Webpack安装和基本工作流

好处：将js服务端的写法、es6的写法等编译打包，让浏览器识别，按需加载资源

全局安装

Npm install webpack -g

简单使用

Webpack 入口文件路径 输出文件路径。例如webpack js/entry.js bundle.js

使用配置文件

项目根路径创建webpack.config.js

内容

module.exports = {

devtool: "sourcemap",//打包工具，会生成一个map映射文件

entry: "./js/entry.js",//入口文件

output: {//输出文件

filename:"bundle.js"

}

};

然后使用命令webpack就可以直接打包了

按需加载例子，加载使用jq

Npm init 一直回车生成package.json配置文件

npm install jquery --save-dev 安装jq

在module-one.js写入

var $ = require('jquery');

$("h1").html('it is true');

Webpack 提供了 loaders 的机制来实现对不同类型的静态文件进行打包，这样不仅可以减少 http 请求，也对文件压缩进行了优化。

npm 安装相关 loader：

Npm install css-loader style-loader babel-core babel-loader babel-plugin-transform-runtime babel-preset-es2015 babel-preset-stage-0 babel-runtime --save-dev

修改配置文件

module: {

loaders: [

{

test: /\.css$/,//正则表达式，如果匹配到后缀为.css的文件使用style或者css加载器

loader: "style-loader!css-loader"

},

{

test: /\.js$/,

loader: "babel-loader", // Do not use "use" here

exclude: /node\_modules/,

options: {

presets: ['es2015','stage-0'],

plugins: ['transform-runtime']

}

}

]

}

或者inline加载

require('!style-loader!css-loader!../css/style.css');//不推荐

修改module-one.js

import $ from "jquery";

$("h1").html('it is true use babel-loader');

webpack打包搞定

使用webpack进行vue组件化开发

安装的 vuejs 相关 package：

npm install vue vue-loader vue-html-loader vue-style-loader vue-template-compiler --save-dev

2、然后，在webpack-config.js里修改成：

loaders: [

{

test: /\.vue$/,

loader: 'vue-loader'

}

3、创建vue组件 js/components/heading.vue

<template>  
 <div>  
 <h1>{{ message }}</h1>  
 </div>  
</template>  
  
<script>  
 export default {  
 data() {  
 return {  
 message: 'Welcome to Your Vue.js App'  
 }  
 }  
 }  
</script>  
  
<!-- Add "scoped" attribute to limit CSS to this component only -->  
<style scoped>  
  
</style>

4、修改entry.js

import Vue from 'vue';  
  
// Vue.component('Heading',require('./components/heading.vue'));  
// new Vue({  
// el:"#app"  
// });  
//  
import Heading from "./components/heading.vue";  
new Vue({  
 el:"#app",  
 components:{Heading}  
});

修改index.html

<div *id=*"app">  
 <Heading></Heading>  
</div>

webpack搞定

四、Webpack 热加载和一些有用的命令行

1、webpack --display-modules

2、webpack –display-reasons

3、webpack –d –p debug和压缩文件

4、webpack –w 免编译

5、安装 hot reload （热加载）

Npm install webpack-dev-server –g

使用webpack-dev-server --inline –hot 只要文件内容变化自动编译

五、Webpack 插件和配置文件的最佳实践

1、将webpack配置文件生成json文件

webpack –profile –json > stats.json

将生成的stats.json上传至<http://webpack.github.io/analyse/可分析项目>

或者去http://chrisbateman.github.io/webpack-visualizer/