

项目初体验

1、项目地址

```
1 https://github.com/sunweiguu/xiaoxiaoxudeshop
```

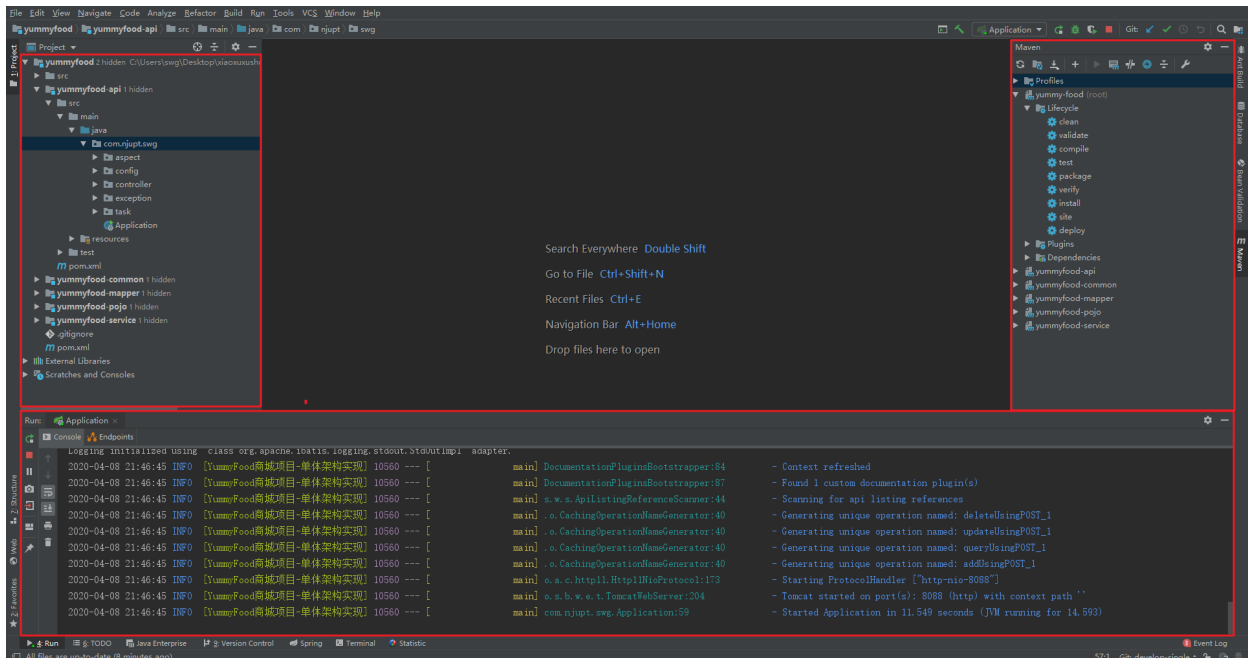
2、下载项目代码

```
1 git clone https://github.com/sunweiguu/xiaoxiaoxudeshop.git --branch master-single
```

注意我这里用分支进行开发，选择`master-single`分支下载即可，这个分支是用来承载单体项目代码的。下载下来之后用IDEA打开即可。

关于IDEA的使用、导入Maven项目，以及最基本的JDK安装、Maven安装、Mysql安装等就不再赘述了。相信对这个项目感兴趣的同学一定已经在本地准备好了基本环境。

导入项目完毕后，整体界面和代码层级应该如下所示：



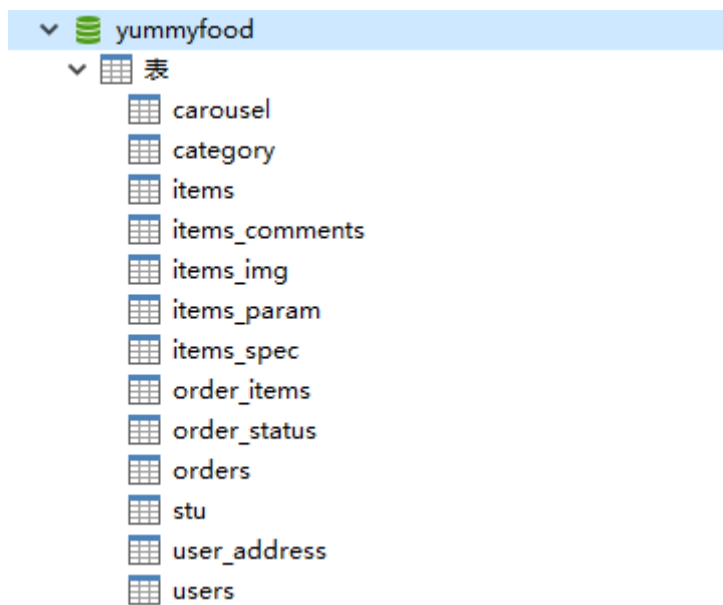
单体项目是一个Maven聚合工程，可以看到由以下几个模块组成：

- 1、yummyfood-api (接口层，即controller层)
- 2、yummyfood-service (业务逻辑层，主要是进行业务逻辑处理的一层)
- 3、yummyfood-mapper (数据操作层，即与数据库打交道的一层)
- 4、yummyfood-pojo (实体类一层，主要包括pojo、VO、BO)
- 5、yummyfood-common (通用的工具类层，如常用工具类、枚举、常量等，上面每一层都可能用到，所以抽取出来公共使用)

后面会详细说明这个聚合工程是如何搭建的。本章节聚焦整体项目功能的展示。帮助大家初次下载代码即可在本地运行起来。IDEA右边是maven的命令，因为我们的项目是一个maven的聚合工程，是基于maven搭建的，因此我们可以利用maven进行工程的打包和管理。注意，此时还不能直接运行代码，因为我们还差一个重要的mysql还没准备。

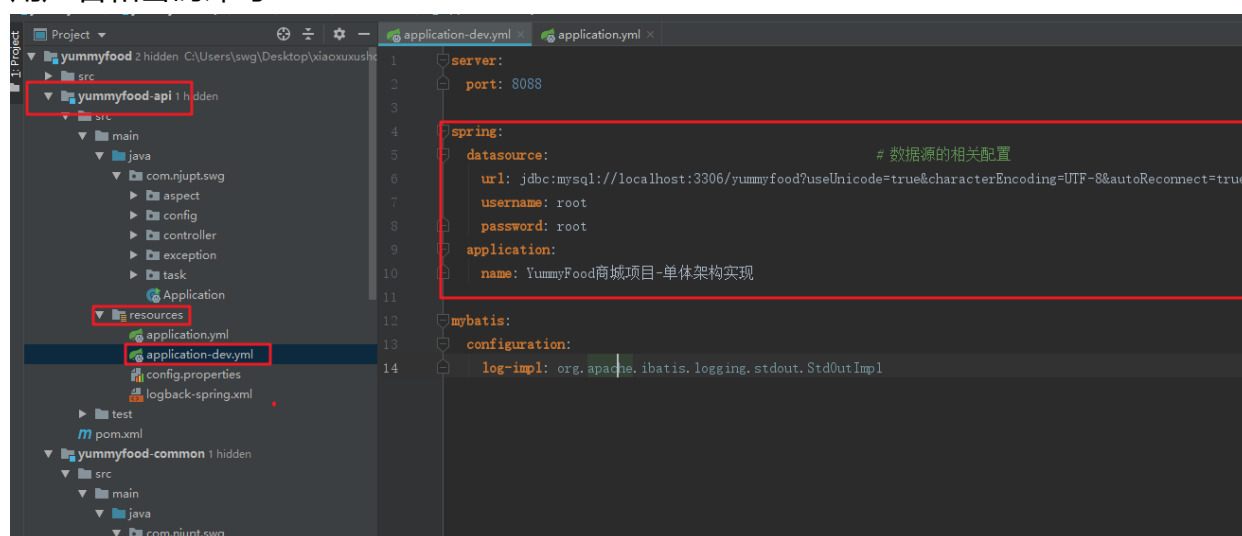
3、数据库的准备

默认本地已安装mysql，并且熟悉navicat，这个工具是一个可视化的客户端，方便进行数据库的操作。新建一个database，导入执行`yummyfood.sql`这个sql脚本。执行成功之后，就会在我们新建的database下出现以下几张表，里面已初始化了一些数据：



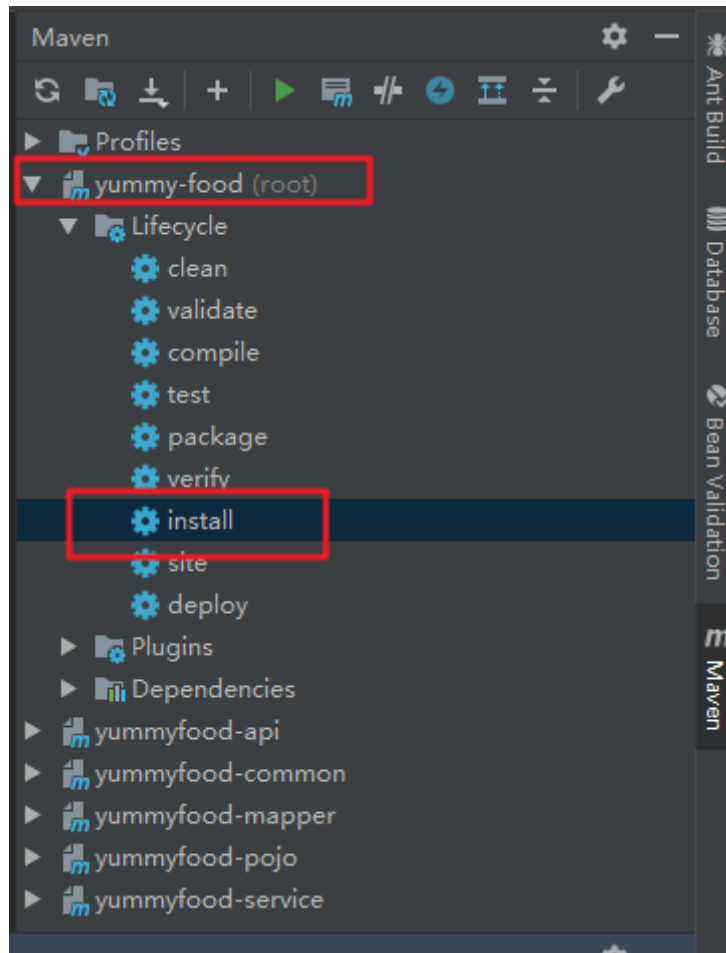
4、项目中关于数据库的配置修改

在`yummyfood-api`工程下找到`application-dev.yml`文件，修改里面关于mysql的相关用户名和密码即可：

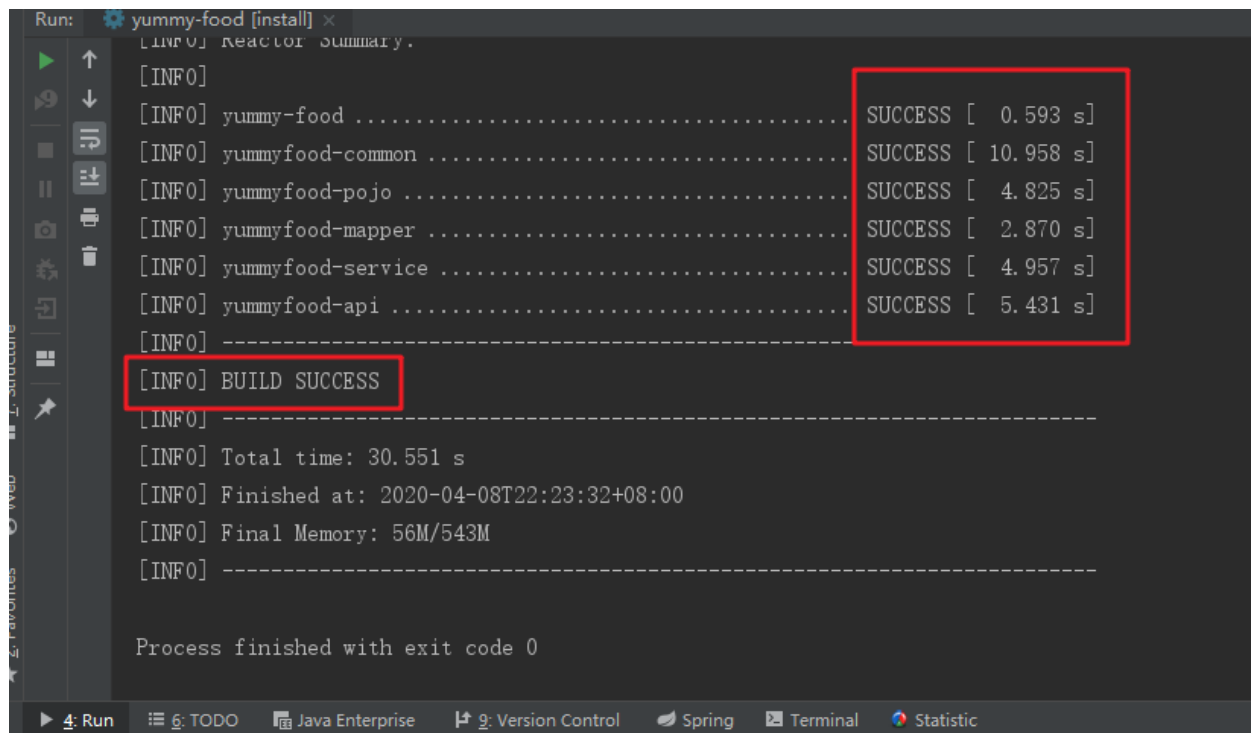


5、启动后端项目

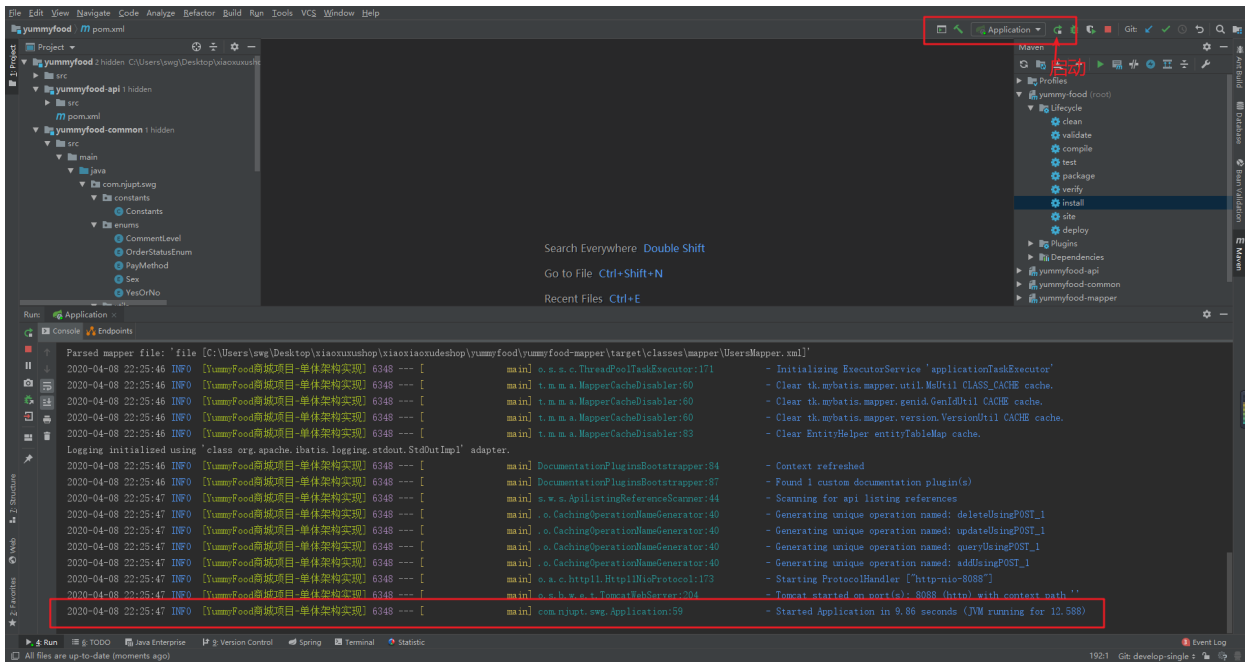
由于是聚合工程，首先需要install，找到聚合工程的父级项目名：



顺利的话，经过一系列的编译、打包之后会跳出build success的字样，说明打包成功了：



此时可直接运行此项目，点击右上角的三角绿色图标即可启动工程，启动成功检查无报错：



此时可以打开浏览器输入一个地址确认接口已正常使用：

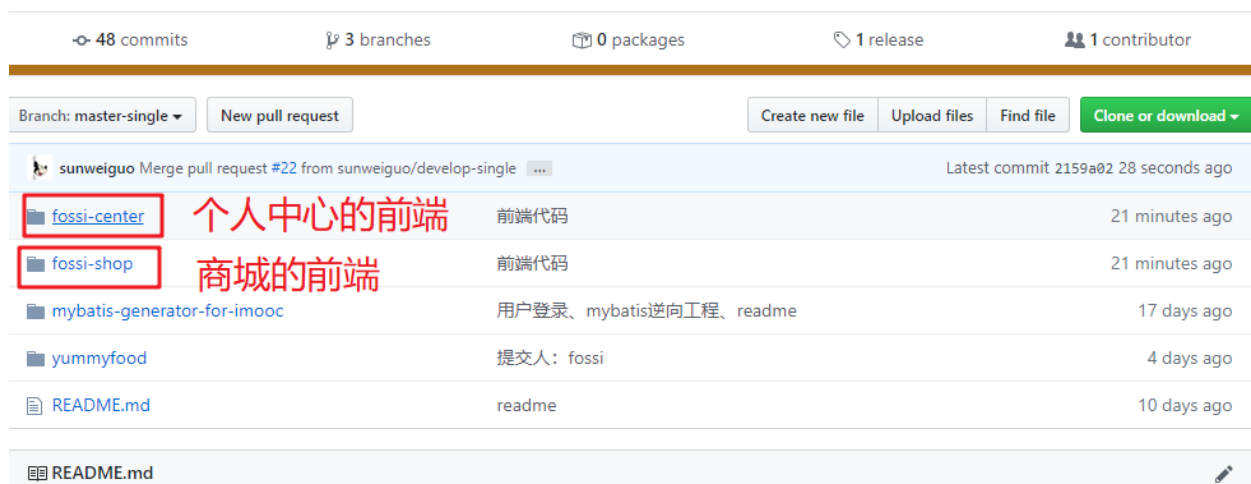
```
1 http://localhost:8088/test/hello
```

看看能不能直接在页面上显示：

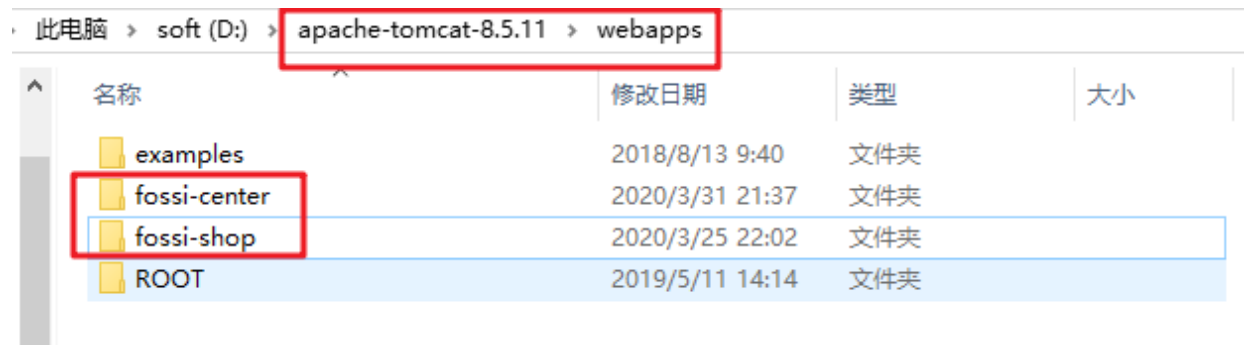
```
1 {
2   "status": 200,
3   "msg": "OK",
4   "data": null
5 }
```

6、启动前端项目

后端项目已正常启动且无报错。那么我们就来看下前端项目如何启动吧。查看刚才clone下来的项目，我们可以看到两个文件夹：



将此文件夹直接拷贝到本机的tomcat的webapps目录下即可：



直接到bin目录下start此tomcat即可，windows下就是双击startup.bat即可，正常启动的黑窗口界面如下所示：

```

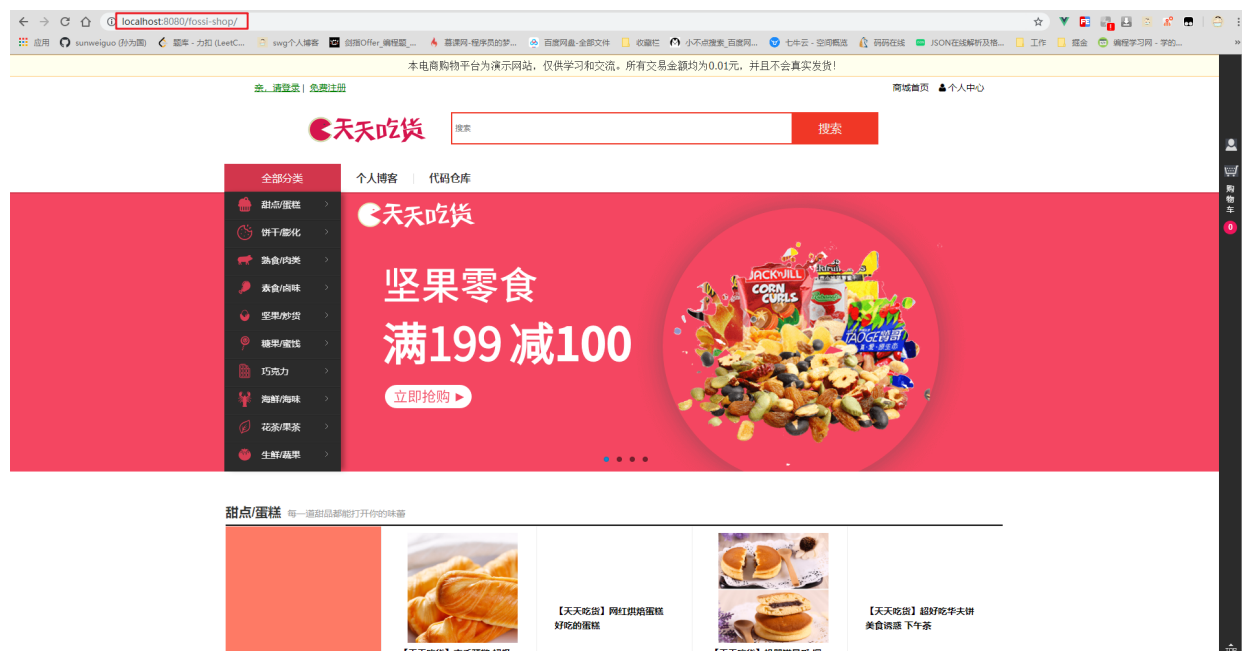
选择Tomcat
22:32:51,033 |-INFO in c.q.l.core.rolling.TimeBasedRollingPolicy@1190876938 - Will use the pattern /permission/error.%d.log for the active file
22:32:51,034 |-INFO in c.q.l.core.rolling.DefaultTimeBasedFileNamingAndTriggeringPolicy - The date pattern is 'yyyy-MM-dd' from file name pattern '/permission/error.%d.log'.
22:32:51,034 |-INFO in c.q.l.core.rolling.DefaultTimeBasedFileNamingAndTriggeringPolicy - Roll-over at midnight.
22:32:51,034 |-INFO in c.q.l.core.rolling.DefaultTimeBasedFileNamingAndTriggeringPolicy - Setting initial period to Wed Apr 08 22:32:51 CST 2020
22:32:51,035 |-INFO in ch.qos.logback.core.rolling.RollingFileAppender[fileErrorLog] - Active log file name: /permission/error.2020-04-08.log
22:32:51,035 |-INFO in ch.qos.logback.core.rolling.RollingFileAppender[fileErrorLog] - File property is set to [null]
22:32:51,036 |-INFO in ch.qos.logback.classic.joran.action.RootLoggerAction - Setting level of ROOT logger to INFO
22:32:51,036 |-INFO in ch.qos.logback.core.joran.action.AppenderRefAction - Attaching appender named [consoleLog] to Logger[ROOT]
22:32:51,038 |-INFO in ch.qos.logback.core.joran.action.AppenderRefAction - Attaching appender named [fileInfoLog] to Logger[ROOT]
22:32:51,038 |-INFO in ch.qos.logback.core.joran.action.AppenderRefAction - Attaching appender named [fileErrorLog] to Logger[ROOT]
22:32:51,038 |-INFO in ch.qos.logback.classic.joran.action.ConfigurationAction - End of configuration.
22:32:51,041 |-INFO in ch.qos.logback.classic.joran.JoranConfigurator@40887df2 - Registering current configuration as safe fallback point

2020-04-08 22:32:51,985 - {dataSource-1} inited
08-Apr-2020 22:32:53.108 信息 [localhost-startStop-1] org.springframework.web.context.ContextLoader.initWebApplicationContext: initialization completed in 4778 ms
08-Apr-2020 22:32:53.170 信息 [localhost-startStop-1] org.apache.catalina.startup.HostConfig.deployDirectory: Deployment of web application directory D:\apache-tomcat-8.5.11\webapps\ROOT has finished in 8,881 ms
08-Apr-2020 22:32:53.176 信息 [main] org.apache.coyote.AbstractProtocol.start: Starting ProtocolHandler [http-nio-8080]
08-Apr-2020 22:32:53.192 信息 [main] org.apache.coyote.AbstractProtocol.start: Starting ProtocolHandler [ajp-nio-8009]
08-Apr-2020 22:32:53.201 信息 [main] org.apache.catalina.startup.Catalina.start: Server startup in 9993 ms
  
```

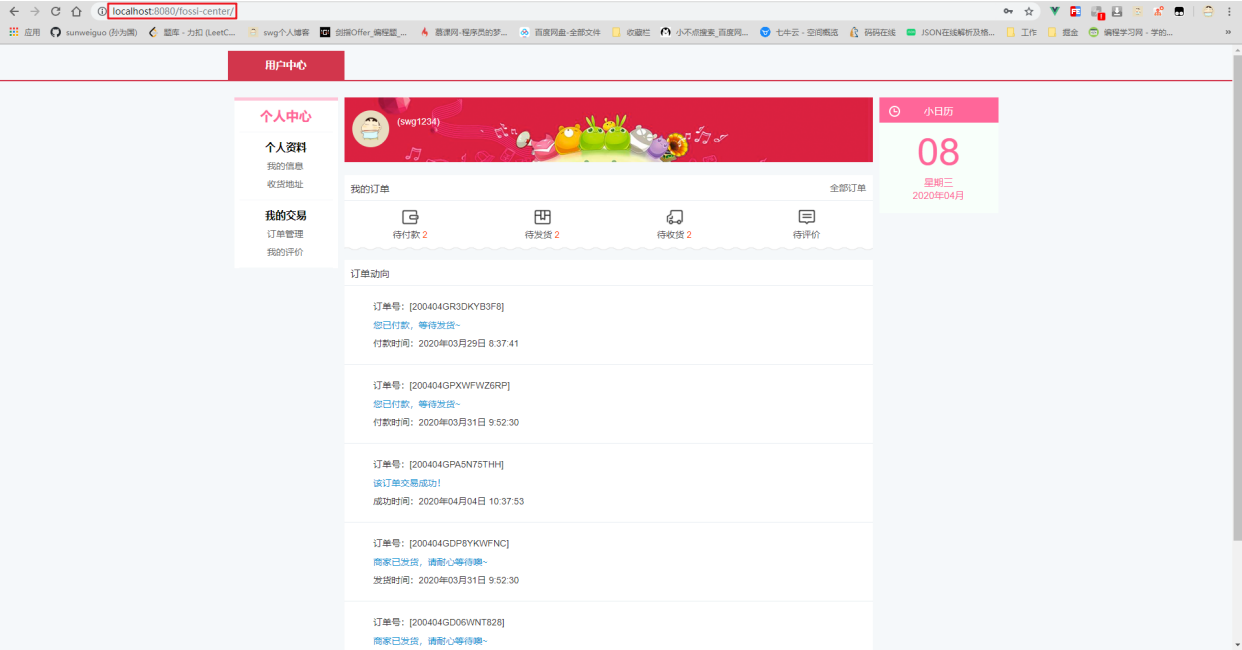
此时浏览器访问：

1 http://localhost:8080/fossi-shop/

即可访问商城的前端界面：



点击用户中心，注册或登录后，即可访问用户的个人中心界面：



尽情地下单吧，不过本地项目还不能支付，先把想吃的食物放进购物车生成订单试试吧~如果您能顺利完成以上所有步骤，那么恭喜你，单体项目将对于你来说毫无压力~~~后续我们详细来看看后端所有的实现逻辑，一步一步搭建起我们第一个商城项目！