1. Src下的java文件存放的我们后端的代码
2. Controller主要是调用了service层的方法，和一些业务逻辑的编写，我们可以根据具体方法名称看出来这个方法具体的功能，比如AdminController，里面的update方法就是对管理员信息的修改，toLogin跳转到管理员登录页面，login对前端传来参数的判断，成功或失败都跳转到对应的页面，logout退出登录，下面的controller我就不一一列举了，根据方法名就可以判断具体功能
3. Dao下面都是一些接口，里面的方法都是针对数据库操作的，他跟下面resource下的mapper里的mapper.xml是对应的，mapper.xml里都是sql语句，dao里面的每个方法都在mapper.xml有具体对数据库操作的实现
4. Entity里都是我们封装的实体类，基本上一个数据库的表对应着一个实体类，其他的比如page和pages是分页功能需要的实体类，ShoppingCart是用户在下单后存放餐车需要的实体类
5. Interceptor里这两个类起到一个拦截的功能，分别对前台未登录的用户和后台未登录的用户进行拦截并拒绝访问的操作
6. 再往下的service，它里面有一些接口和这些接口的实现类，这些接口基本上都是一些增删改查的方法，然后实现类来做具体的实现，这些实现类都导入了dao层的接口，调用dao层的方法来对数据库进行具体的操作
7. 下的utils就是我们的工具类了
8. Resource文件存放这我们的配置文件和mapper.xml的代码
9. ApplicationContext.xml文件主要配置我们spring注解的包扫描注释，数据库连接的配置，dao层接口的配置，对事物和aop的配置
10. Log4j就是日志的配置
11. springMVC.xml主要配置了我们静态资源访问路径，controller层的扫描还有加载视图解析器和文件解析器
12. sqlMapConfig.xml主要配置了实体类的包扫描和dao层的包扫描
13. webapp主要存放着我们前端的资源，下面的public里的admin是我们后台页面需要的资源，images包含着前台页面菜品的图片，qiantai文件下都是我们前台页面需要的资源了
14. 最后web-inf里的web.xml主要配置了listner，servlet和解决中文乱码的过滤器