

用webpack去翻译index.js文件，同级目录出现dist文件夹，dist下会有打包压缩好的main.js

```
npx webpack index.js
```

创建项目

```
mkdir webpack-demo  
cd webpack-demo  
npm init -y
```

在项目中安装webpack

如果在项目中安装webpack,不是全局安装的，在终端运行例如webpack -v的命令，会报错,因为node会在全局模块目录中查找，因此找不到，这时可以用npx命令，如npx webpack -v,npx会帮我们在当前项目的node-modules目录下查找webpack

```
npm install webpack webpack-cli --save-dev  
npm install webpack webpack-cli -D  
npm install webpack webpack-cli -g
```

推荐指定版本的安装方法

导入导出方式

```
// ES Module 导入  
import Header from './src/header.js'  
// ES Module 导出  
export default Header  
  
// CommonJs 导入  
const Header = require('./src/header.js')  
// CommonJs 导出  
module.exports=Header
```

默认配置文件

打包过程中，默认查找webpack.config.js文件，是webpack的默认配置文件，如果找不到这个文件，可以指定打包的配置文件，比如有文件叫webpackconfig.js,可以运行npx webpack --

config webpackconfig.js,来指定以webpackconfig.js为打包的配置文件

```
//webpack.config.js
const path=require('path');
module.exports={
  mode:'production',//默认production,development|none,设置为development时,生成的
bundle.js是不被压缩的代码
  //entry: './src/index.js',
  entry:{main: './src/index.js'},
  output:{
    filename:'bundle.js',//打包后的文件名
    path:path.resolve(__dirname,'dist'),//打包后的文件要放在哪,后面是绝对路径,打包后生
成dist文件夹
  }
}
```

npm scripts标签配置项

在package.json的scripts标签中配置,可以指定运行命令要干什么, npm scripts中运行npm run bundle, 实际就是运行webpack,这个时候,会优先从项目node_modules中查找webpack,而不会全局查找

```
//package.json
//运行npm run bundle,就是运行webpack命令
"scripts": {
  "bundle":"webpack"
},
```

loader

解析项目中引入的文件,比如.jpg文件,.vue文件等,通过相应的loader解析,打包到dist目录下,并返回打包的文件名

常见报错

Unexpected identifier, Cannot use import statement outside a module

模块引入用ES module引入方式,相应文件应该用ES Module导出方式导出