些状态该保留？哪些又需要更新？)由 HotModulePlugin 来完成，提供了相关 API 以供开发者针对自身场景进行处理，像react-hot-loader 和 vue-loader 都是借助这些 API 实现 HMR。

## 5. 有哪些常见的 loader

1. babel-loader: 把 ES6 转换成 ES5
2. css-loader: 加载 CSS, 支持模块化、压缩、文件导入等特性
3. style-loader: 把 CSS 代码注入到 JavaScript 中, 通过 DOM 操作去加载 CSS

## 6. loader 和 plugin 的区别

Loader直译为"加载器"。webpack将一切文件视为模块,但是webpack原生是只能解析JS和JSON 代码内容,如果想将其他文件也打包的话,就会用到loader。所以loader的作用是让webpack拥有了加载和解析非JavaScript代码功能。

Plugin直译为"插件"。Plugin可以扩展webpack的功能,让webpack具有更多的灵活性。在 webpack 运行的生命周期中会广播出许多事件,Plugin 可以监听这些事件,在合适的时机通过 webpack 提供的 API 改变输出结果。

Loader在module.rules中配置,也就是说他作为模块的解析规则而存在。类型为数组,每一项都是一个Object,里面描述了对于什么类型的文件(test),使用什么加载(loader)和使用的参数(options)

plugin在plugins中单独配置。类型为数组,每一项是一个plugin的实例,参数都通过构造函数传入。