1. 模块化：
2. **nodejs模块化的规范(commonjs)**：

单个暴露：

exports.属性 = 值

exports.方法 = function(){}

module.exports.属性 = 值

module.exports.方法 = function(){}

批量暴露:

module.exports = {}

**注意**：单个暴露与指暴露不能一起使用

1. 使用nodejs模块化规范(commonjs)定义的模块只能在后端使用，但不能在前端使用，如果想在前端使用则需要使用工具browserify来编译代码：

1)browserify官网：browserify.org

browserify主要用来将后台的代码编译成前端可以运行的代码；

2)安装 browserify当作工具来使用

npm install -g browserify

//-g全局安装时要注意:

//1)、安装时不用考虑所在位置

//2)、安装后的文件存放位置：C:\Users\当前系统用户名\AppData\Roaming\npm

3)用法

browserify 源文件 -o 目标文件

4)实例

browserify ./src/app.js -o ./dist/bundle.js

//browserify打包成功以后没有提示， 这个工具具有创建文件夹的功能

1. **前端(es6)模块化语法**：

(1). 单个暴露

export var m = 'mm'

export let n = 'nn'

export function fn(){

}

(2). 整体暴露

let x = 'xx'

let y = 'yy'

let fn2 = function(){

}

export { x,y,fn2 }

1. 使用模块:

import { 属性名/方法名... } from '文件路径'

1. 使用es6模块化语法定义的模块不能在vscode工具中直接运行，需要使用工具babel将es6语法转换成es5语法，然后才能在vscode工具中运行：
2. babel官网：www.babeljs.cn

babel的功能：将es6语法转成es5语法、还能操作jsx语法(react)等

1. 在dos小黑窗中安装babel-cli, babel-preset-es2015和browserify:

npm install babel-cli browserify -g

npm install babel-preset-es2015 --save-dev

参数说明：

--save 项目依赖（真正上线，也需要）

--save-dev 开发依赖（开发阶段需要，上线不需要[因为已转换好了]）

presets 预设(将es6转换成es5的所有插件打包)

(3)在项目中定义babel配置文件：.babelrc文件，内容如下：

{

"presets": ["es2015"]

}

3)、编译

使用Babel将ES6编译为ES5代码(但包含CommonJS语法) :

babel ./src -d ./dist

如果想在前端中运行es6模块化的代码则需要使用Browserify编译js :

browserify ./dist/app.js -o ./dest/bundle.js