# 运营管理平台开发流程介绍

## 项目下载安装运行

### 1.1、下载并安装nodeJS环境

http://nodejs.cn/download/，下载安装好后，命令窗口输入：npm -v查看是否安装成功。

然后再安装淘宝npm镜像

npm install -g cnpm --registry=https://registry.npm.taobao.org

最后输入命令：cnpm -v查看时候安装成功。

### 1.2、angular-cli安装

命令：cnpm install -g angular-cli

### 1.3、安装与运行

1.3.1、使用cnpm i -g yarn 或 npm i -g yarn安装yarn

1.3.2、配置registry地址：yarn config set registry <http://registry.npm.taobao.org>

1.3.3、配置sass-binary-site地址:

yarn config set sass-binary-site <http://npm.taobao.org/mirrors/node-sass>

### 1.4、进入项目

命令：yarn install

### 1.5、启动

命令：cnpm start

### 1.6、打包部署

命令：cnpm run build

## 目录介绍

Project

|-- e2e

|-- src

|-- environments 环境

|-- environment.prod.ts 生产环境

|-- environment.ts 非生产环境

|-- app

|-- app.component.{ts,html,css,spec.ts} 使用HTML模板、CSS样式和单元测试定义AppComponent组件

|-- global-constant 静态全局数据

|-- business 页面主组件

|-- demo demo实例

|-- department 部门管理页

|-- home 主页

|-- menu 菜单管理页

|-- user 用户管理页

|-- business-sevice 组件服务

|-- business-shared 页面细分组件

|-- user 用户管理子组件集合

|-- user-list.component.html 组件的html

|-- user-list.component.scss 组件的样式

|-- user-list.component.spec.ts 组件的单元测试

|-- user-list.component.ts 组件的业务逻辑

|-- user-shared.module.ts 组件的模块管理

|-- assets 静态资源

|-- error-page 404页面

|-- login 登陆页

|-- noPower-page 无权限的提示页面

|-- pipe 管道

|-- main 系统主体

|-- index.html 入口主页面的HTML文件

|-- main.ts 应用的主要入口点

|-- polyfills.ts 不同的浏览器对Web标准的支持程度也不同。

|-- styles.scss 导入一些全局样式

|-- test.ts 元测试的主要入口点

|-- tsconfig.{app|spec}.json ypeScript编译器的配置文件。tsconfig.app.json是为Angular应用准备的，而tsconfig.spec.json是为单元测试准备的。

|-- package.json 依赖资源表

```

## 项目router和模块依赖

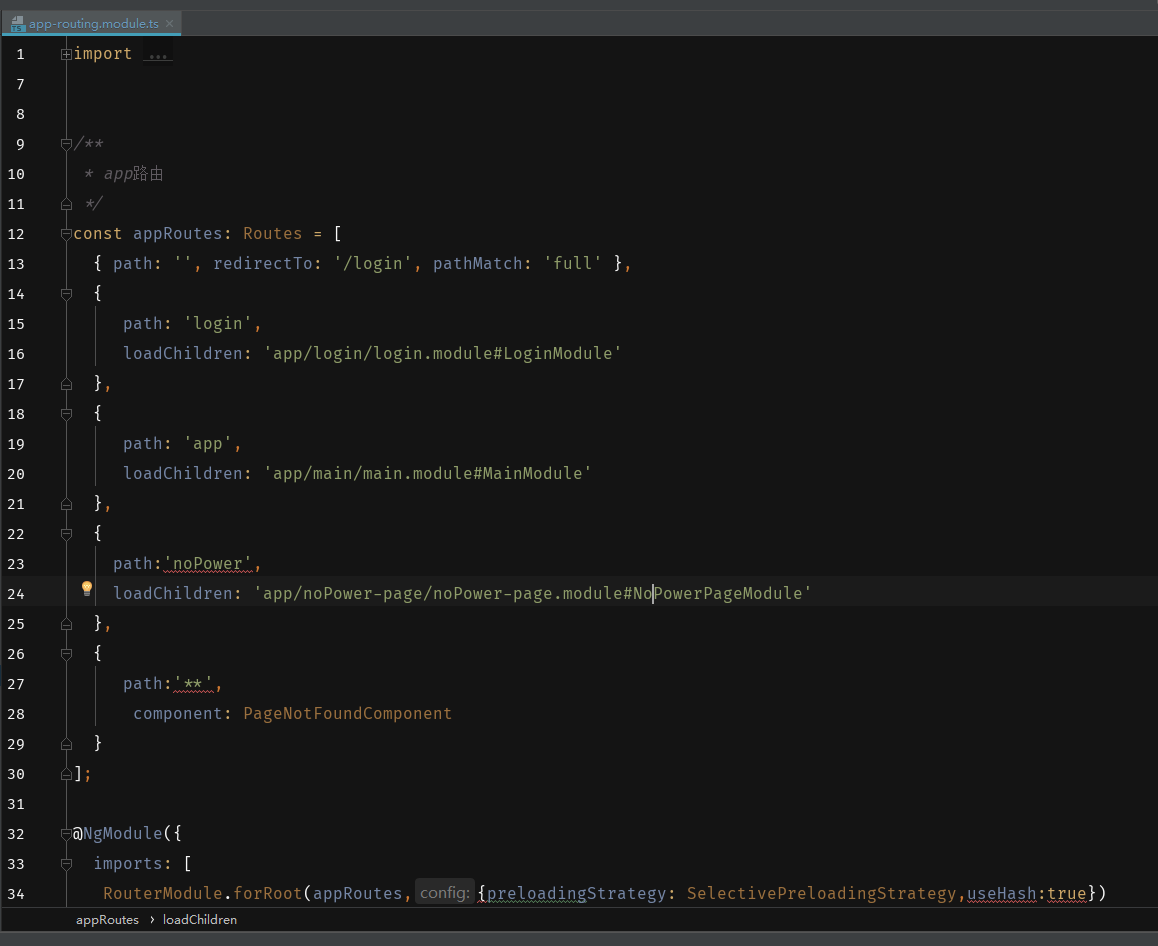
### 3.1、定义

Router：路由其实就是项目所有模块的导航；

Module：模块的目的是用来组织app的逻辑结构，可以管理模块内部的Components、Directives、Pipes，引入Service，并控制外部组件对内部成员的访问权限。

### 3.2、项目总路由

app/app-routing.module.ts：分别包含登录、无权限提示页、404页、项目主页



### 3.3、项目主页

app/main/main.module#MainModule意思是加载main模块，然后再根据main模块的路由找到对应路由的模块。

如图1是main模块的路由设置(app/main/main-routing.module.ts)，里面的子路由path=’user’时候，即浏览器访问http://localhost:4200/#/app/user时，调用了app/business/user/user.module#UserModule模块

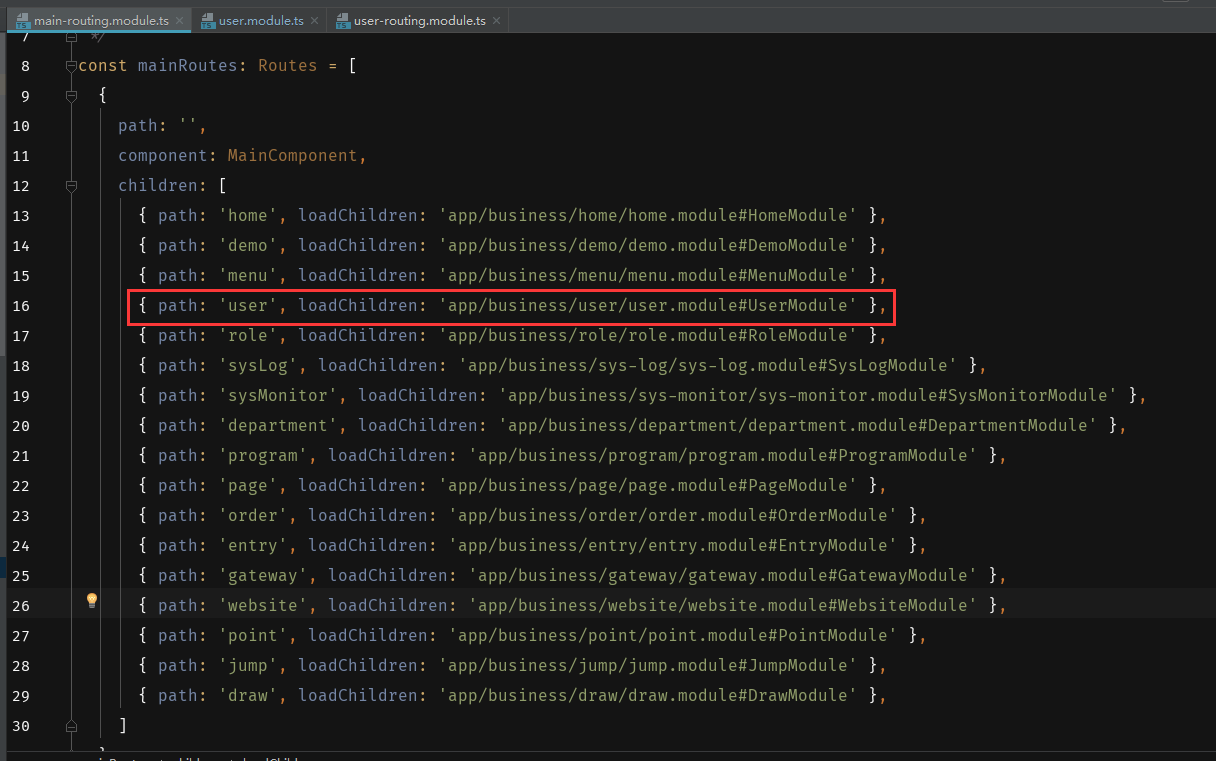


图2是user模块：

①是导入所需的模块(module)、组件(component)、管道(pipe)、服务(service)；

②imports是导入其他模块，就是要使用其他模块的功能，必须要导入

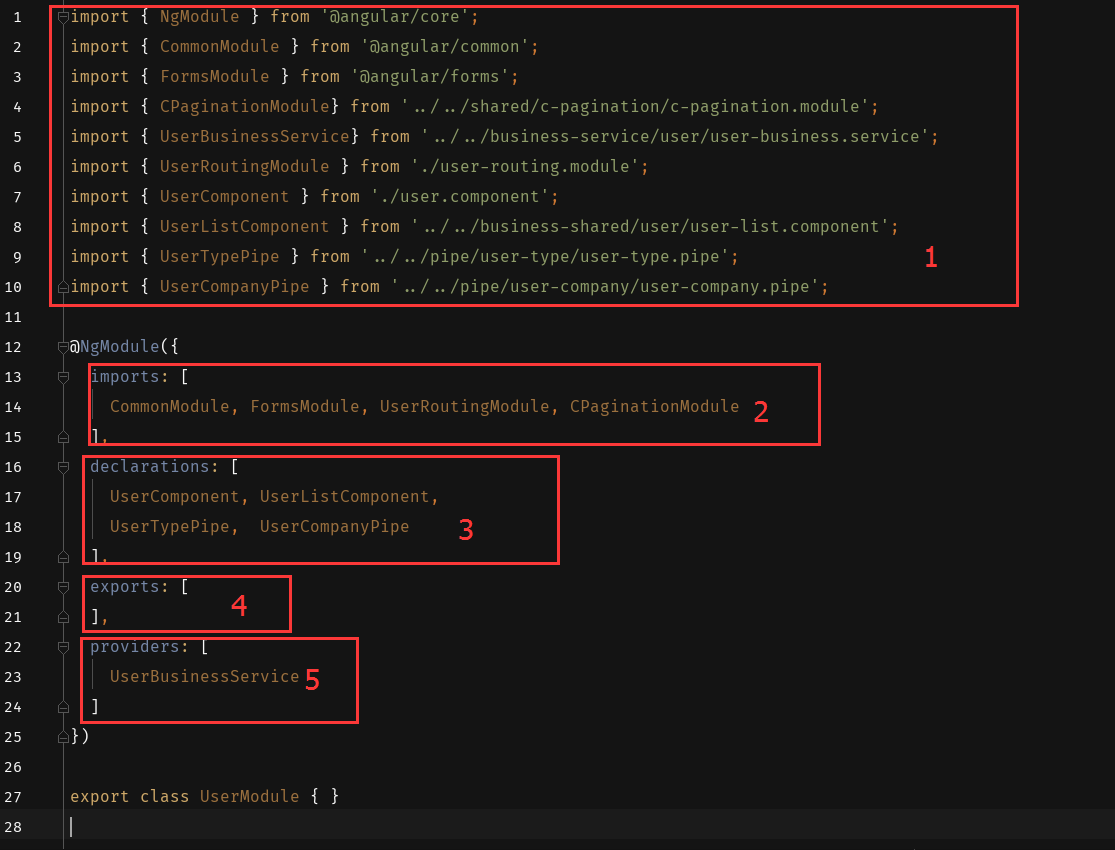
③declarations声明本模块包含的内容，比如list组件

④exports外部可见的内容。相当于java中声明为public的那些类

⑤providers服务提供者，主要用来定义服务

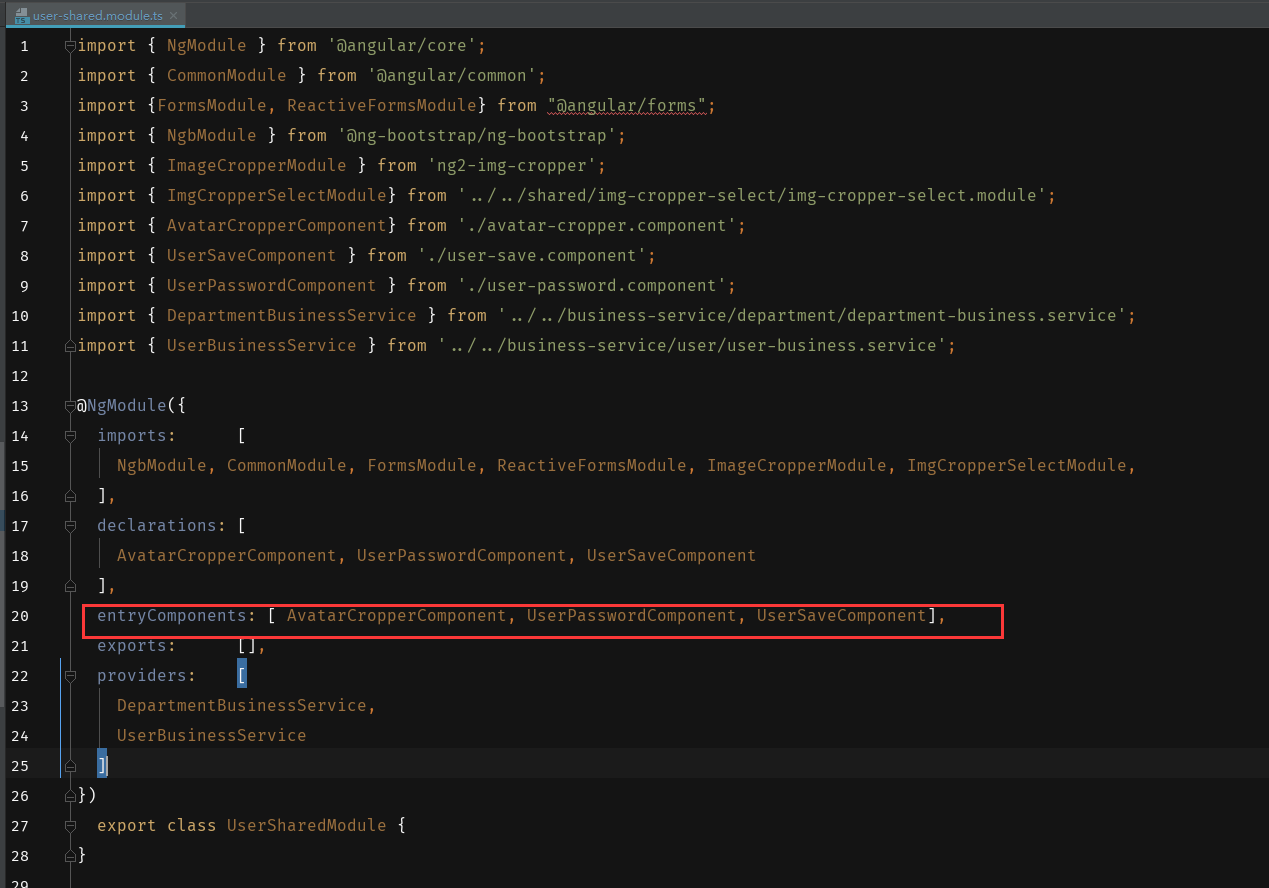
⑥bootstrap启动模块。只在根模块(app.module.ts)使用。在除了根模块以外的其他模块不能使用。

⑦entryComponents这里的组件将会在这个模块定义的时候进行编译并为它们创建工厂，一般都是

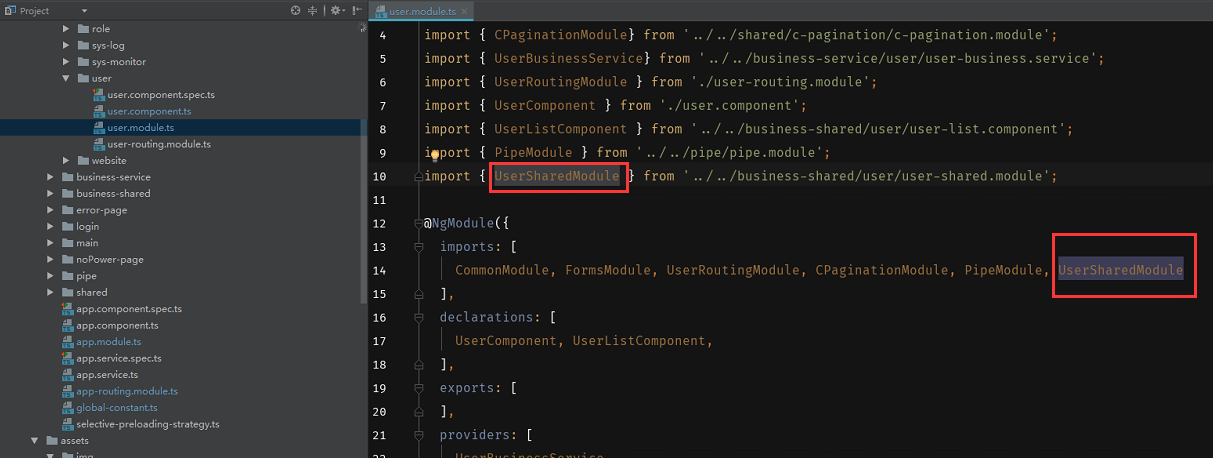


### 3.4、项目组件

从user模块加载所需的组件，包括list和一些弹窗组件。其中所有的组件，除list外，都添加在app/business-shared/user/user-shared.module.ts文件



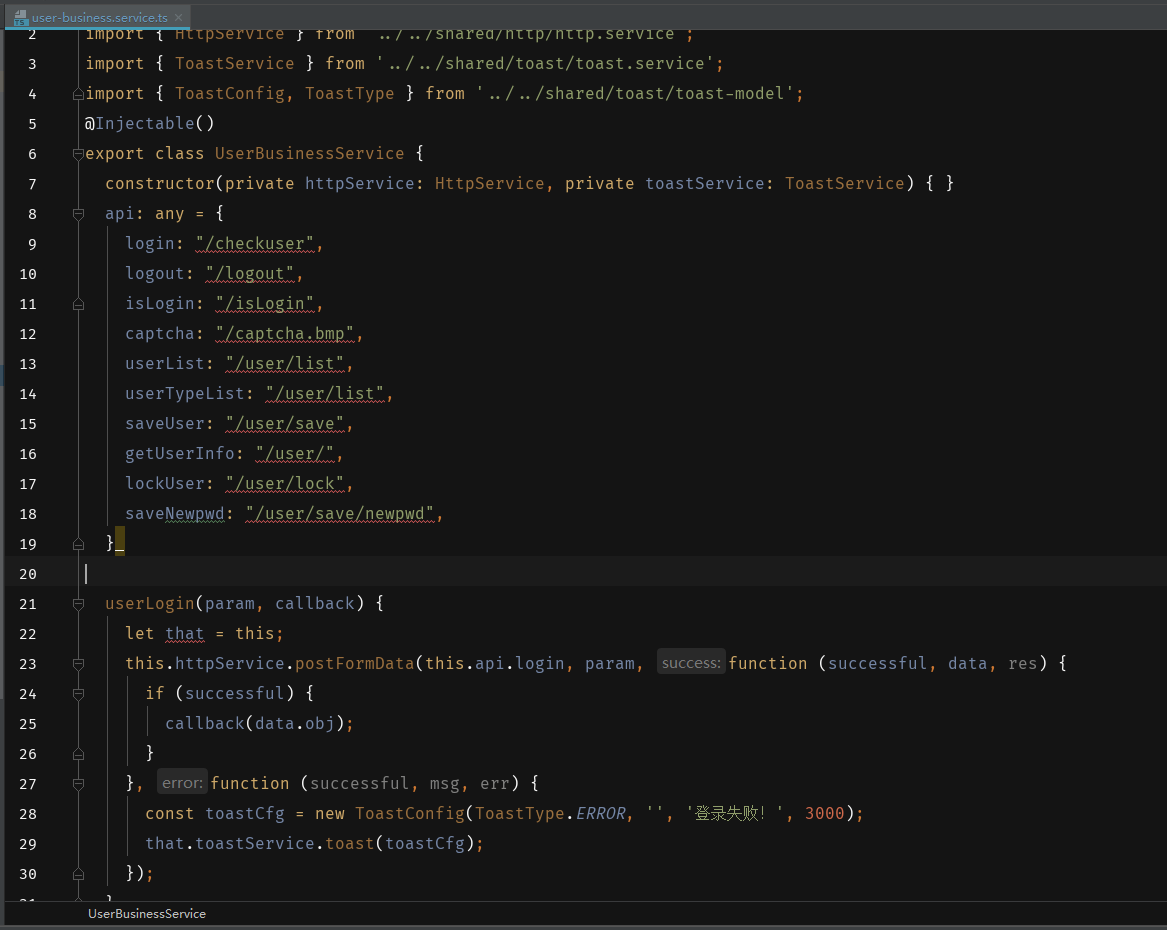
最后引入module，如：app/business/user.module.ts



## Service服务

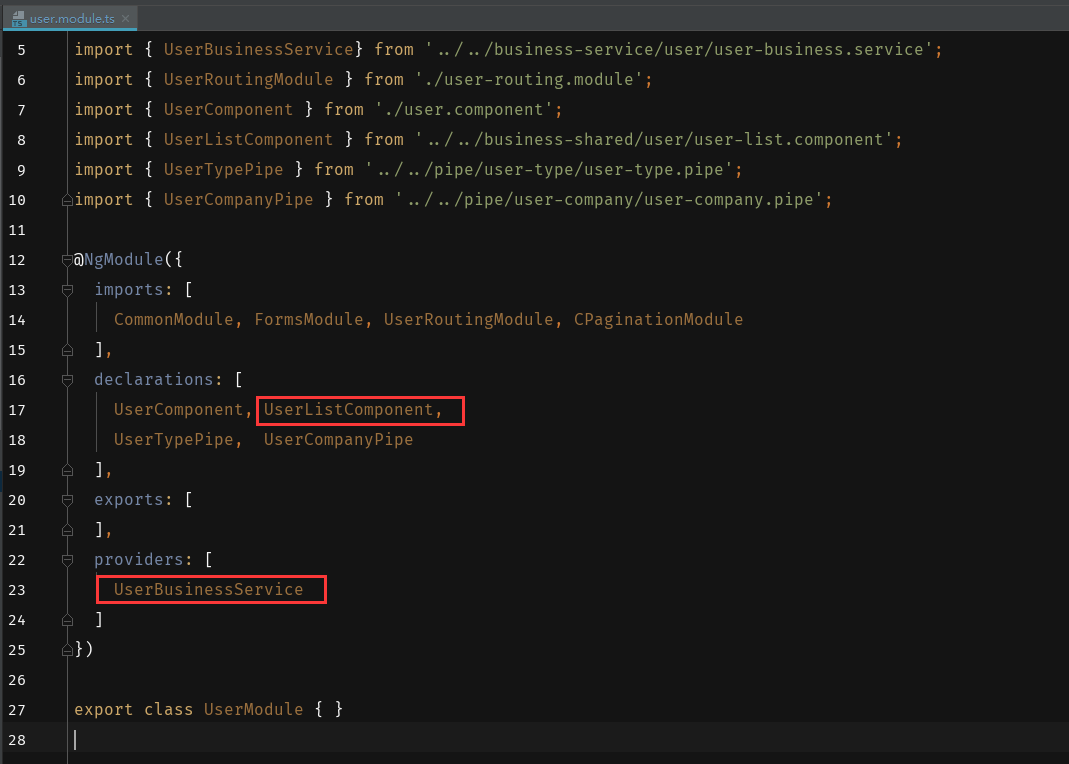
### 4.1、创建

service都是存放接口地址和请求方法，是所有接口数据的来源。(app/business-service/user/user-business.service.ts)



### 4.2、加载

在需服务数据的模块里加载。比如UserListComponent需要UserBusinessService，即在加载UserListComponent的UserModule模块里引入。



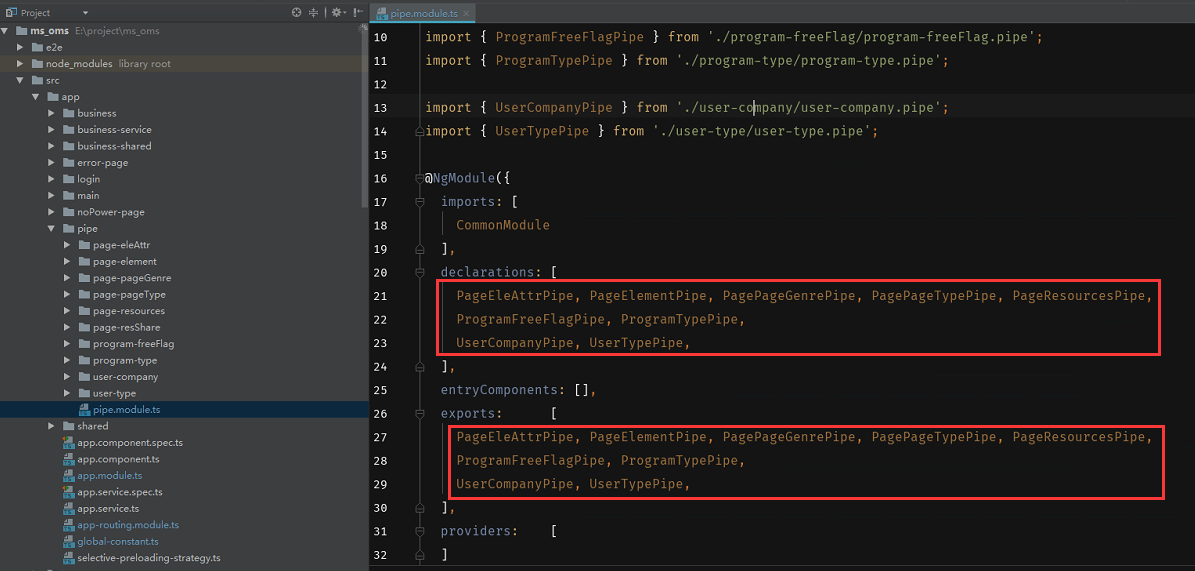
## pipe管道

### 5.1、作用

起到过滤数据的作用。如接口数据data.role=1，转为显示“管理员”。

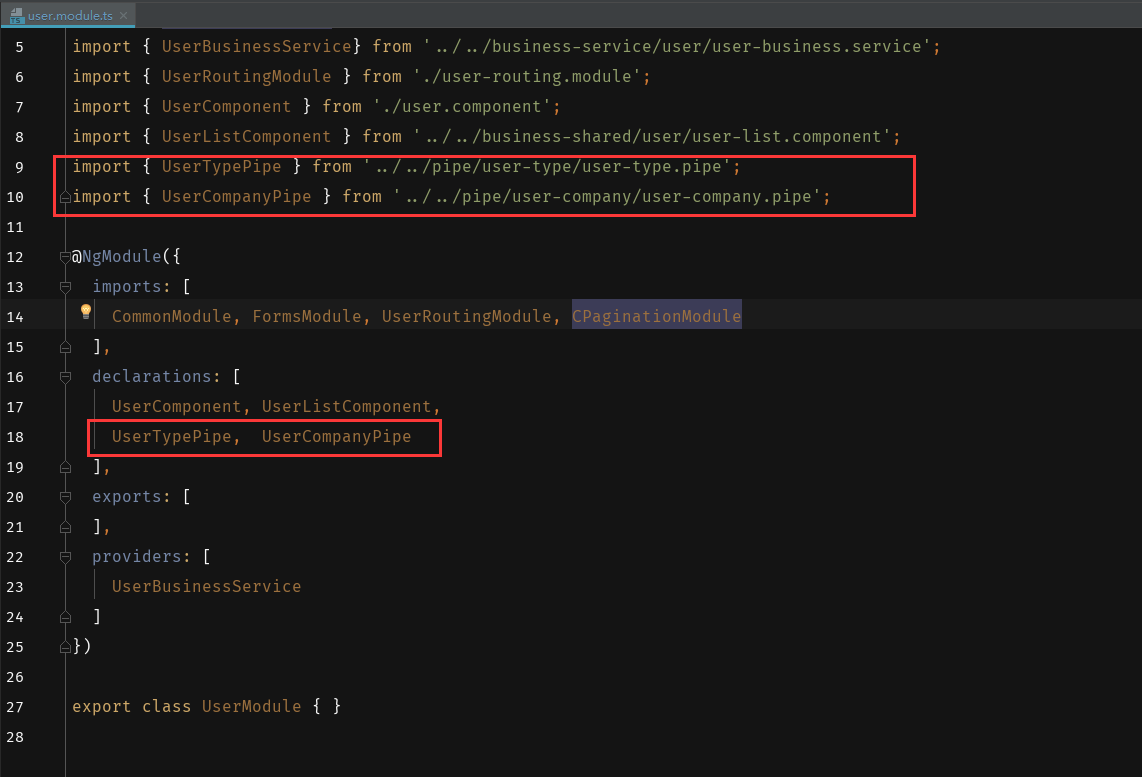
### 5.2、创建

在app/pipe/创建对应的过滤器名，如app/pipe/user-type。再加入pipe.module.ts



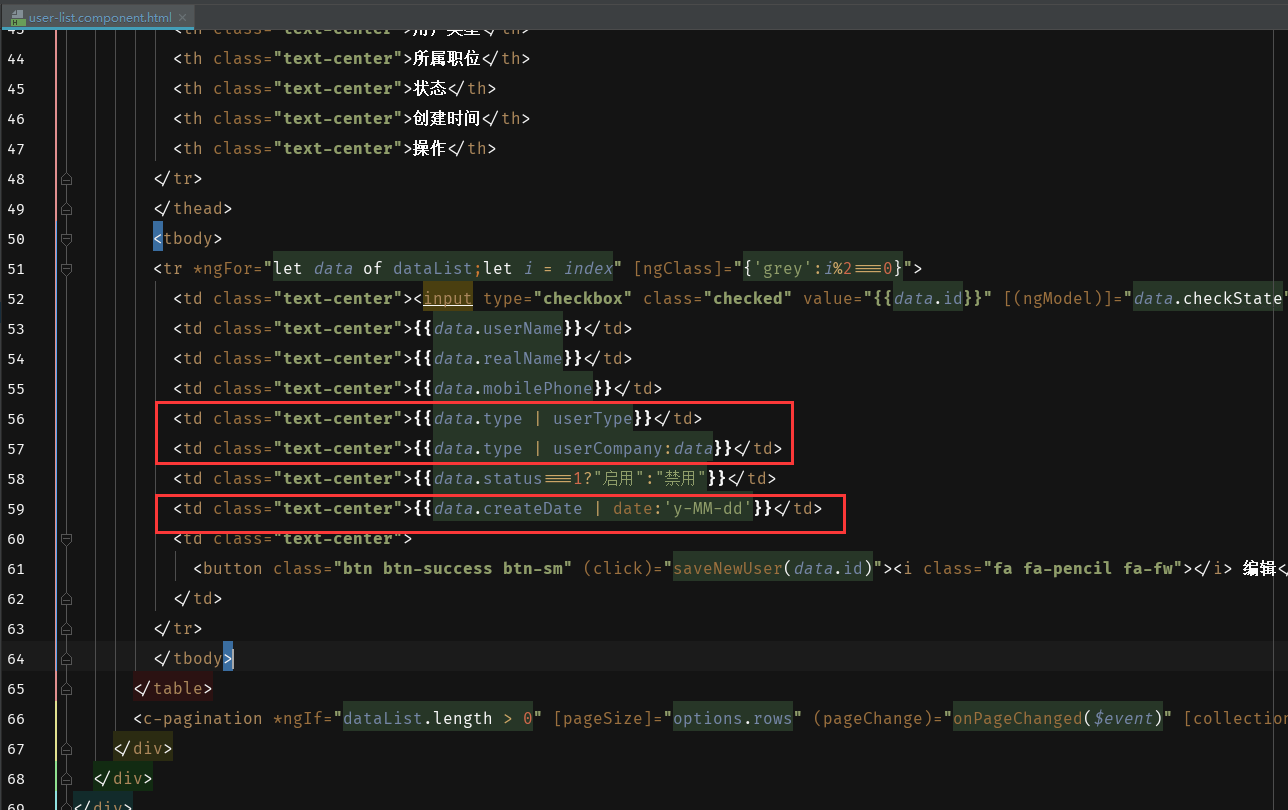
### 5.3、引入

在所需调用管道的模块里引入。比如user父模块里的list组件需要引用，则在user.module.ts加入。即组件在哪里加载，管道也在哪里加载。



### 5.4、使用

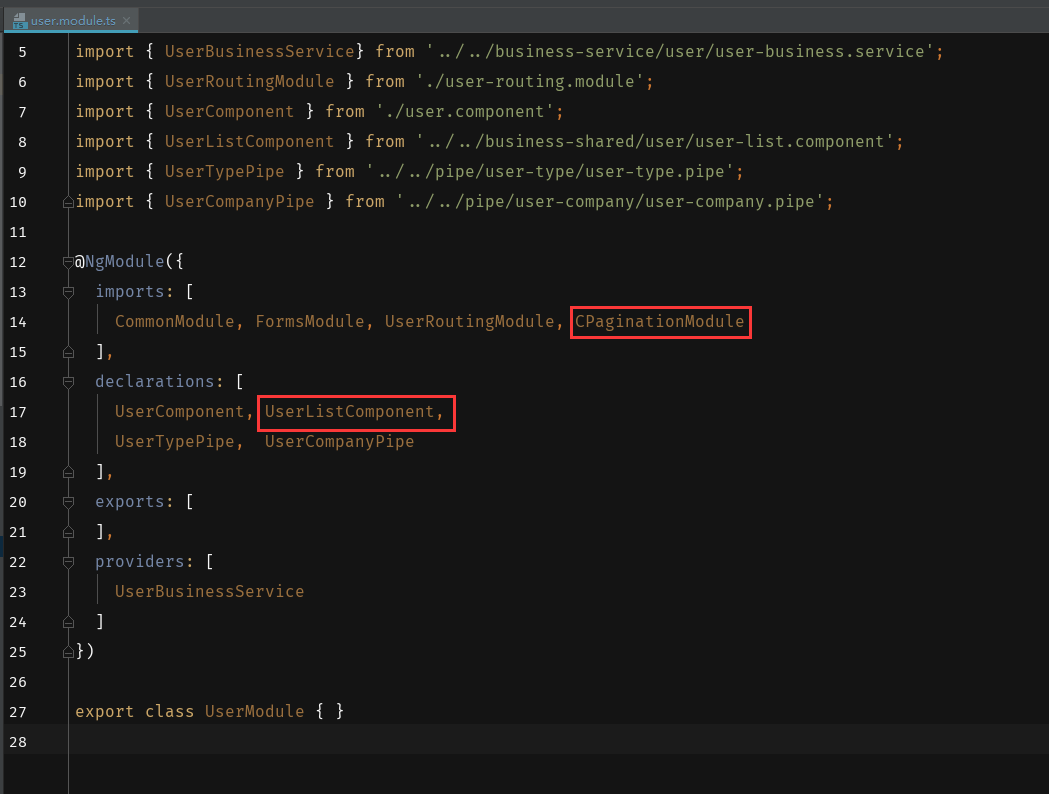
直接在模板里调用管道名（ng自带了很多管道，如下面的date）



## 分页插件CPaginationModule

### 6.1、引入

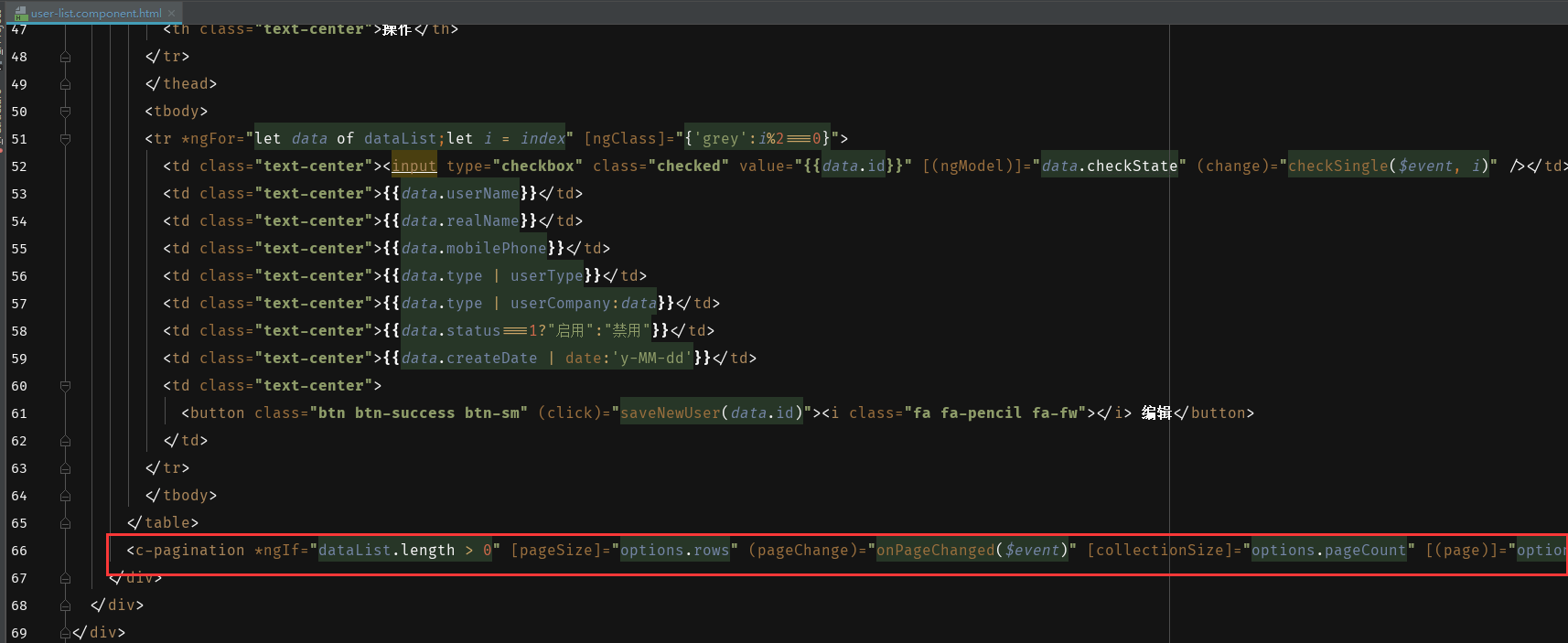
哪个组件需要，就在对应调用该组件的模块里引用。如下图



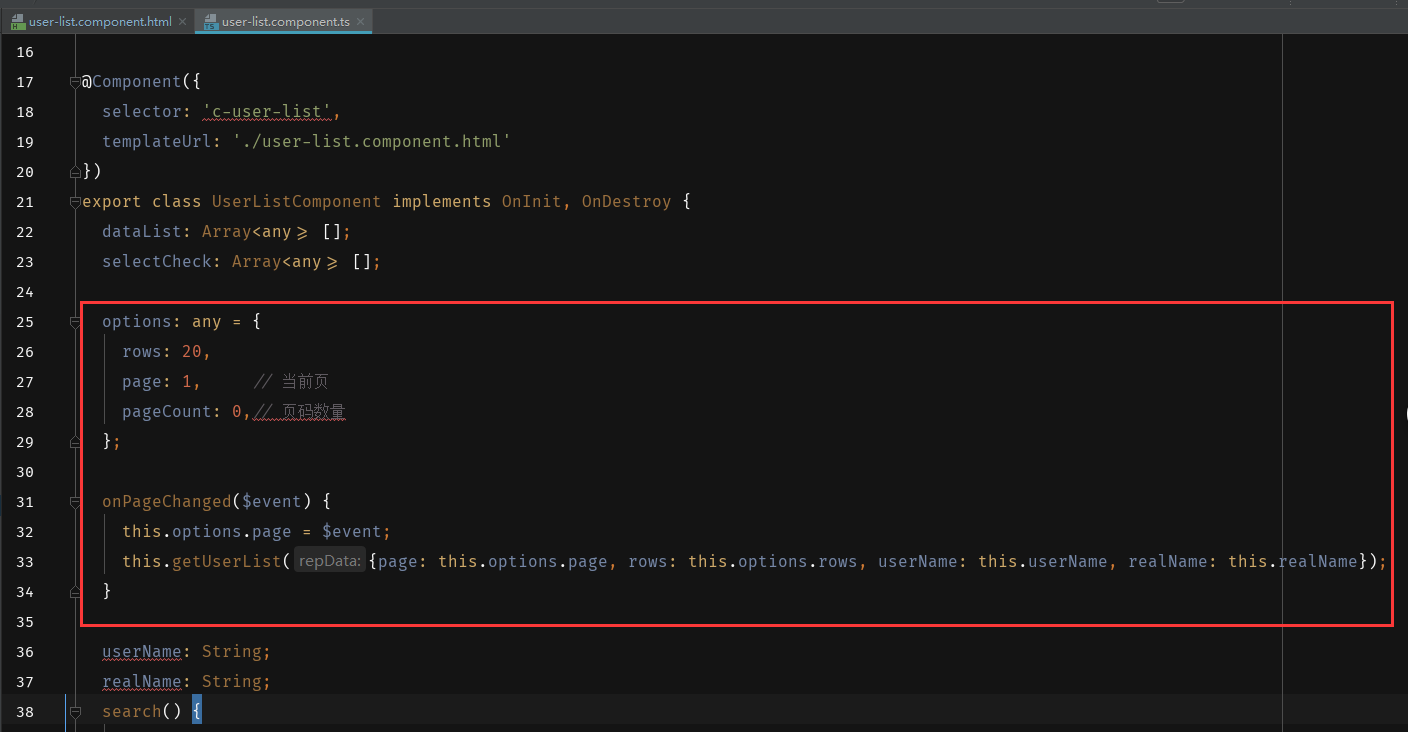
### 6.2、使用

首先在模板代码里加入分页的。具体代码看

app/business-shared/user/user-list.component.html



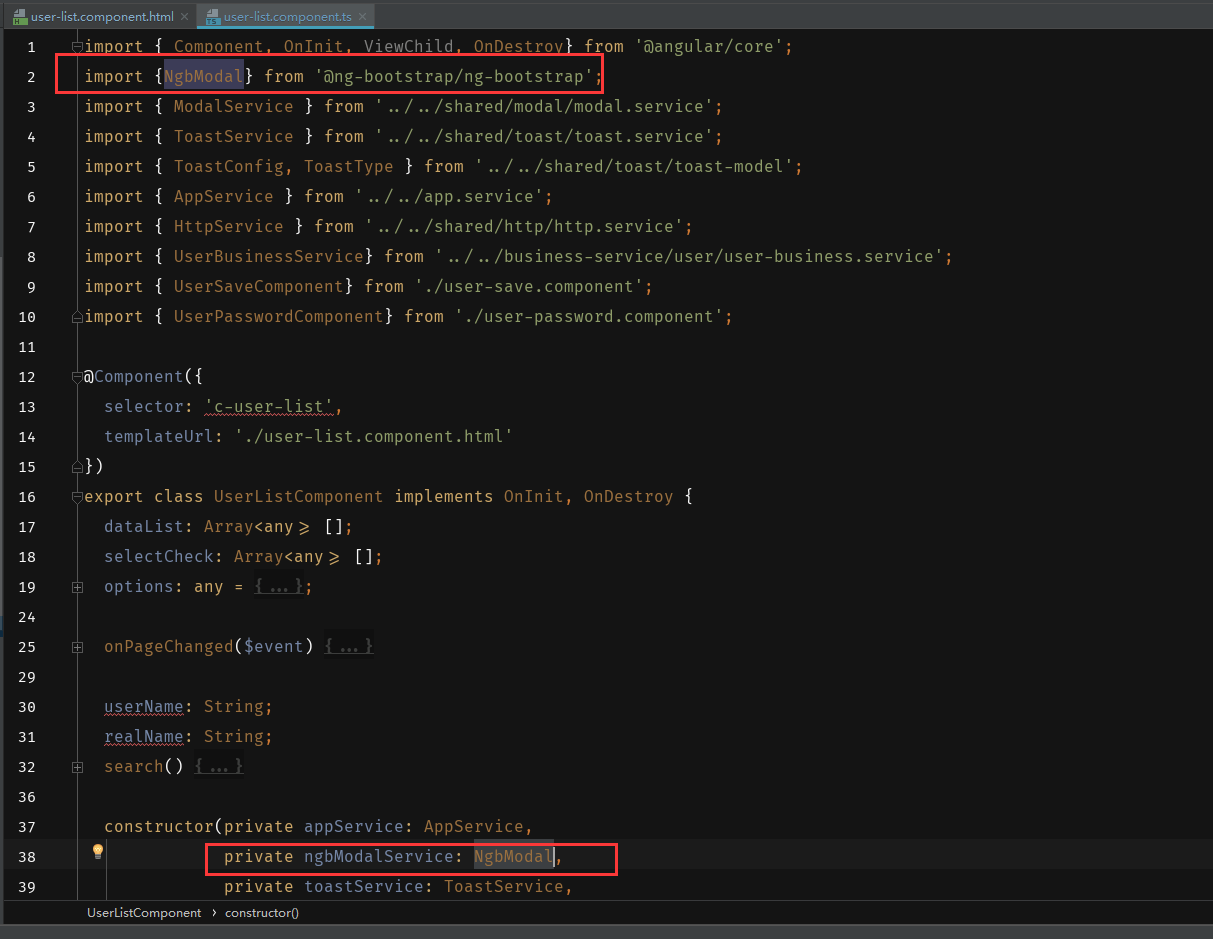
然后在UserListComponent类里添加初始化的分页数据，其中onPageChanged是页码改变时调用的方法。

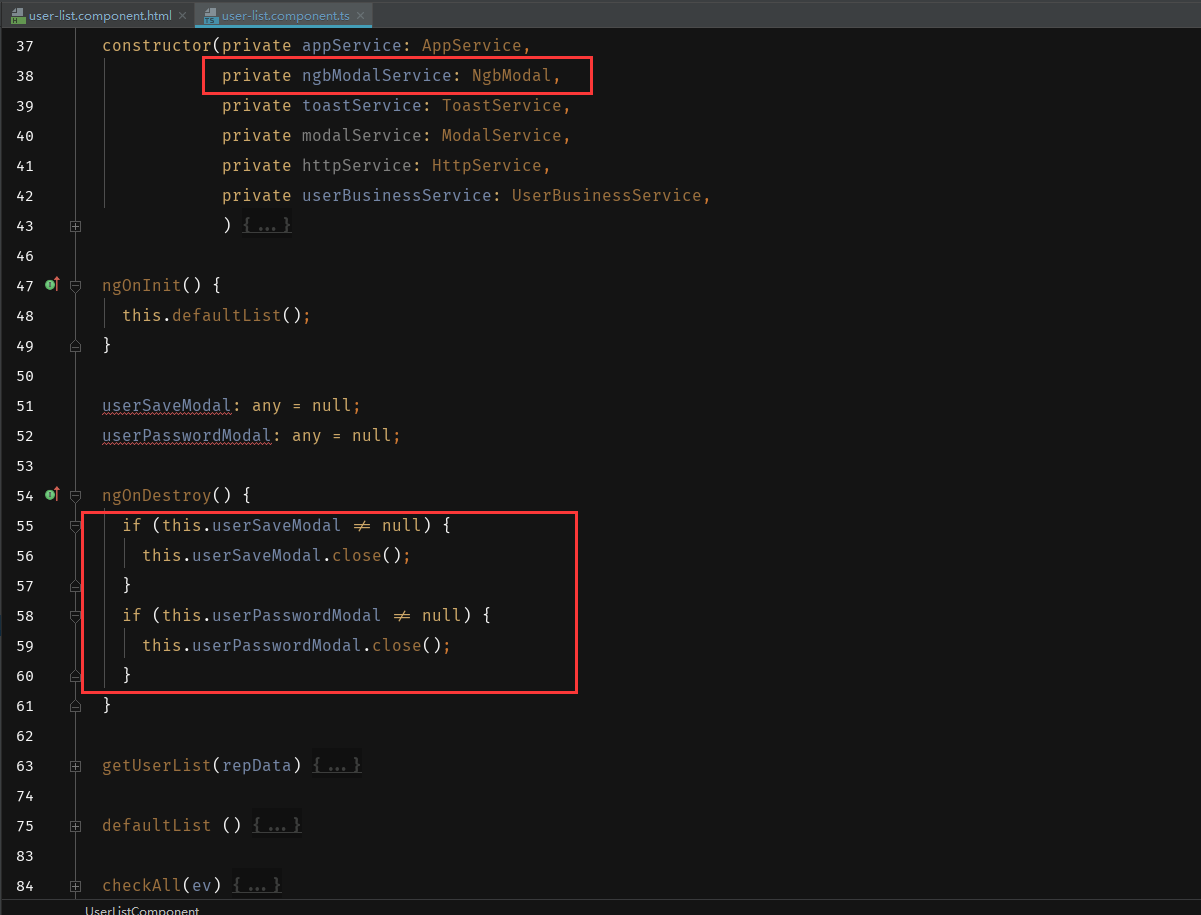


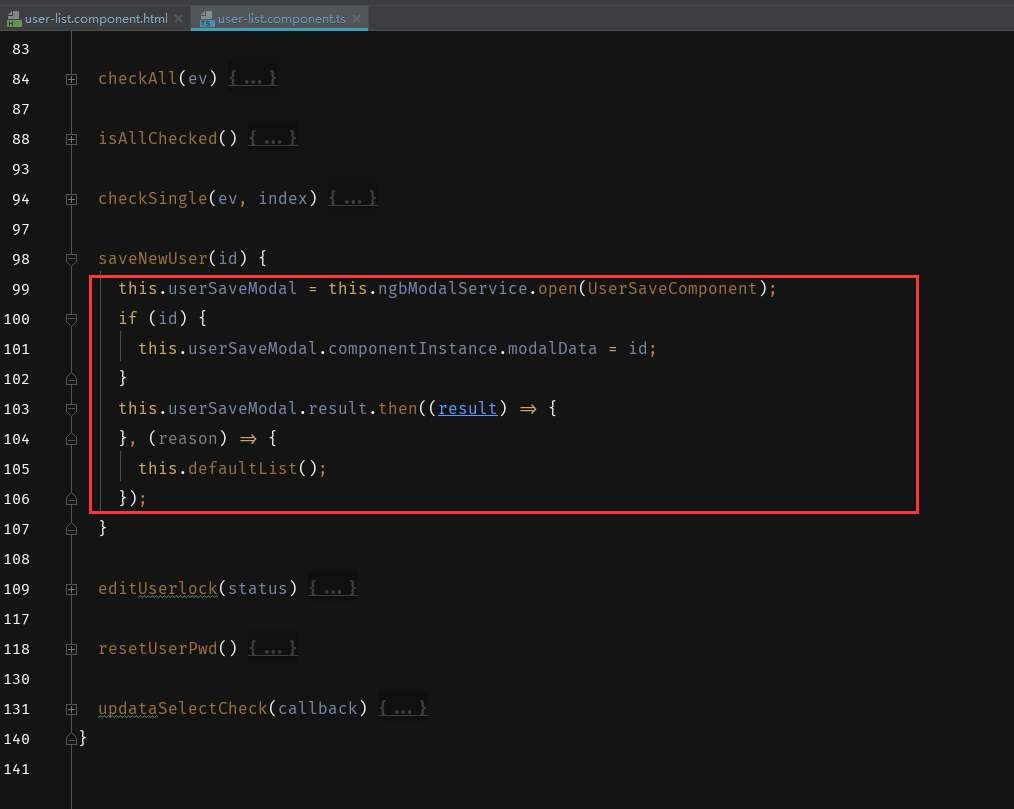
## 弹窗插件NgbModal

首先在需要调用弹窗的组件里注入弹窗NgbModal，然后constructor将其类实例化操作，后面可以直接this.ngbModalService.open(UserSaveComponent)调用，其中UserSaveComponent为弹窗内容组件。

注：调用弹窗，必须在组件销毁ngOnDestroy的同时，close弹窗，如下：







## 滚动条

首先在对应的模块里引入滚动条的模块，然后在组件模板里直接加标签c-custom-scrollbar调用即可。

