**1、**webpack --mode development

**2、**Webpack -w 监听打包（配置文件改变时应退出监听，重新打包）

**3、**Output:{filename:index-[hash：8].js} 主要有文件改变，所有文件都打包

**4、**Output:{filename:index-[chunkhash:8].js}打包只改变的文件

**5**、配置loader：module:{rules:[{test:/.js$/,use:[‘babel-loader’]}]}

**6、**ES6 js加载器 babel-loader,babel-loader 依赖于 babel、 babel-core

**7、**css加载器 style-loader、css-loader(由后向前解析)

**8、**图片 加载器 url-loader

**9、**图片打包限制：use:[‘url-loader?limit=8192&name=/[name].[ext]’](需要file-loader)

**10、**publicPath 是像图片这种静态资源打包后的根路径

11、多个入口文件，输出的文件名应‘[name].js’根据入口文件名

12、压缩插件：uglifyjs-webpack-plugin(需要单独下载对应的版本的webpack)

13、plugins:[new uglifyjs()] uglifyjs=requires(‘uglifyjs-webpack-plugin’)

14、--save-dev 开发时依赖比如压缩插件，上线后不需要

15、plugins:[new webpack.optimize.CommonChunkPlugin( （独立打包插件）

{name:”commons”,

filename:”common.js”,

minChunks:3}//公共模块被引用3次就会被打包

)]

1. css独立打包require(extract-text-webpack-plugin) --save-dev(不支持webpack4.o以后版本需要install npm install --save-dev extract-text-webpack-plugin@next)
2. Css 独立打包： {test:/\.css$/,use:ExtractTextPlugin.extract(  
    {fallback:”style-loader”,

use:”css-loader”})}

1. css独立打包 new ExtractTextPlugin(“[name].css”)
2. 独立打包css时会发生于图片打包地址不统一情况，publicPath：需要配置服务器地址
3. 在package.json中找scripts属性对应的对象添加属性：用来替代dos命令（使用时用npm run 对应属性名）
4. Less加载器 需要less-loader less
5. less引入样式文件@import(“url”)
6. 压缩代码插件：UglifyjsPlugin=require("uglifyjs-webpack-plugin");
7. plugins:[new UglifyjsPlugin()]
8. CommonsChunkPlugins 打包公共模块的插件（require(“webpack”)
9. 添加CommonsChunkPlugins插件

new webpack.optimize.CommonsChunkPlugin(

{name:“commons”,filename:”commons.js”,minChunks:3})(3代表工作模块被引用3次时独立打包)

1. css独立打包：require(“extract-text-webpack-plugins”);  
    new cssPlugins(“[name].css”)
2. Css独立打包rules配置：test:/.css/,use:ExtractTextPlugin.extract({fallback:”style-loader”,use:”css-loader”})
3. Css独立打包时会与require引入的图片地址会出现问题，需要将publicPath路径改为服务器地址：
4. Webpack工具：
5. devServer 配置： contentBase： 打开浏览器进去的路径 Port端口号
6. Webpack-dev-server --hot 模块热替换，只改变改变的文件