# Commonjs模块化

**commonJS的核心思想是通过require方法同步加载其依赖的其他模块，然后通过exports或者module.exports来导出需要暴露的接口**

**暴露接口方式一：**

**module.exports = function(){}**

**module.exports后可以是一个函数，也可以是一个对象。**

**暴露接口方式二：**

**exports.value = function(){} //value为一个变量名**

**exports.value后可以是一个函数，也可以是一个对象。(exports本质是一个空对象)**

**至于导出需要在入口文件中**

**导出方式：**

**var value = require(url); //value为一个变量名,url为导出模块的路径**

**在入口文件导出后使用：**

**当导入的模块使用module.exports后接的是一个函数时，使用方式为：**

**value();**

**当导入的模块使用module.exports后接的是一个对象时，使用方式为：**

**value.name(); //name为导入模块后接的对象中的命名**

**当导入的模块使用exports.value后接的是一个函数时，使用方式为：**

**value.name(); //name为导入模块中exports后接的value**

**当导入的模块使用exports.value后接的是一个对象时，使用方式为：**

**value.name.objName() //name为导入模块中exports后接的value,objName为导入模块后接的对象中的命名**

**接下来就是打包处理：**

**browserify main.js –o bundle.js //main.js为入口文件，bundle.js为输出文件**

**(此操作需要browserify工具支持，下载命令：npm install browserify -g)**

**引用方式：页面引入输出文件即可**

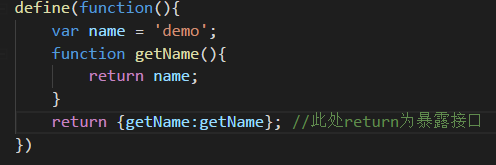
# AMD & require.js

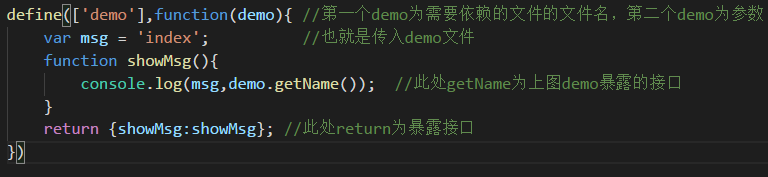
**AMD规范是异步加载模块，允许指定回调函数。等模块异步加载完成后即可调用回调函数。**

**AMD需要基于require.js**

**AMD是提前加载**

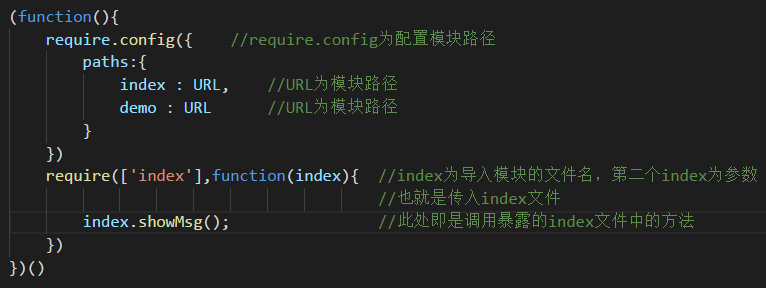
**定义模块方式：**





如没有所依赖的文件请参照图一

导出方式：



页面引入方式：



src为AMD所基于的require.js的路径 data-main为你导出模块的路径

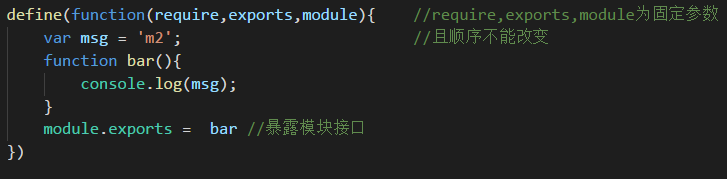
# CMD

**CMD异步加载，与AMD的主要区别在于AMD是提前加载依赖，而CMD是按需加载。**

**CMD需要基于sea.js**

**定义模块方式：**

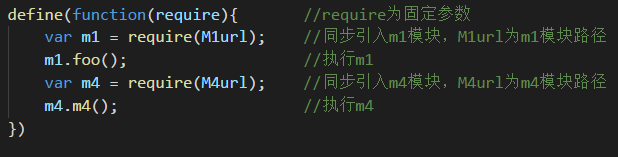
****

****

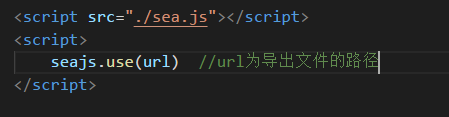
****

****

**导出方式：**

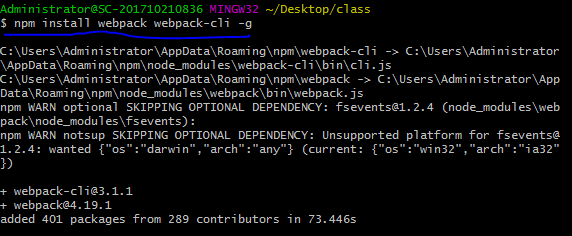
****

**页面引入方式：**

****

# **Webpack4**

安装webpack至全局（-g）：



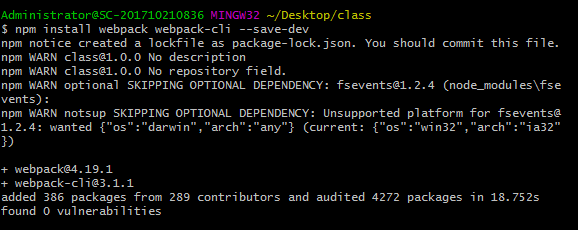
命令：npm install webpack webpack-cli –g //下载webpack以及其依赖补丁至全局

查看webpack版本：



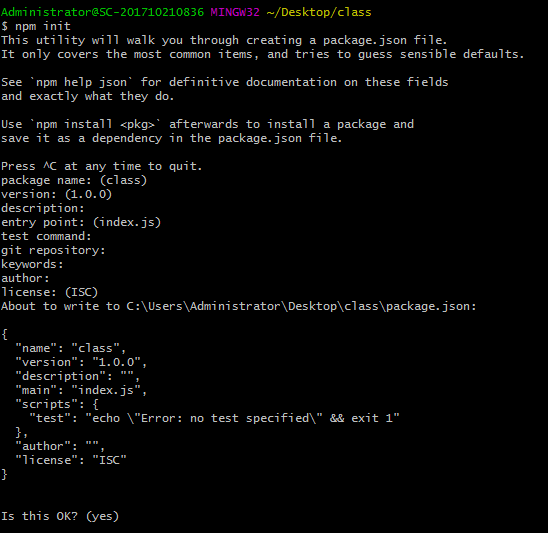
命令：webpack –v //查看webpack版本

安装webpack至工作目录（--save-dev或者-D）



命令：npm insrall webpack webpack-cli --save-dev //下载webpack以及其依赖补丁至当前工作目录

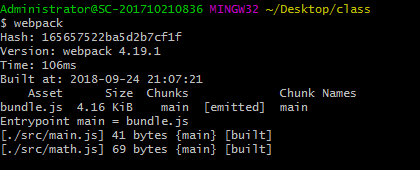
注意：执行此操作之前需要初始化工作目录



命令：npm init //初始化当前所在目录

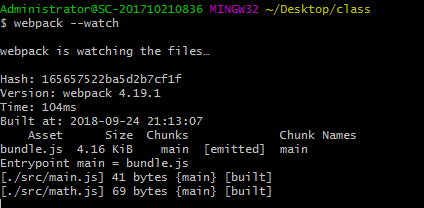
## 使用webpack：

执行打包：



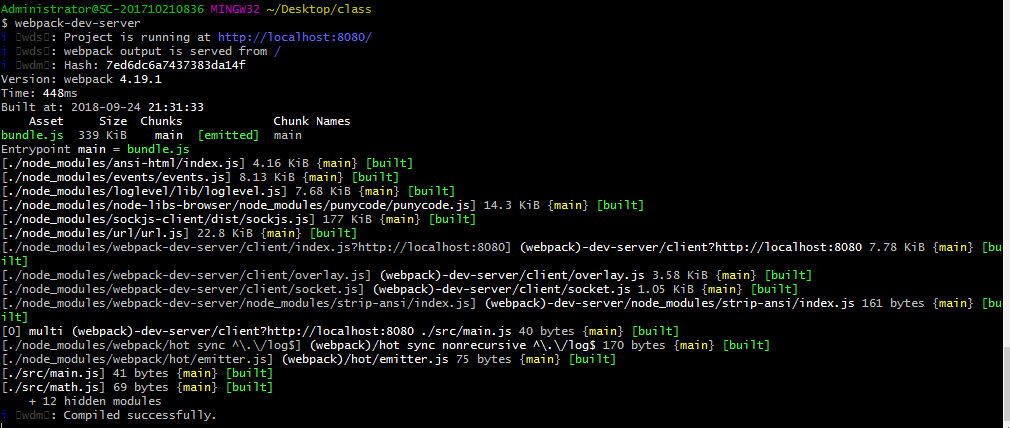
命令：webpack //开始打包文件

webpack监听：



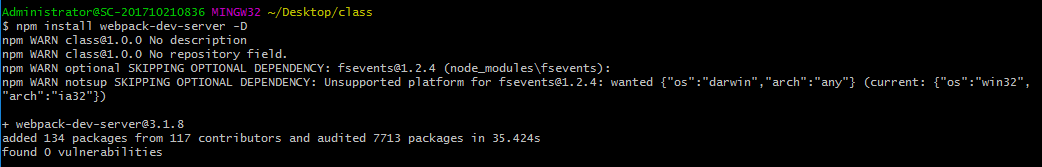
命令：webpack –watch //webpack监听

**Webpack开启服务器：**



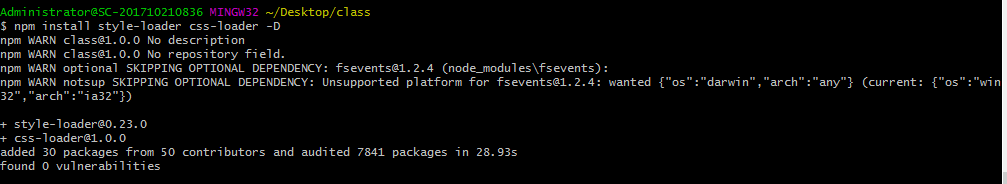
命令：webpack-dev-server //开启服务器

注意：执行此操作之前需要下载功能包



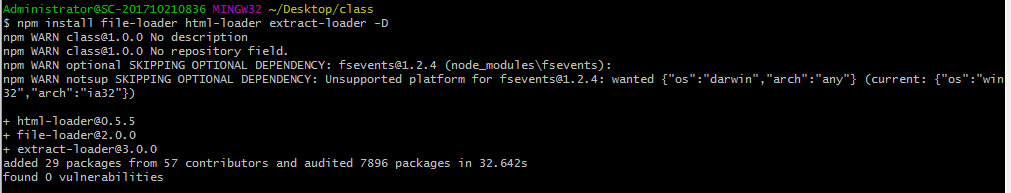
命令：npm install webpack-dev-server –g //下载功能包至全局

打包css文件需要下载style-loader和css-loader：



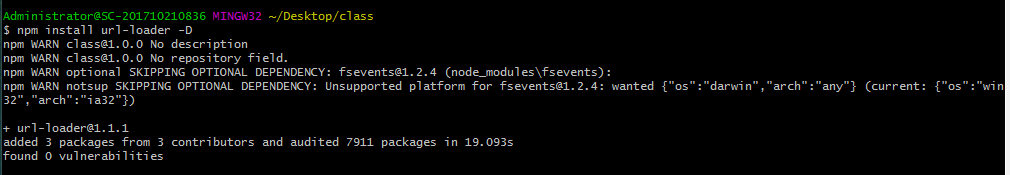
命令：npm install style-loader css-loader –D //下载打包css依赖的功能包

打包html文件需要下载file-loader，html-loader和extract-loader：



命令：npm install file-loader html-loader extract-loader –D //下载打包html依赖的功能包

打包图片文件需要下载url-loader：

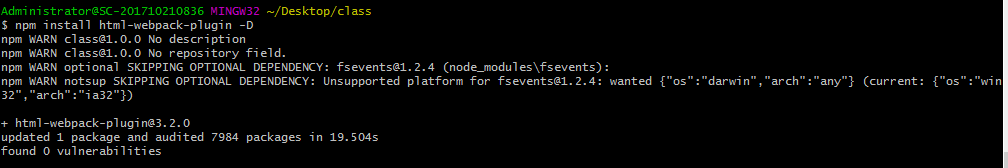


命令：npm install url-loader –D //下载打包图片的功能包

## 使用plugin插件

### 使用html-webpack-plugin插件

下载html-webpack-plugin插件：



命令：npm install html-webpack-plugin –D //下载html-webpack-plugin插件

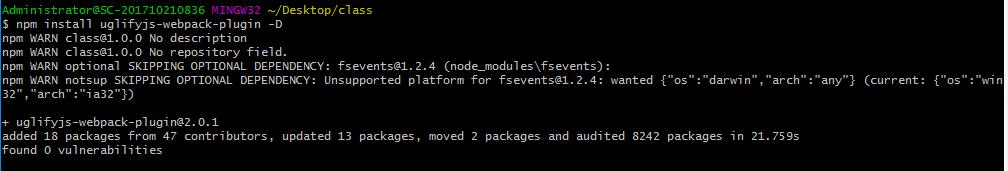
配置html-webpack-plugin插件：





### 使用uglify-webpack-plugin插件

下载uglify-webpack-plugin插件：



命令：npm install uglifyjs-webpack-plugin –D //下载uglifyjs-webpack-plugin插件

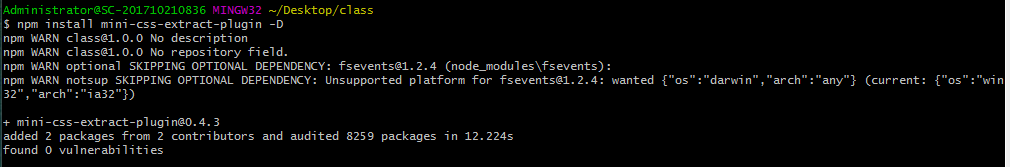
配置uglifyjs-webpack-plugin插件：





### 使用mini-css-extract-plugin插件

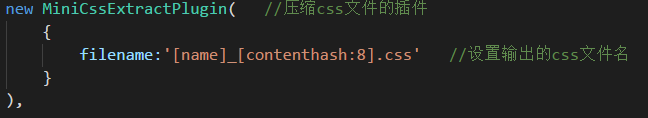
下载mini-css-extract-plugin插件：



命令：npm install mini-css-extract-plugin –D //下载mini-css-extract-plugin插件

配置mini-css-extract-plugin插件：





## Webpack配置文件代码及注释

