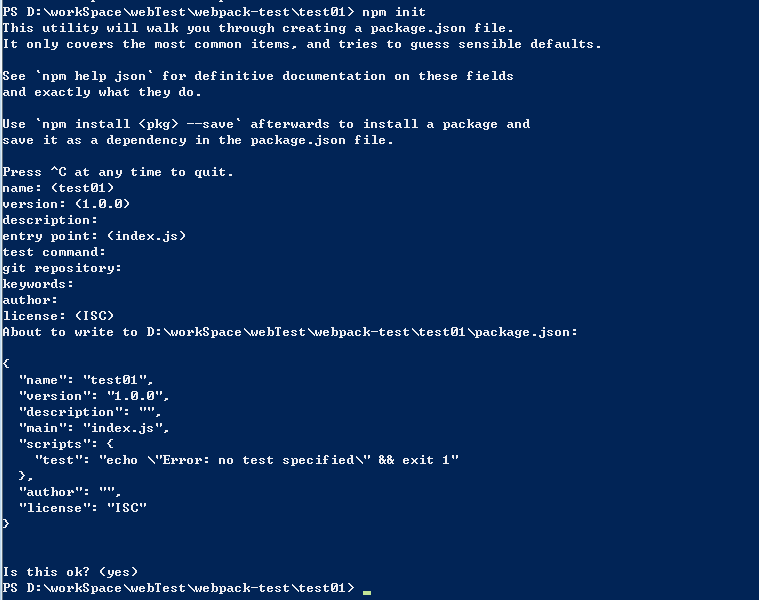
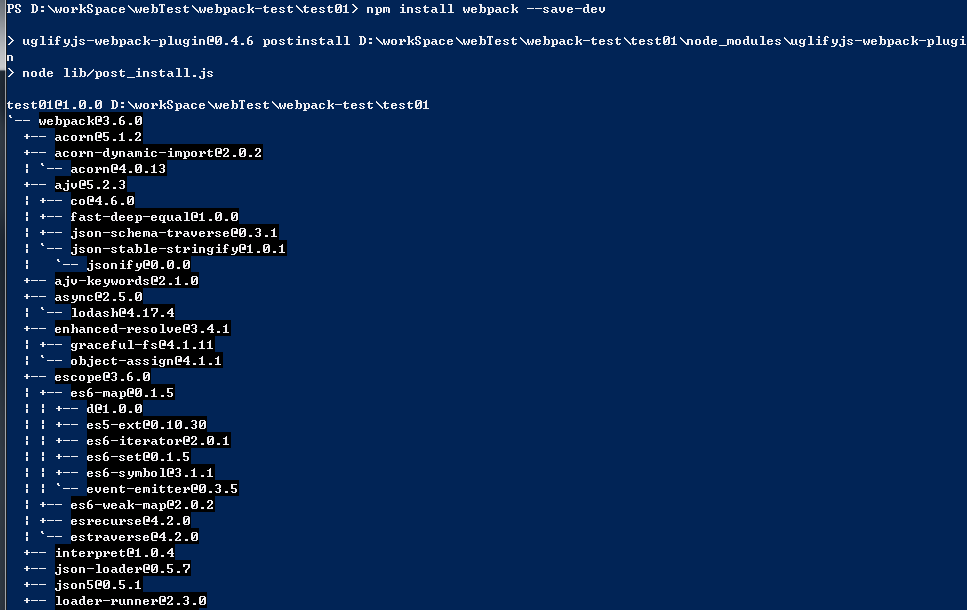
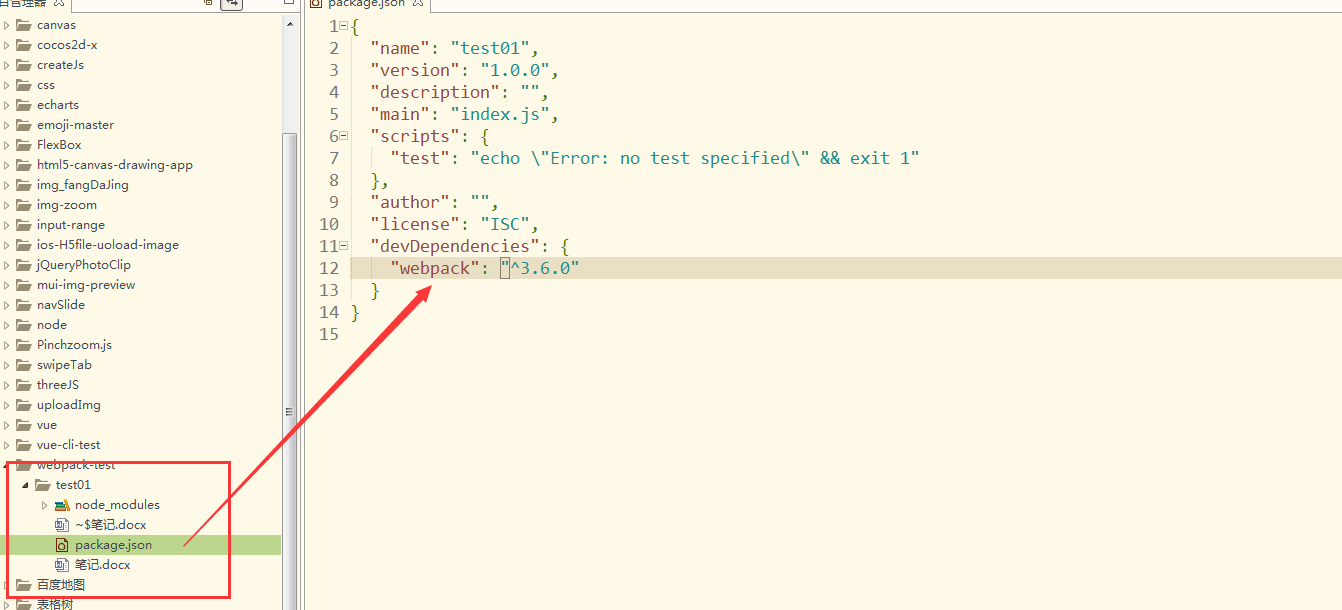
1. 初始化生成package.json ==》npm init



2、安装webpack模块 ==》npm install webpack --save-dev

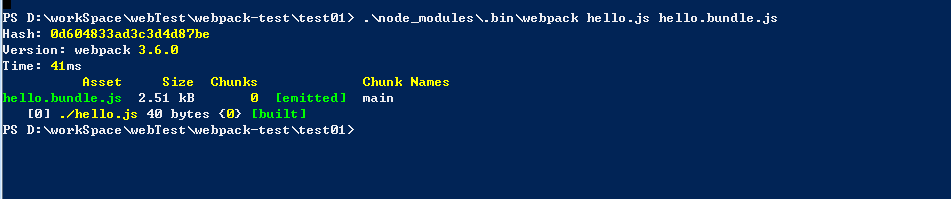


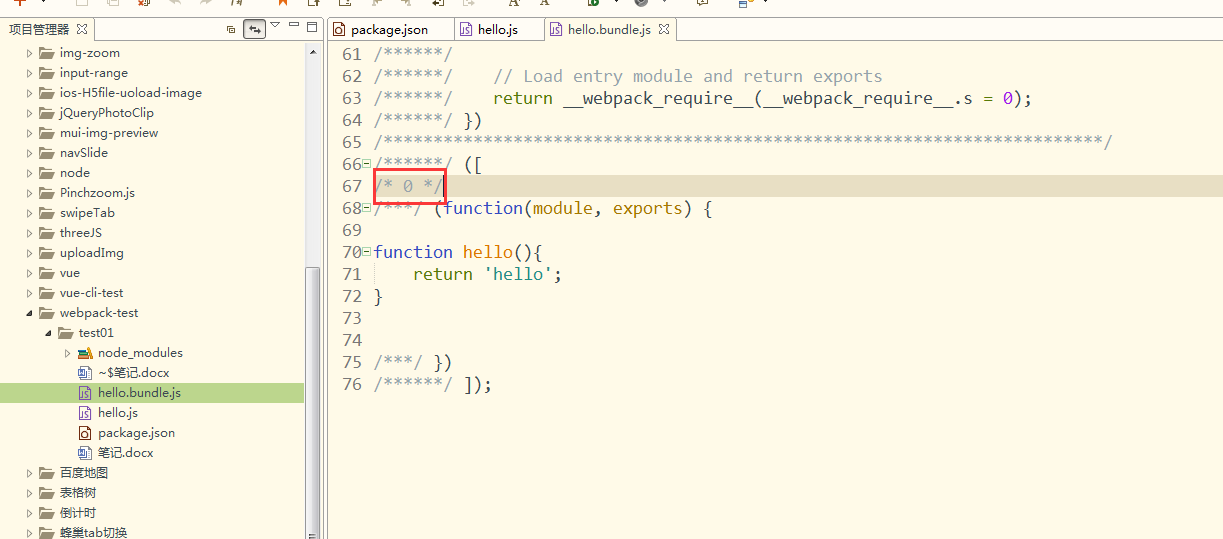


1. 新建文件，通过命令打包==》

全局安装了webpack==》webpack hello.js hello.bundled.js

本地安装的webpack==》.\node\_modules\.bin\webpack hello.js hello.bundle.js





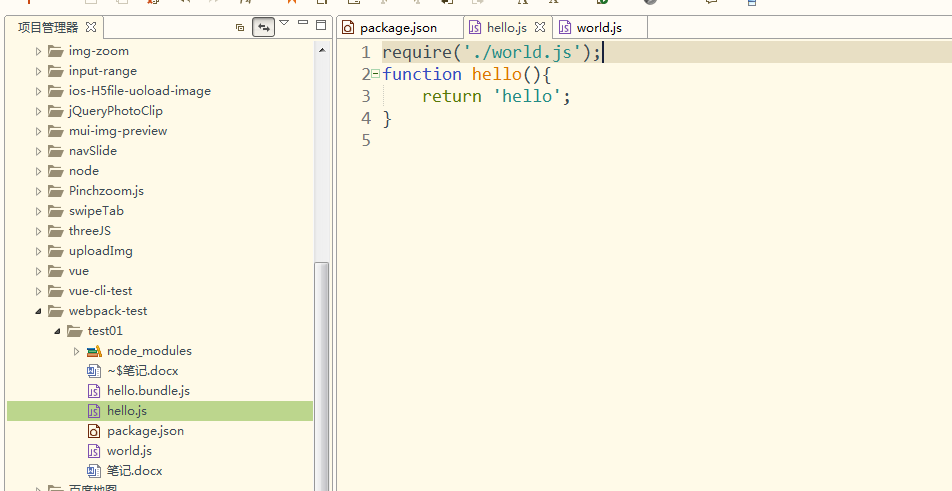
命令说明：

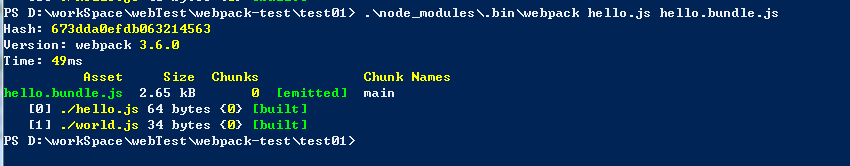


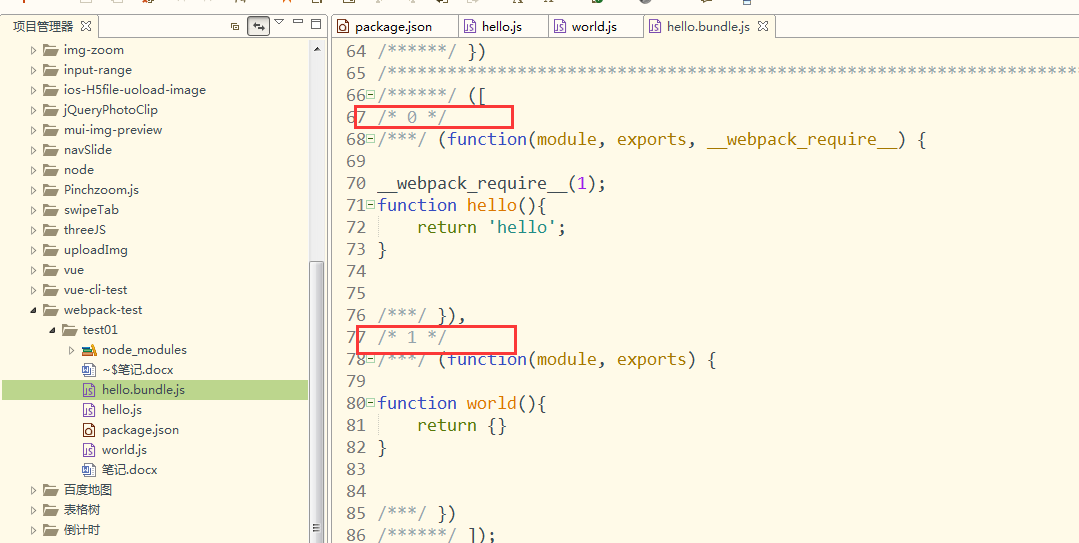
1. 测试require引入文件打包

（1）、Webpack会根据require，遍历所有引用的文件，然后将这些文件打包成一个文件（all in one）

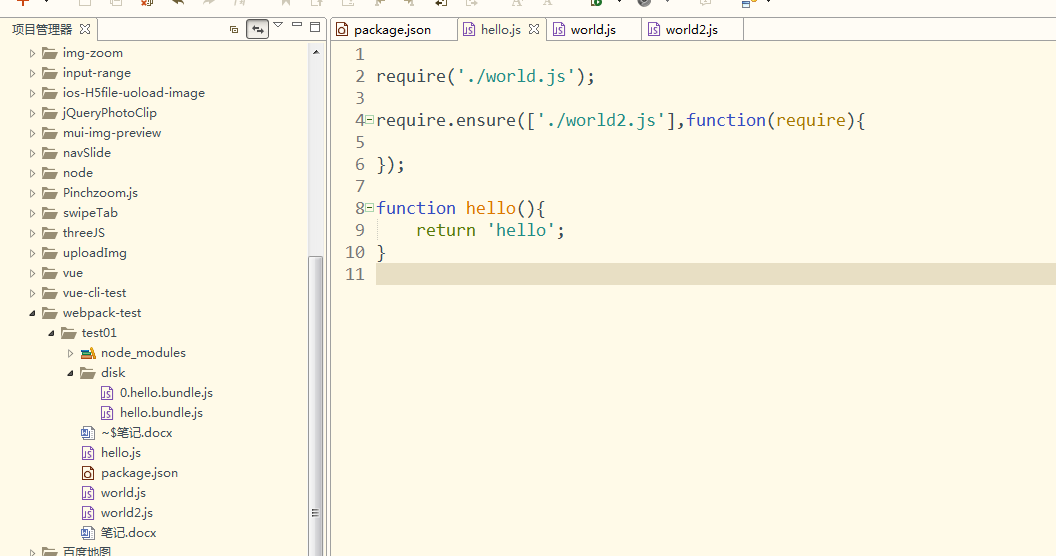
（2）、如果需要按需加载，则需要了解webpack的code spliting，简单的说，就是使用require.ensure引入文件

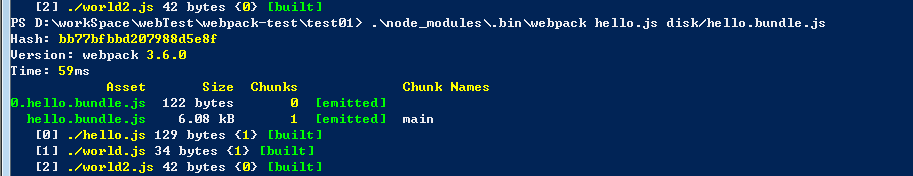




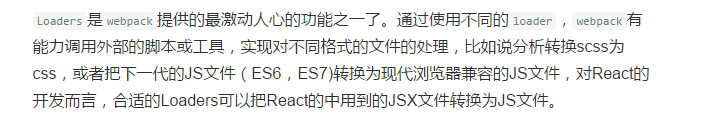


1. code spliting 测试



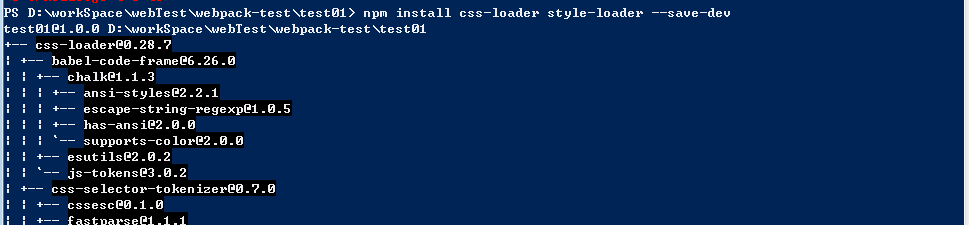


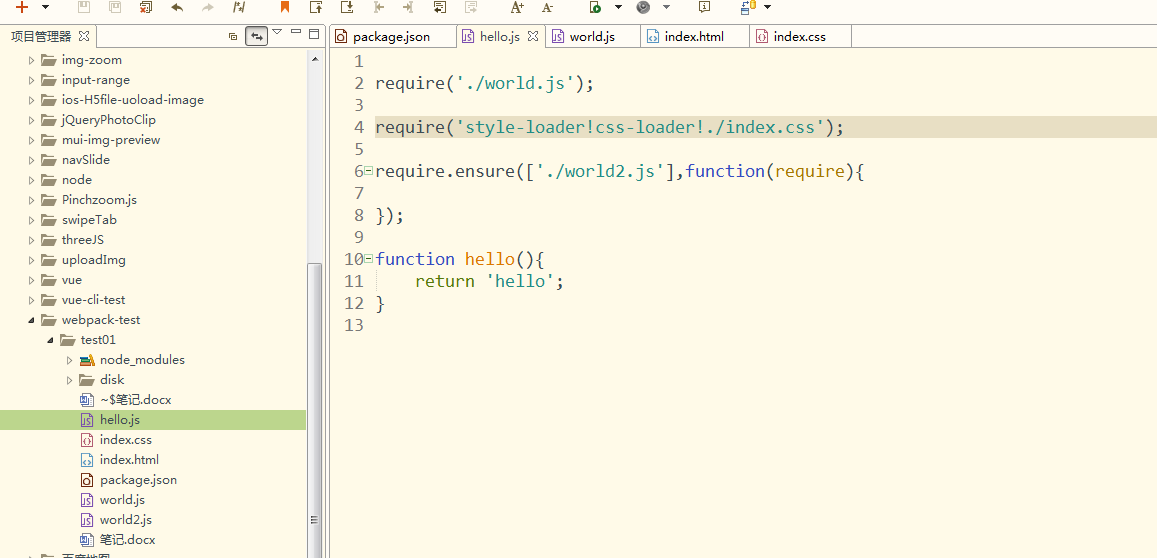
1. 打包css文件，需要引入css-loader，style-loader

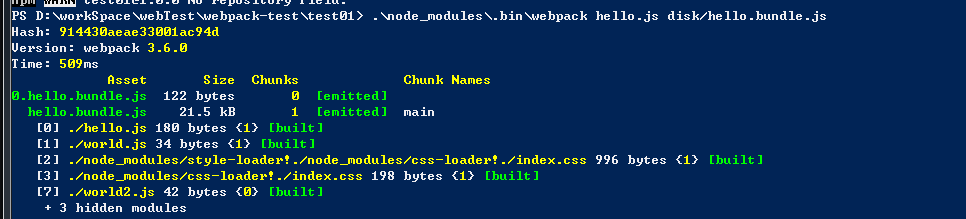


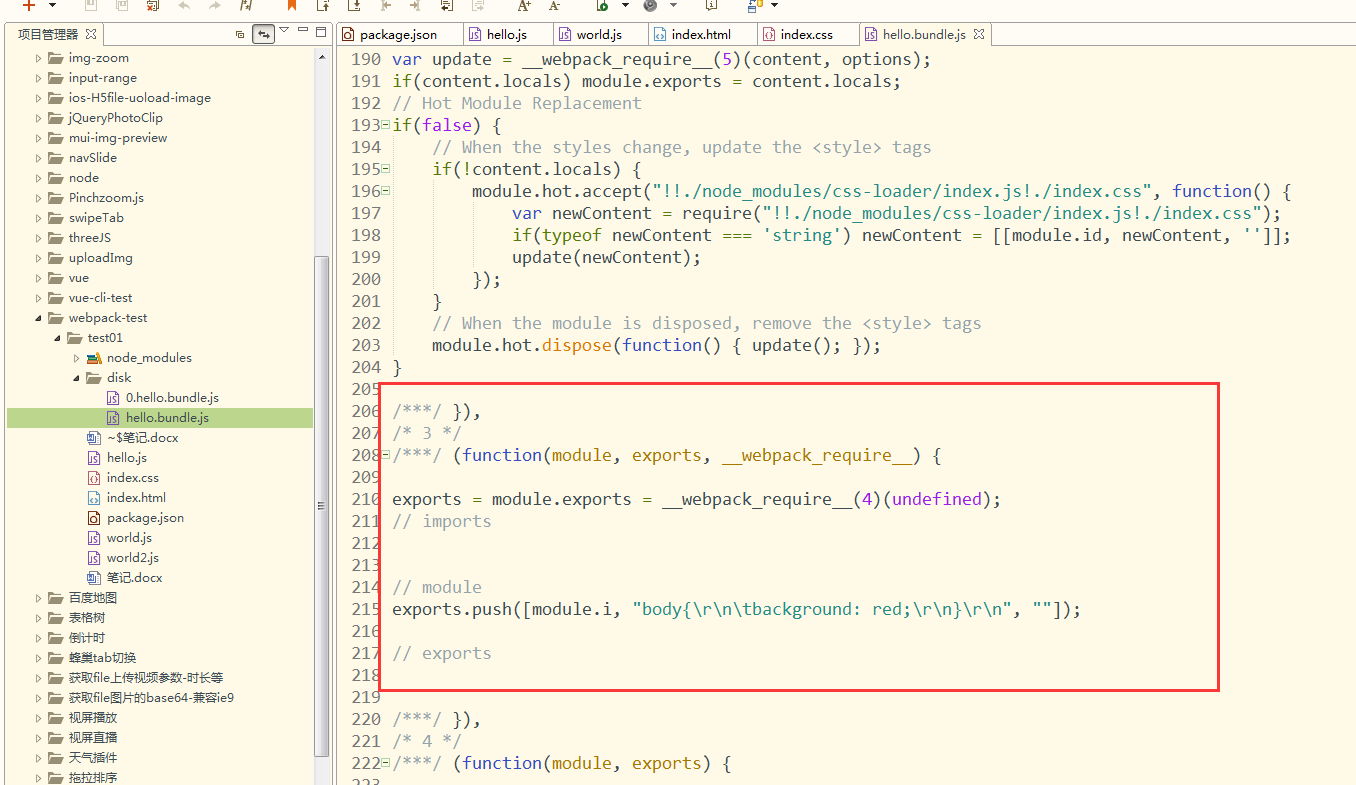
css-loader使得webpack可是处理.css为后缀的文件，将其打包

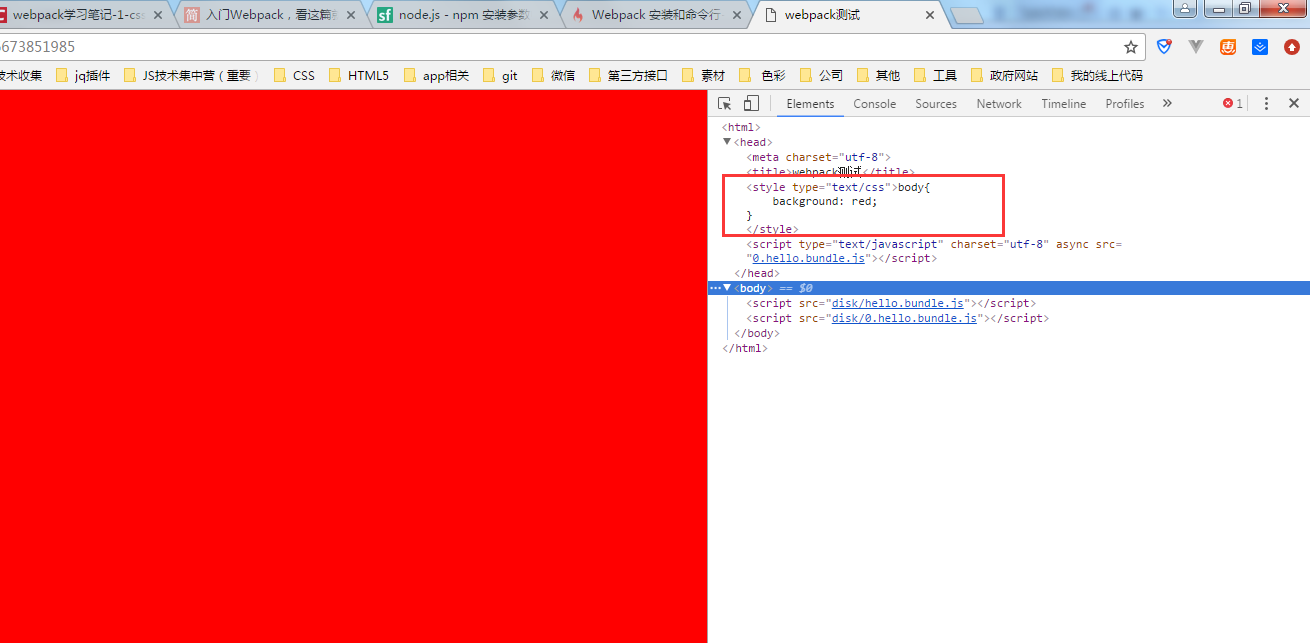
style-loader则是将css-loader处理后的代码以<style>.....</style>的形式插入html的head中



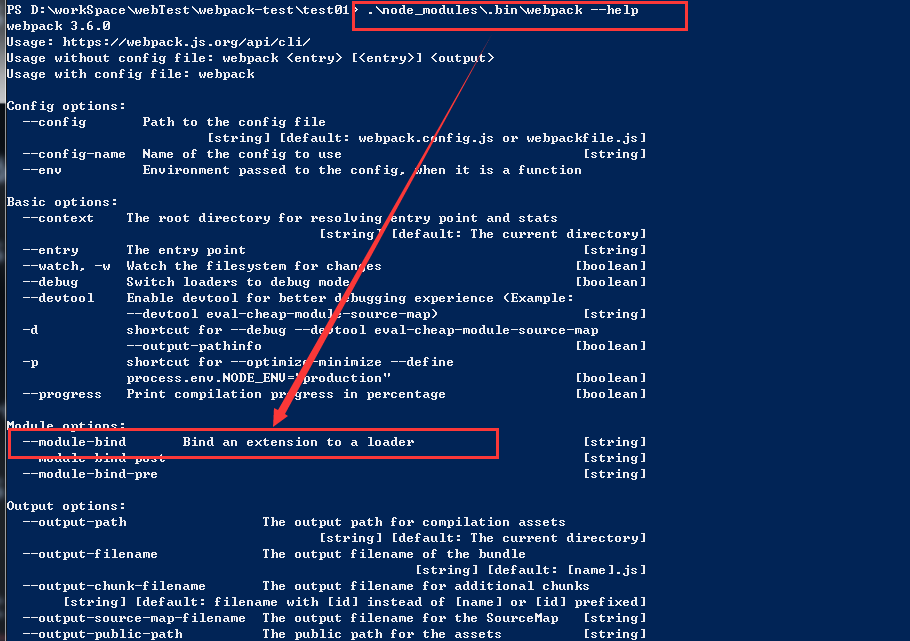








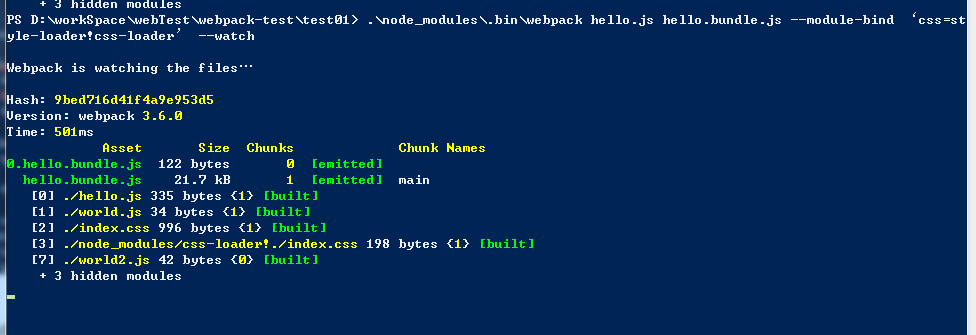
1. 通过命令指定loader打包css



.\node\_modules\.bin\webpack hello.js hello.bundle.js --module-bind ‘css=style-loader!css-loader’

8、通过--watch命令，监听文件修改，自动打包

.\node\_modules\.bin\webpack hello.js hello.bundle.js --module-bind ‘css=style-loader!css-loader’ --watch



文件修改后

