什么是WebPack，为什么要使用它？

1. 为什要使用WebPack

现今的很多网页其实可以看做是功能丰富的应用，它们拥有着复杂的JavaScript代码和一大堆依赖包。为了简化开发的复杂度，前端社区涌现出了很多好的实践方法。

模块化，让我们可以把复杂的程序细化为小的文件;

类似于TypeScript这种在JavaScript基础上拓展的开发语言：使我们能够实现目前版本的JavaScript不能直接使用的特性，并且之后还能转换为JavaScript文件使浏览器可以识别；

Scss，less等CSS预处理器

...

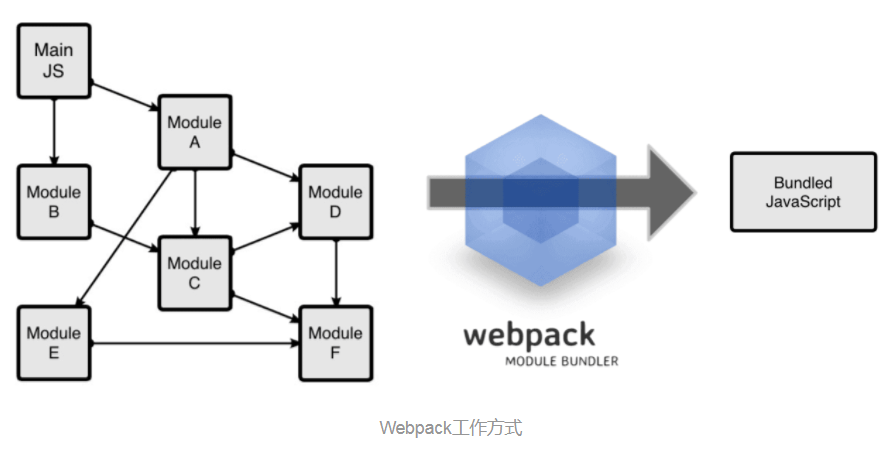
这些改进确实大大的提高了我们的开发效率，但是利用它们开发的文件往往需要进行额外的处理才能让浏览器识别,而手动处理又是非常繁琐的，这就为WebPack类的工具的出现提供了需求。

2.什么是Webpack

Webpack 是德国开发者 Tobias Koppers 开发的模块加载器兼打包工具，在webpack中，它能把各种资源，例如JS（含JSX）、coffee、样式（含less/sass）、图片等都作为模块来使用和处理。因此, Webpack 当中 js 可以引用 css, css 中可以嵌入图片 dataUrl。

对应各种不同文件类型的资源, Webpack 有对应的模块 loader比如vue用的是vue-loader当然这是后话，在后面我们再来说。

Webpack工作方式



1. 安装Webpack

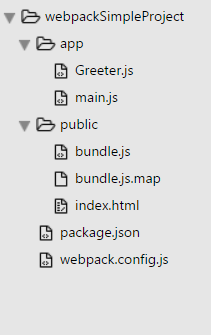
全局安装： npm install webpack -g;

安装到项目目录： npm install webpack --save-dev

1. 使用Webpack

一个简单的Webpack工程打包。

项目目录如下：



首先创建一个项目文件夹，然后使用npm init初始化npm项目，会自动生成package.json文件，然后在项目目录下创建app和public文件夹，app下存放项目源代码，public用于存放打包后的代码。

Greeter.js

var greet = document.createElement('div');

greet.textContent = "Hi there and greetings!";

export {greet}

main.js

import {greet} from './Greeter.js'

console.log(greet);

document.querySelector("#root").appendChild(greet);

Index.html

<!-- index.html -->

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<title>Webpack Sample Project</title>

</head>

<body>

<div id='root'>

</div>

<script src="bundle.js"></script>

</body>

</html>

#### 通过配置文件来使用Webpack

Webpack拥有很多其它的比较高级的功能（比如说本文后面会介绍的loaders和plugins），这些功能其实都可以通过命令行模式实现，这样不太方便且容易出错的，更好的办法是定义一个配置文件，这个配置文件其实也是一个简单的JavaScript模块，我们可以把所有的与打包相关的信息放在里面。

继续上面的例子来说明如何写这个配置文件，在当前练习文件夹的根目录下新建一个名为webpack.config.js的文件，我们在其中写入如下所示的简单配置代码，目前的配置主要涉及到的内容是入口文件路径和打包后文件的存放路径。

webpack.config.js

module.exports = {

entry: \_\_dirname + "/app/main.js",//已多次提及的唯一入口文件

output: {

path: \_\_dirname + "/public",//打包后的文件存放的地方

filename: "bundle.js"//打包后输出文件的文件名

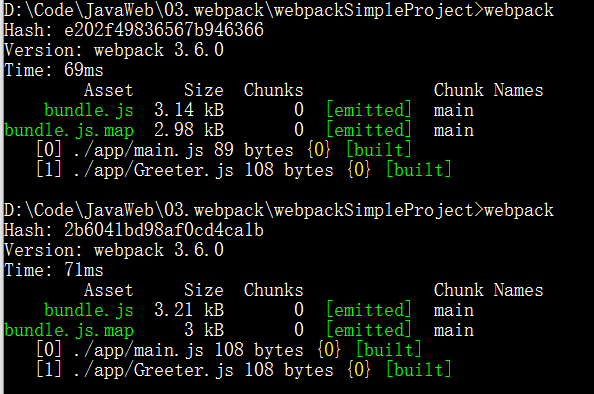
},

devtool: "source-map"

}

注：“\_\_dirname”是node.js中的一个全局变量，它指向当前执行脚本所在的目录。

有了webpack.config.js文件之后，只需在跟webpack打包配置文件同级目录下运行webpack命令，webpack就会自动读取改配置文件，根据配置完成打包。



至此，就完成了webpack的简单使用。