

53_代码实战：尝试给首次登录用户发放优惠券和生日券

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：

- 定义 ConsumerCoupon 实体类
- 定义 ConsumerMapper 访问数据库
- 修改 LogonListener
- 定义 ConsumerController

2、定义 ConsumerCoupon 实体类

既然是对优惠券的操作那么首先需要定义一个 Coupon 类，如图 1 所示，在 spring/dto 下面建立 Coupon 类，主要用来描述优惠券的基本信息。包括优惠券 Id、优惠券名称、状态、有效天数、金额、优惠券总数。

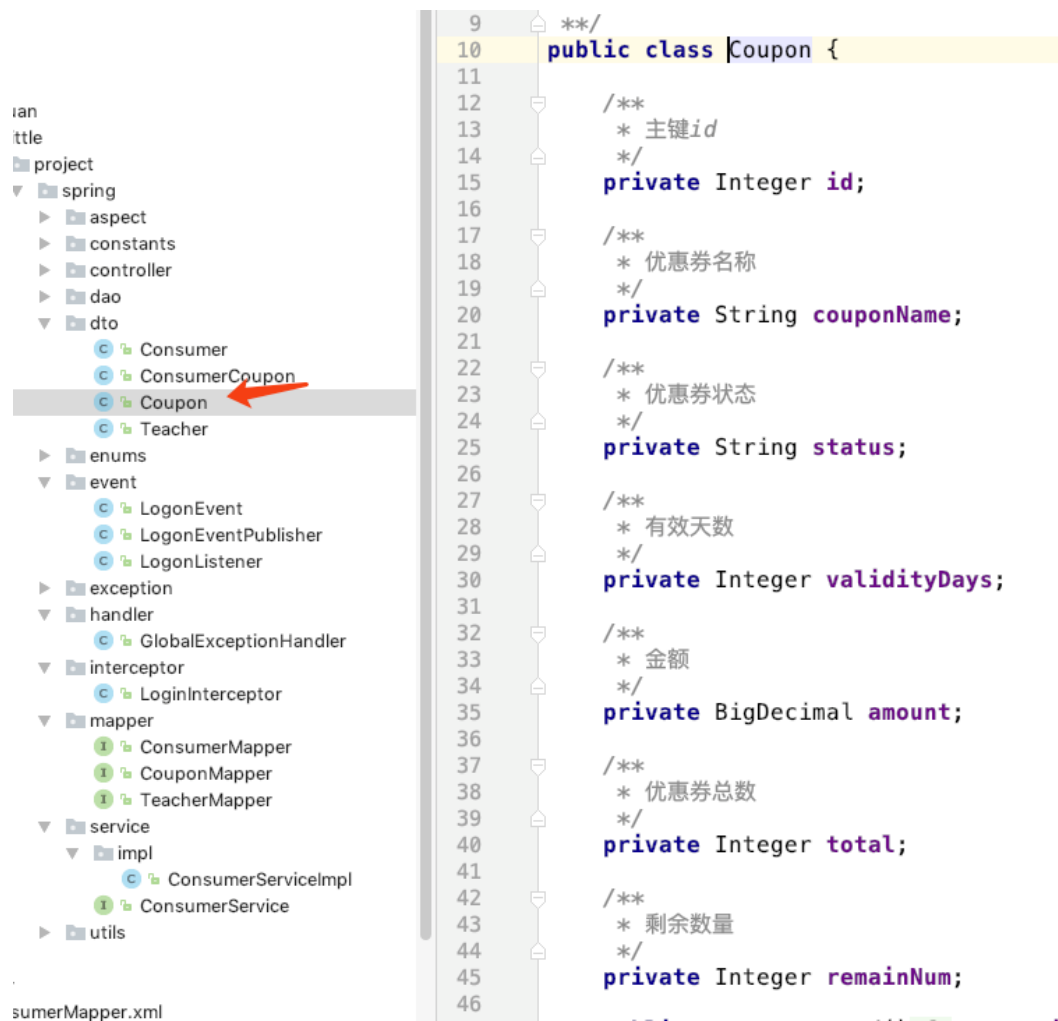


图 1 Coupon 类定义

定义完优惠券的基本信息以后，还需要定义优惠券与消费者的关联信息，毕竟优惠券如果要起到作用需要和消费者产生关系。如图 2 所示，在 `spring/ConsumerCoupon` 下面定义 `ConsumerCoupon` 类，该类用来描绘优惠券和消费者之间的关系。包括消费者 `Id`、优惠券 `Id`、消费者名称、优惠券名称、状态、领取金额、领取时间和失效时间等。

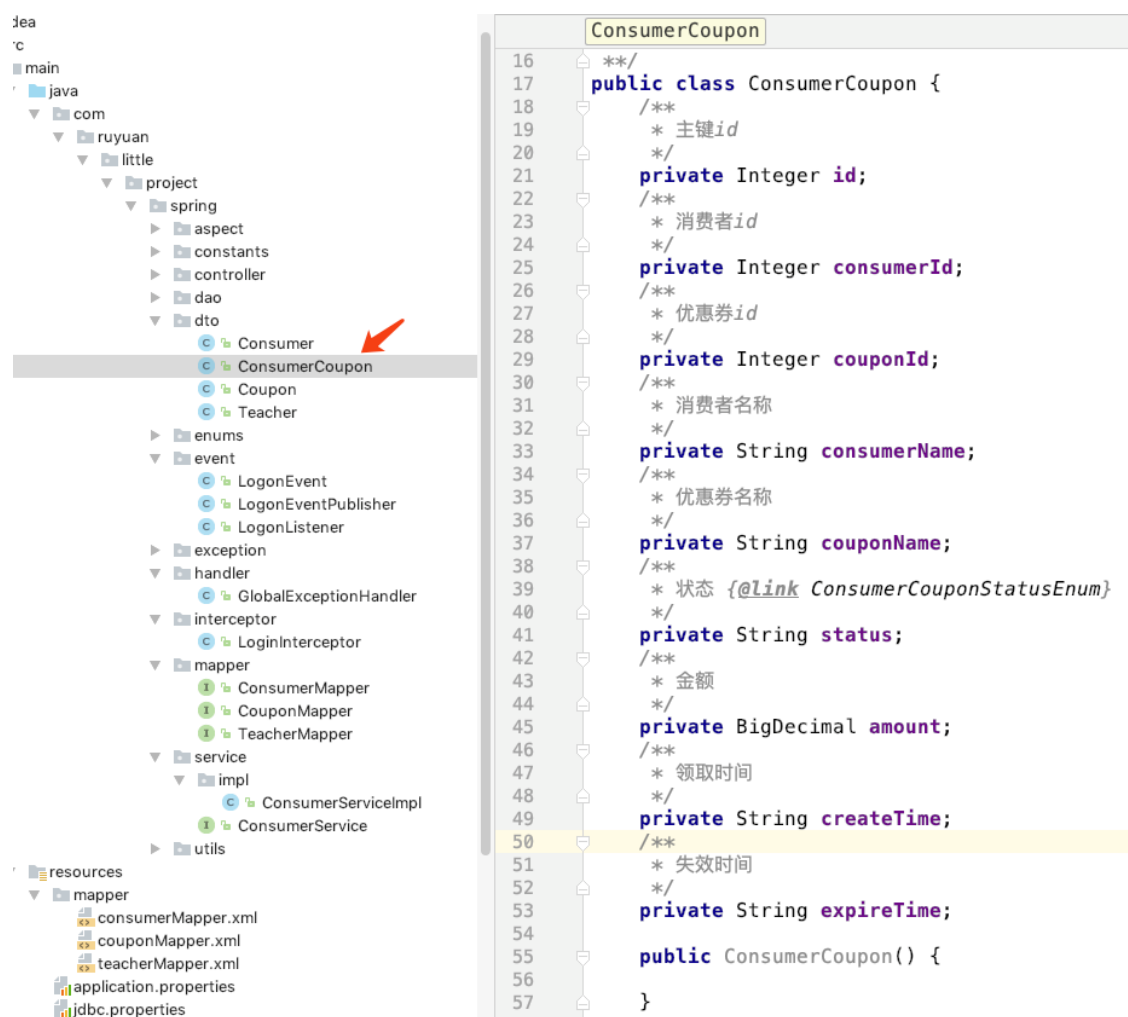


图 2 优惠券与消费者关联关系

3、定义 ConsumerMapper 访问数据库

定义完优惠券和优惠券关联关系之后，就需要对优惠券进行数据库的操作。这里需要定义 Mybatis 的数据库访问接口和操作数据库的 Mapper 文件。先来描述数据库的访问接口，它是应用程序调用 Mybatis 的入口。如图 3 所示，在 `spring/mapper` 下面建立 `CouponMapper` 的接口文件，由于它是一个接口文件因此在其中定义的都是接口方法。包括添加优惠券信息、消费者获取优惠券、通过消费 Id 和时间查找优惠券以及更新优惠券信息。

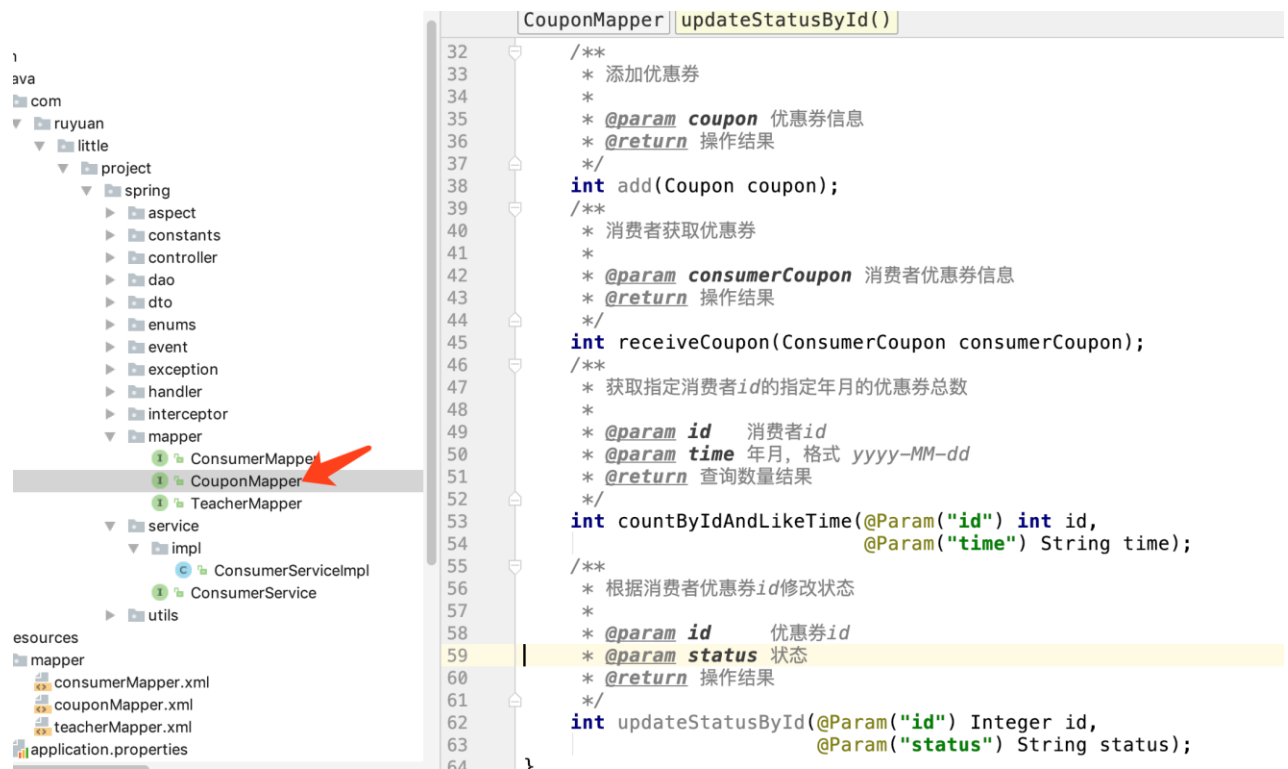


图 3 定义 CouponMapper 的接口文件

由于 CouponMapper 是一个接口文件，为了实现优惠券信息的真实访问需要实体类去实现该接口。如图 4 所示，在 resources/mapper 目录下面定义一个 couponMapper.xml 文件，这个文件就完成 CouponMapper 接口的具体实现的。其中可以看到 findById 方法，就是通过传入的优惠券 Id 到数据库中的 t_coupon 表中查询优惠券信息的。

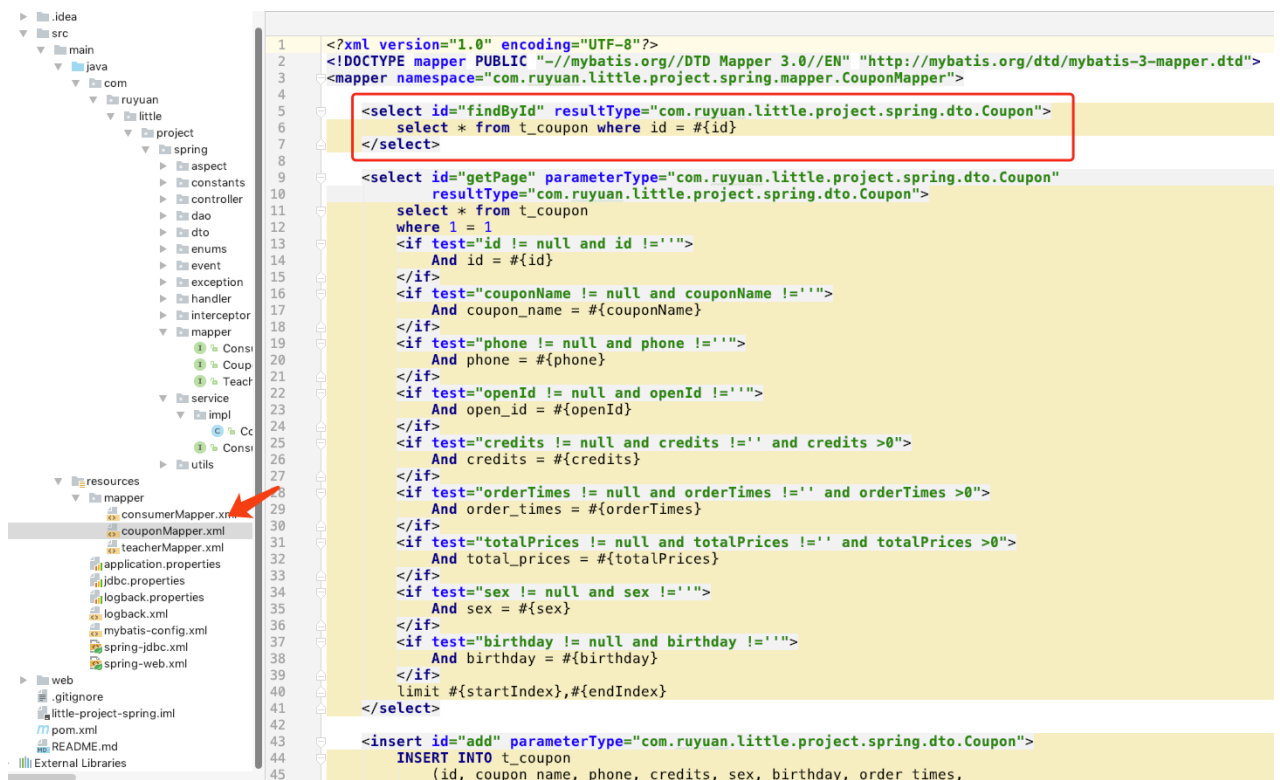


图 4 couponMapper.xml 根据优惠券 Id 查找优惠券信息

既然能够查询到优惠券，那么也可以将某个优惠券与消费者进行关联。如图 5 所示，依然是在 couponMapper.xml 中定义了一个 receiveCoupon 的实现，实际上就是一个 insert 语句，在 t_consumer_coupon 这张表中插入一条记录。从这个表的名字上看，这个就是记录优惠券和消费者之间的关联表。插入的信息包括消费者 Id、优惠券 Id、优惠券名称、消费者名称等相关信息。

```

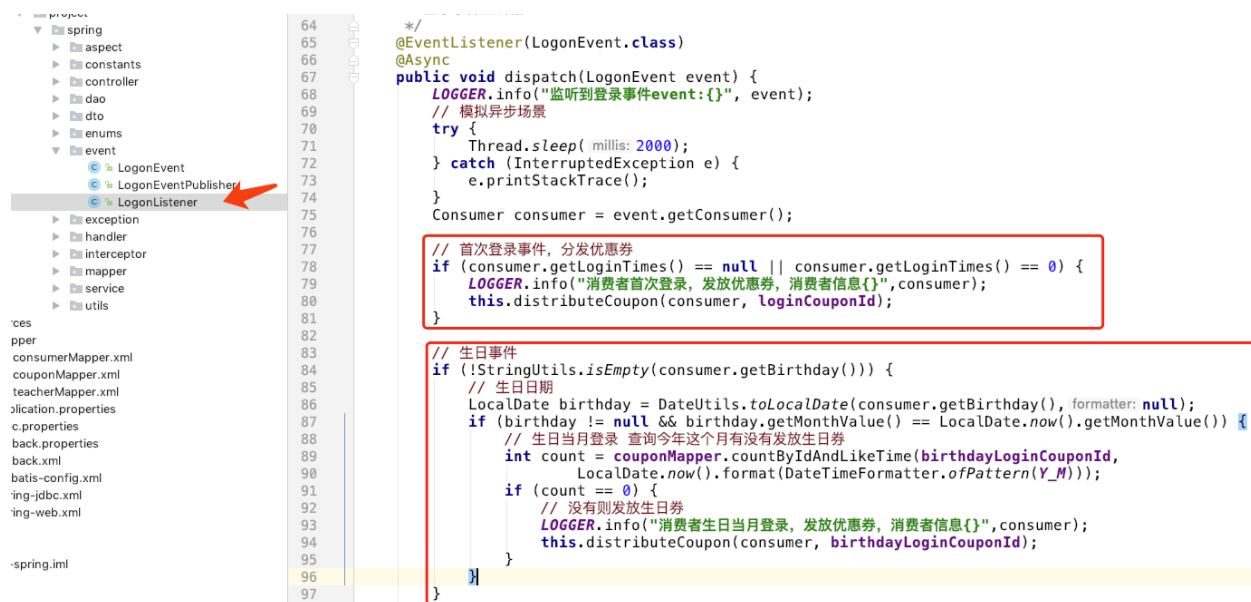
<insert id="receiveCoupon" parameterType="com.ruyuan.little.project.spring.dto.ConsumerCoupon"
    useGeneratedKeys="true" keyProperty="id">
    INSERT INTO
    t_consumer_coupon(
    consumer_id,
    consumer_name,
    coupon_id,
    coupon_name,
    status,
    amount,
    create_time,
    expire_time)
    values(
    #{consumerId},
    #{consumerName},
    #{couponId},
    #{couponName},
    #{status},
    #{amount},
    #{createTime},
    #{expireTime})
</insert>

```

图 5 couponMapper.xml 让消费者获取优惠券

4、修改 LogonListener

说完了优惠券相关的实体以及数据库访问代码之后，在来看看在哪里需要调用优惠券方法的代码。从前面几节课可以知道，优惠券的发放是伴随在登陆之后，通过异步的方式进行的。在系统中监听登陆消息的代码就在 LogonListener 中，这里对其进行修改。如图 6 所示，修改 LogonListener 类中的 dispatch 方法，加入登陆发放优惠券和生日优惠券的代码。其中登陆优惠券的代码需要判断是否第一次登陆，如果是第一次就调用 distributeCoupon 方法发放优惠券。对与生日优惠券而言，如果消费定义了生日信息，首先判断在生日当月是否已经领取了优惠券，如果没有领取对其进行优惠券发放的操作。



```

/**
 * 分发优惠券
 *
 * @param consumer 消费者
 * @param couponId 优惠券id
 */
private void distributeCoupon(Consumer consumer, Integer couponId) {
    Coupon coupon = couponMapper.findById(couponId);
    ConsumerCoupon consumerCoupon = new ConsumerCoupon(consumer, coupon);
    LOGGER.info("发放优惠券{}", consumerCoupon);
    couponMapper.receiveCoupon(consumerCoupon);
}

```

图 7 distributeCoupon 方法

5、定义 ConsumerController

除了优惠券发放的代码之外，还需要有查询优惠券的接口。在领取优惠券以后，用户依旧可以通过 ConsumerController 对优惠券信息进行了查询。如图 8 所示，在 spring/controller 下面建立 ConsumerController 类分别建立 findCouponByConsumerId 方法传入优惠券 Id 对优惠券进行查找；findCouponByConsumerIdAndStatus 通过消费者 Id 和状态进行查找等方法。



图 8 定义 ConsumerController

6、总结

本节课围绕优惠券发放的代码实现展开，介绍了 **ConsumerCoupon** 实体类、**ConsumerMapper** 访问数据库的定义、以及修改 **LogonListener** 加入发放优惠券的代码、最后是 **ConsumerController** 的定义。下节课我们会带大家代码进行测试。

这里将[代码](#)给大家，下期见，拜拜。

友情提示：本章讲述代码只是部分核心代码，完整代码请查阅文末链接中代码，谢谢。