

54_停下脚步：先验证一下系统优惠券是否到账？

1、开篇

上节课带大家从打包、发布、启动服务几个步骤将登录功能演示了一遍。通过查看日志的方式，了解了代码从登陆、发送事件消息、接受事件消息，到最后更新登陆次数的执行过程。聊完了登陆之后，本节课我们会梳理互联网教育系统的优惠券管理模块的业务流程，发放优惠券功能会在登陆之后执行，同样也是通过异步的方式执行。今天课程的内容包括以下几个部分：

- 打包服务
- 上传服务
- 启动服务
- 测试代码

2、打包服务

这里直接对代码进行打包。如图 1 所示，打开我们熟悉的酒店管理后台应用服务，在 IntelliJ IDEA 中选择下方的“Terminal”按钮，在显示的命令行中（红色箭头的地方）输入对应的命令，用来对项目进行打包。

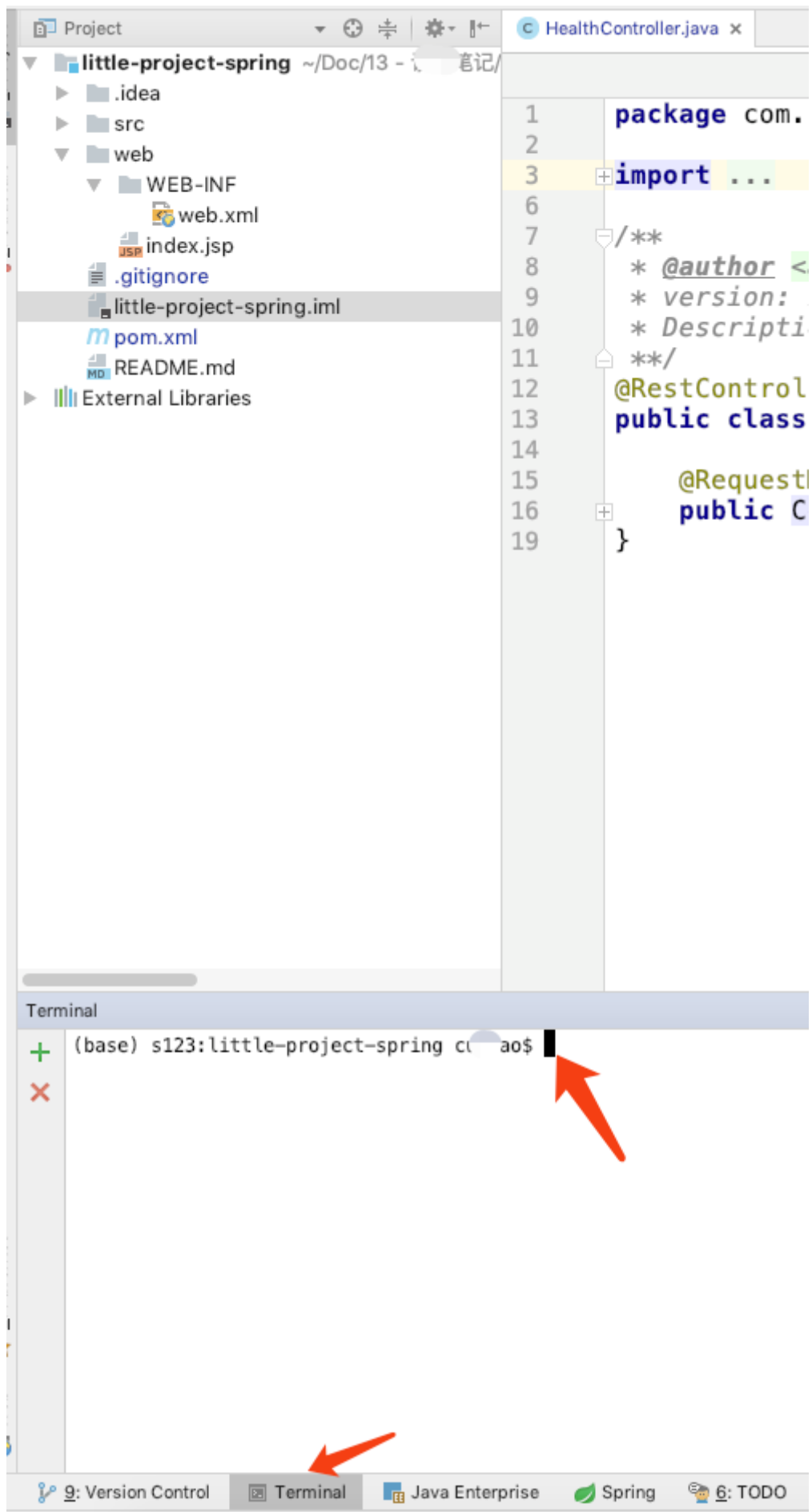


图 1 打开命令行

如图 2 所示，这里输入“mvn clean package”的命令。

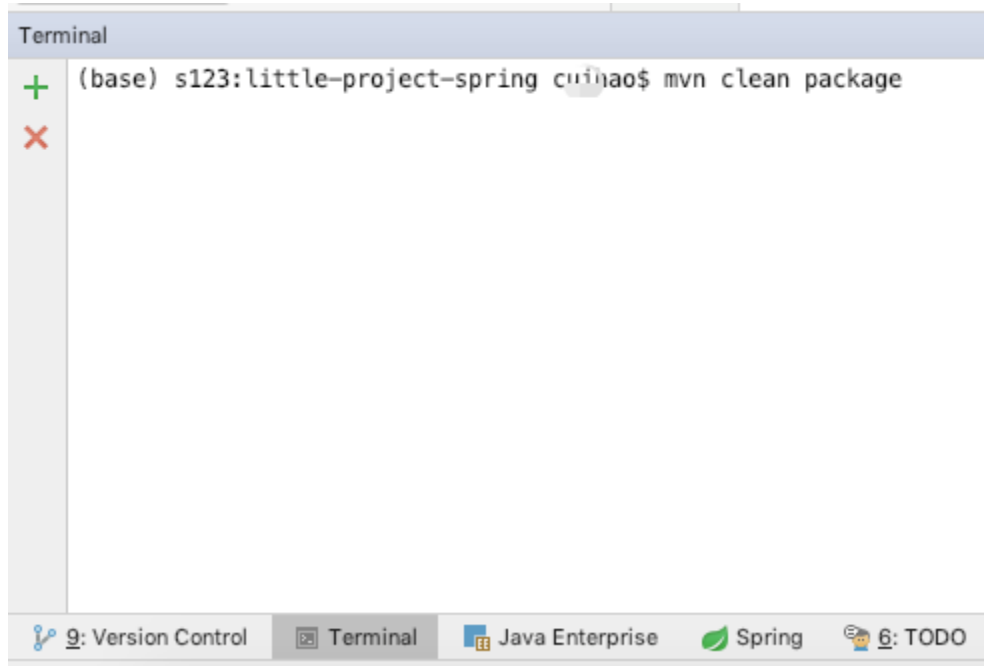


图 2 输入打包命令

在输入命令以后回车，如图 3 所示，会在输出信息中看到“BUILD SUCCESS”的字样表示，打包成功。同时在项目文件的 **target** 目录下面会看到一个“little-project-spring.war”的文件，这就是我们将要发布的 **war** 包。后面的操作会用到它。

Project Explorer:

- little-project-spring ~ /Doc/13 - 读书笔记/
 - .idea
 - src
 - target
 - classes
 - generated-sources
 - little-project-spring
 - maven-archiver
 - maven-status
 - little-project-spring.war**
 - web
 - .gitignore
 - little-project-spring.iml
 - pom.xml
 - README.md
- External Libraries

HealthController.java:

```
1 package com.ruyuan.lit
2
3 import ...
4
5
6
7 /**
8  * @author <a href="ma
9  * version: 1.0
10 * Description:健康检查的
11 */
12 @RestController
13 public class HealthCon
14
15     @RequestMapping(va
16     public CommonRespo
17
18
19 }
```

Terminal:

```
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ li
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /Users/cuihao/Doc/13 - 读 记/石杉/
[INFO] --- maven-compiler-plugin:3.1:testCompile (default-testCompile) @ little-
[INFO] Nothing to compile - all classes are up to date
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ little-project-spr
[INFO] Tests are skipped.
[INFO] --- maven-war-plugin:2.2:war (default-war) @ little-project-spring ---
[INFO] Packaging webapp
[INFO] Assembling webapp [little-project-spring] in [/Users/cu o/Doc/13 - 读书
[INFO] Processing war project
[INFO] Webapp assembled in [201 msecs]
[INFO] Building war: /Users/cuihao/Doc/13 - 读书笔记/石杉/ 战/Spri
[INFO] WEB-INF/web.xml already added, skipping
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 3.912 s
[INFO] Finished at: 2021-05-19T14:50:57+08:00
[INFO] -----
(base) s123:little-project-spring cuihao$
```

图 3 打包成功

3、上传服务

在上传服务（jar 包）之前需要保证，在对应的 ECS 中建立服务运行的目录如下：

`/home/admin/little-project-spring/`

这个目录已经由儒猿团队在 ECS 的镜像中创建好了，大家不用手动创建，我们后面的发布就基于这个目录。其中“/home/admin/little-project-spring”是用来存放部署脚本的，注意我们这里使用的 `deploy.sh` 脚本已经由儒猿团队上传了。

“/home/admin/little-project-spring/”目录是存放服务的 war 包的，这个包是需要我们自己上传的。

在 ECS 上建立好目录结构以后，再回到本地的项目中，依旧是在 IntelliJ IDEA 的命令行中输入以下命令：

```
scp target/little-project-spring.war root@47.117.120.102:/home/admin/little-project-spring /
```

命令的意思是通过 `scp` 命令将刚才打包的“little-project-spring.war”文件 copy 到对应 ECS 服务器的“/home/admin/little-project-spring/”目录中。这里的

“47.117.120.102”是我的测试地址，大家可以更换为自己申请的 ECS 的 IP 地址。

这里需要特别说明一下，由于我在执行命令的根目录在“little-project-spring”项目下面，如果你在其他的地方执行上述两条命令，需要指定好源文件的目录。同时在使用 `scp` 命令以后会让大家输入服务器的密码，该密码可以从实战云平台上获取。完成上面两个命令以后，服务就已经部署到 ECS 了。

4、启动服务

完成部署以后，需要到 ECS 服务器上面启动部署的酒店管理后台的服务。还是登录 ECS 服务器，通过以下命令进入到“`deploy.sh`”文件所在的目录。

```
cd /home/admin/little-project-spring
```

`deploy.sh` 文件是儒猿团队为大家生成的发布的脚本文件，通过 Linux shell 脚本完成发布的参数配置和启动命令。包括启动应用等待的时间、应用端口号、健康检查的 URL 以及 jar 包的目录和日志信息。有兴趣的同学可以打开看看，这里就不做展开的介绍了。

保证当前目录下面存在“deploy.sh”文件，使用如下命令启动服务。

```
sh deploy.sh restart
```

命令使用了“restart”作用与“start”是一致的，用“restart”的目的以免在重复发布过程中，学员忘记是否启动过服务。因此使用“restart”，这样即便是已经启动过服务，也会重新加载服务。

运行命令在看到图 4 所示的“success”字样的时候，就说明服务启动成功了。

```
[root@iZuf64qotsm6gzy9jnkW8cZ little-project-spring]# sh deploy.sh restart
no java process
starting java process
Using CATALINA_BASE:   /home/admin/little-project-spring/apache-tomcat-9.0.45
Using CATALINA_HOME:   /home/admin/little-project-spring/apache-tomcat-9.0.45
Using CATALINA_TMPDIR: /home/admin/little-project-spring/apache-tomcat-9.0.45/temp
Using JRE_HOME:        /usr
Using CLASSPATH:        /home/admin/little-project-spring/apache-tomcat-9.0.45/bin/bootstrap.jar:/home/admin/little
-project-spring/apache-tomcat-9.0.45/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
started java process
checking http://127.0.0.1:8090
Wait app to pass health check: 1...
code is 200
check http://127.0.0.1:8090 success
```

图 4 启动服务成功。

5、登陆授权

完成上述操作以后，切换到互联网教育系统的小程序上面。如图 5 所示，打开小程序在弹出的登陆授权框中选择“登陆”按钮并且点击。

07:36



首页



请输入教师名称搜索



销量↓

评分↓

价格↓



我们需要您的授权

登录后才能为您提供更多的服务

取消

登录

销量66

评分4.62

价格¥ 160

专注于6-10岁儿童教育

王五

英语

销量55

评分4.41

价格¥ 190

专注于6-10岁儿童教育

李四

数学

销量44

评分4.88

价格¥ 200

专注于6-10岁儿童教育



首页



我的优惠券



我的

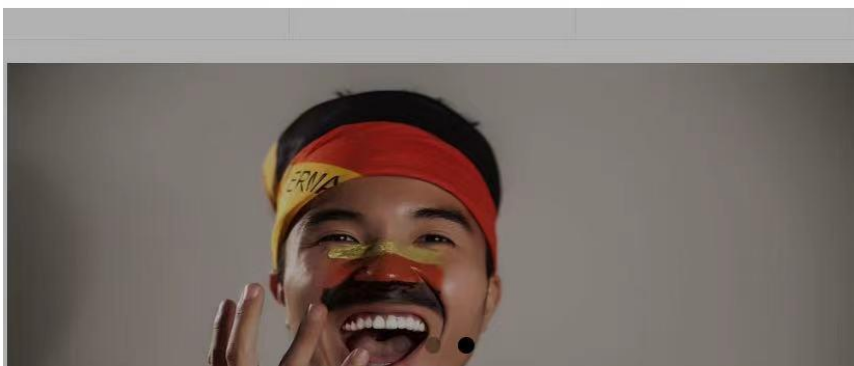
图 5 登陆授权

完成登陆授权以后，如图 6 所示，在弹出的手机授权中点击“手机号”，会让你选择手机号并且授权。

07:37



首页



钱老师

化学

价格 ¥ 205



160



190

专注于6-10岁儿童教育



李四

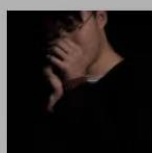
数学

销量44

评分4.88

价格 ¥ 200

专注于6-10岁儿童教育



孙老师

生物

销量41

评分4.51

价格 ¥ 193

专注于6-10岁儿童教育



首页



我的优惠券



我的

图 6 手机号授权

6、测试代码

登陆授权的完成标志着数据库中已经新增了消费者的登陆记录，由于我们这里需要测试登陆优惠券和生日优惠券的发放功能，因此需要到 ECS 的数据库中造一些数据协助这次测试。

在登陆 ECS 以后输入以下代码段用来登陆 MySQL 数据库。

```
mysql -uroot -p
```

输入登陆 MySQL 的登陆命令以后会弹出“Enter Password”的字样，随后输入“root”回车，我们给大家分配到 ECS 上的 MySQL 数据库的密码是“root”。

如图 7 所示，在输入 MySQL 密码以后会显示如下信息表示登陆成功。

```
[root@iZuf62wm67bh2kd11bmk72Z little-project-spring]# mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.6.51 MySQL Community Server (GPL)

Copyright (c) 2000, 2021, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

图 7 登陆到 MySQL

如图 8 所示，此时通过“select * from spring.t_consumer;”查询消费者的状态，发现刚刚微信登陆的用户以及登陆的次数 login_times 为 1。

```
mysql> select * from spring.t_consumer;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| id   | consumer_name | phone   | credits | sex | birthday | order_times | total_prices | open_id | email | address | login_times |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 14876 | 微信用户     | 1582...76 | 0       | 0   | NULL     | NULL        | NULL        | NULL   | NULL | NULL    | 1           |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

图 8 t_consumer 表中的信息

如图 9 所示，再回到微信小程序中“我的优惠券”中可以看到“首次登陆优惠券”。

07:43



我的优惠券



首次登录优惠券

是否使用：未使用

减免金额：20

领取时间：2021-06-02 07:20:36

过期时间：2021-06-12 23:59:59



首页



我的优惠券



我的

图 9 我的优惠券

如果首次登陆的优惠券被使用以后，还想测试首次登陆发放优惠券的功能，可以通过以下 SQL 语句对表 `t_consumer` 进行更新，将 `login_times` 设置为 0，然后再登陆就可以再次收到优惠券了。

- 重置首次登录手动执行 SQL: `update spring.t_consumer set login_times = 0 where phone = 手机号;`

除了测试首次登陆优惠券发放功能以后，这里还要测试一下生日优惠券发放的功能。由于系统没有入口修改消费者的生日信息，因此需要修改数据库中 `t_consumer` 表的内容。我将 SQL 修改的内容放到下面，供大家参考。由于生日优惠券发放是按照月份匹配的，我这里设置为 6 月的生日，大家可以根据测试代码时当前所在的月份进行设置。

- 修改用户生日手动执行 SQL: `update spring.t_consumer set birthday = '1990-06-01' where phone = 手机号;`

修改好生日以后再次尝试登陆系统，如图 10 所示在“我的优惠券”菜单里面会看到“生日券”的信息。

09:02



我的优惠券



生日券

是否使用：未使用

减免金额：15

领取时间：2021-06-02 09:02:07

过期时间：2021-07-02 23:59:59

首次登录优惠券

是否使用：未使用

减免金额：20

领取时间：2021-06-02 07:20:36

过期时间：2021-06-12 23:59:59



首页



我的优惠券



我的

图 10 生日券

7、总结

本节课将添加了优惠券发放的代码进行了测试，通过修改数据库的方式重现优惠券发放的功能。本节课也是这周的最后一节课，这里对一周的课程做一个总结。首先从登陆功能切入，介绍互联网教育系统的登陆业务流程，通过 **Spring Event** 的方式将登陆流程进行异步处理。在登陆的基础上引入发放优惠券的功能，包括：首次登陆优惠券和生日优惠券两个部分的内容。下周会进入教师模块的开发，会涉及到 **Spring Cache** 解决高并发查询的问题。下期见，拜拜。