

2020-6-4 webpack (一) 培训

一. Webpack 简介 - (01webpack简介)

1. webpack是一种前端资源构建工具
 - a. Eg: less 文件不生效, 将less编译成css 文件 (借助需要一个工具)
 - b. Eg: ES6 语法, 浏览器不认识, 怎么办? (需要一个工具 编译成浏览器能识别的代码)
 - c. 所以就出现了“构建工具” - webpack是其中一种
2. webpack也是一个静态模块打包器 (module bundle)
 - a. // 引入js 资源
 - b. // 引入css 等其他资源
 - c. index.js 作为打包的一个入口文件, 为起点, 开始打包
 - d. 里面所有的依赖关系 (js, css, img 等其他资源) 都会形成一个依赖关系关系图
 - e. Index.js 依赖 css, jq, 形成一个代码块 (**chunk** 块), 然后 依次打包, 输出, 输出的东西就叫 **bundle**
 - f. Js, jq, css 等这些可以统称为模块
3. 在webpack看来, 前端所有的资源文件(js/json/css/img/less/...)都会作为模块处理, 它将根据模块的依赖关系进行静态分析, 打包生成对应的静态资源 (bundle)

二. Webpack 五个核心概念

1. Entry
 - a. 入口 (Entry) 指示webpack以哪个文件为入口起点开始打包, 分析构建内部依赖图
2. Output
 - a. 输出 (Output) 指示webpack打包后的资源bundles输出到哪里去, 以及如何命名
3. Loader (翻译官)
 - a. Loader 让webpack能够去处理那些非JS(css/img/...)文件(webpack 自身只理解javascript)
4. Plugins (开飞机, 开大炮类似这些插件去做)
 - a. 插件(Plugins)可以用于执行范围更广的任务。插件的范围包括, 从打包优化和压缩, 一直到重新定义环境中的变量等
5. Mode
 - a. development & production

三. Webpack 初体验 (实际演练, 操作)- (02webpack初体验)

1. yarn add webpack webpack-cli -g (全局安装webpack 和 webpack-cli)
2. webpack自身原本的功能
3. webpack ./src/index.js -o ./build/built.js --mode=development
4. webpack ./src/index.js -o ./build/built.js --mode=production
5. Webpack 默认只能处理js和json文件, 不能处理css, img资源, 做验证
6. 生产环境比开发环境多一个压缩js代码功能
7. 生产环境和开发环境将ES6 模块化编译成浏览器能识别的模块化

四. Webpack打包样式资源 - (03打包样式资源)

1. webpack.config.js (webpack的配置文件)
 - a. 作用: 指示webpack干哪些活 (当你运行webpack指令的时候, 会加载里面的配置)

- b. 所有构建工具都是基于node js平台运行的, 模块化默认采用commonJS (module.exports = {})
 - c. SRC 里面的源代码我们是基于ES6 模块话来编写的
- 2. Loader执行顺序从右往左, 或者从下往上
- 3. Loader先安装, 再使用, 不需要引入
- 4. CSS文件 (style-loader, css-loader)
 - a. css-loader 作用: 将CSS文件作为commonJS模块加载到js文件中,里面内容是样式字符串
 - b. style-loader 作用: 动态创建一个style 标签, 将加载到js中的css字符串插入到style标签中, 页面就生效了。
- 5. LESS文件 (style-loader, css-loader, less-loader)
 - a. Less-loader 作用: 先将 less文件转化成 css 文件
 - b. 其它步骤同打包CSS文件一样
- 6. ...
- 7.

五. Webpack打包HTML文件 - (04打包HTML资源)

- 1. 使用Plugin 来打包HTML文件
- 2. html-webpack-plugin
 - a. 默认会创建一个空的html文件, 没有任何结构和样式
 - b. 默认会引入打包输出的所有资源, 然后输出
- 3. 需要有结构的HTML文件
 - a. template(复制指定的html文件, 并自动引入打包输出的所有资源)

六. Webpack打包图片资源 - (05打包图片资源)

- 1. 第一种是在样式中引用图片资源
 - a. Url-loader & file-loader
- 2. 第二种是在html中引用图片资源
 - a. Html-loader

七. Webpack打包其它资源 - (06打包其它资源)

- 1. 用阿里图标做个demo.
- 2. exclude属性: (排除指定资源文件)
- 3. file-loader即可

八. DevServer - (07devServer)

- 1. 自动打包, 不需要每次都重新运行webpack
- 2. 开发服务器: devServer:用来自动化 (自动编译, 自动打开浏览器, 自动刷新浏览器)
- 3. webpack-dev-server 特点:
 - a. 只会在内存中编译打包, 不会有任何输出

后续:

- 1. 完整的开发环境基本配置
- 2. 输出文件按目录区分

3. 把css文件单独提取出来
4. css兼容性处理
5. 压缩css
6. js语法检查, eslint
7. js兼容性处理
8. ...