小米汽车电池预警大作业开发流程及结果展示

一、创建项目及初始化配置

1、IDEA新建SpringBoot项目

技术栈：JDK1.8、SpringBoot 2.2.1.RELEASE。

1. 项目初始化配置

（1）编码：将控制台编码、文件编码更改为UTF-8；

（2）设置热部署与自动编译。

1. 项目测试结果

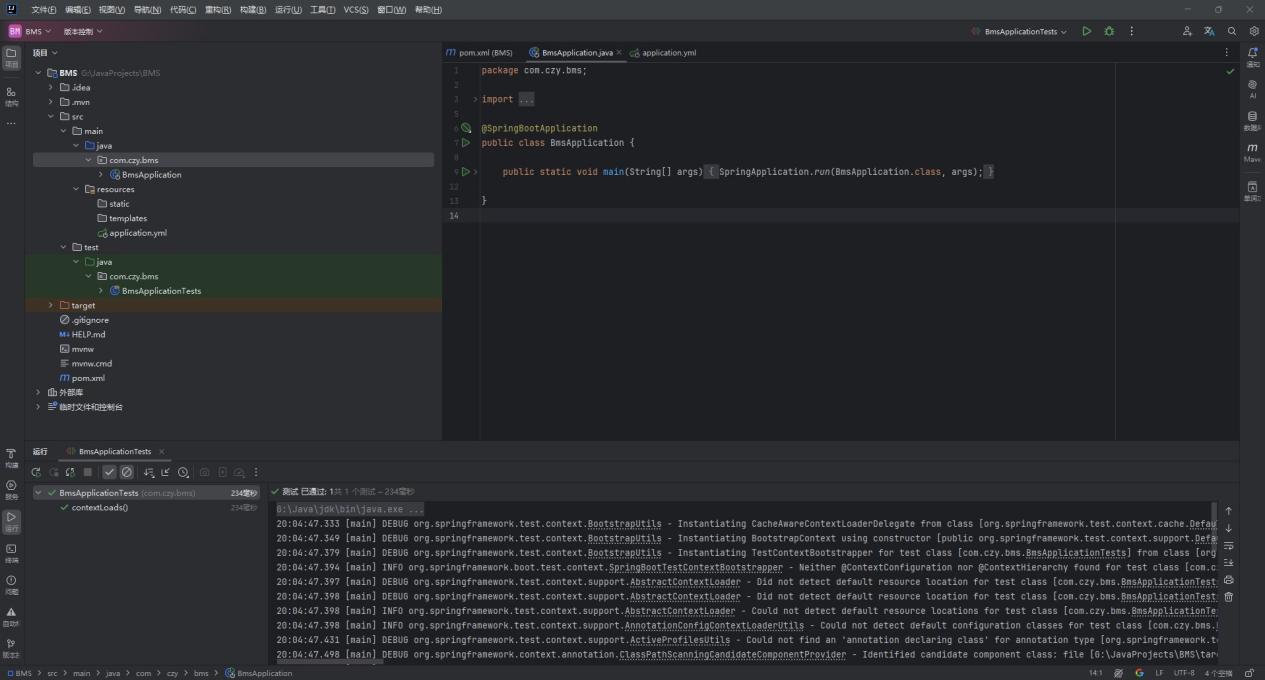


图1.1 项目搭建测试结果图

1. 代码管理GitHub远程仓库
2. 编写TestController测试类
   1. 在application.yml文件中配置端口好和项目名称；
   2. 在TestController类上添加注解@RestController和@RequestMapping("/test")；
   3. 编写测试方法返回字符串结果。
3. TestController类测试结果

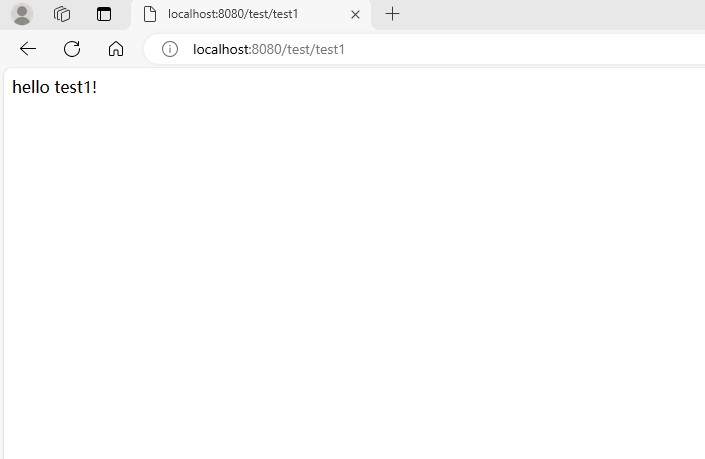


图1.2 TestController类测试结果图

1. 使用HTTP Client完成接口测试

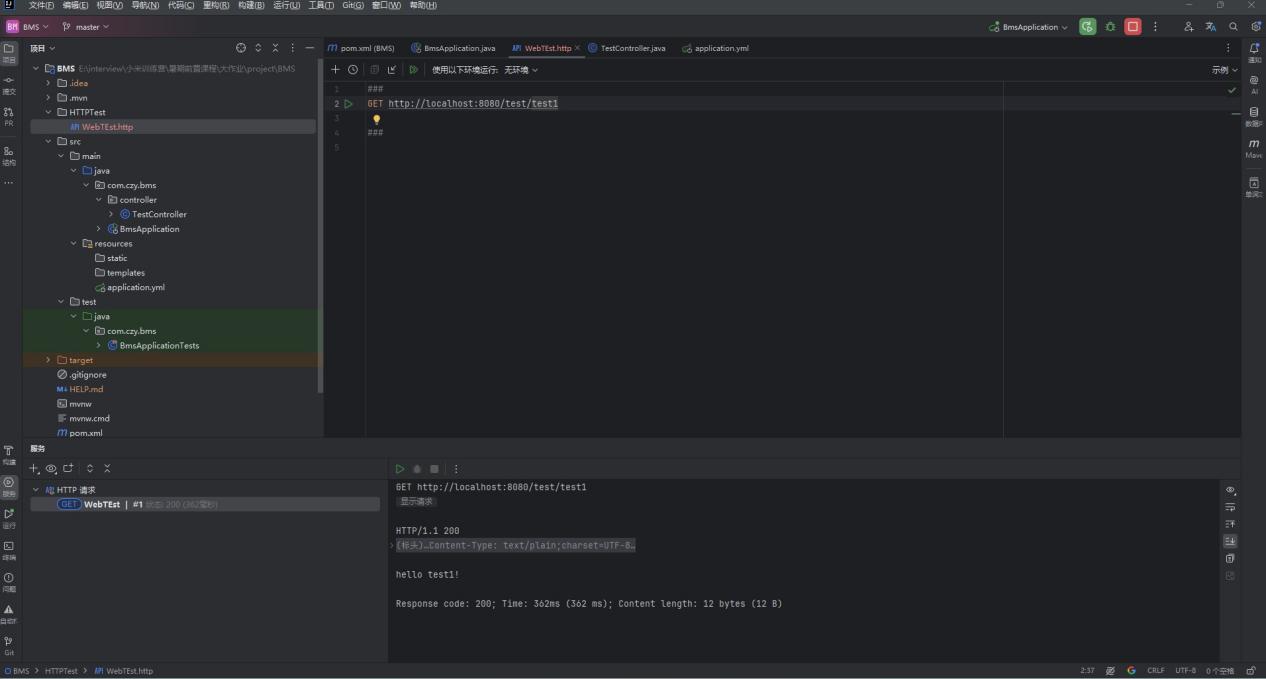


图1.3 HTTP Client接口测试结果图

二、封装返回结果

1、引入Lombok依赖

2、创建response包以及CommonResp类

3、在CommonResp类中定义状态码、信息以及数据

4、在TestController类中编写方法进行测试

5、测试结果

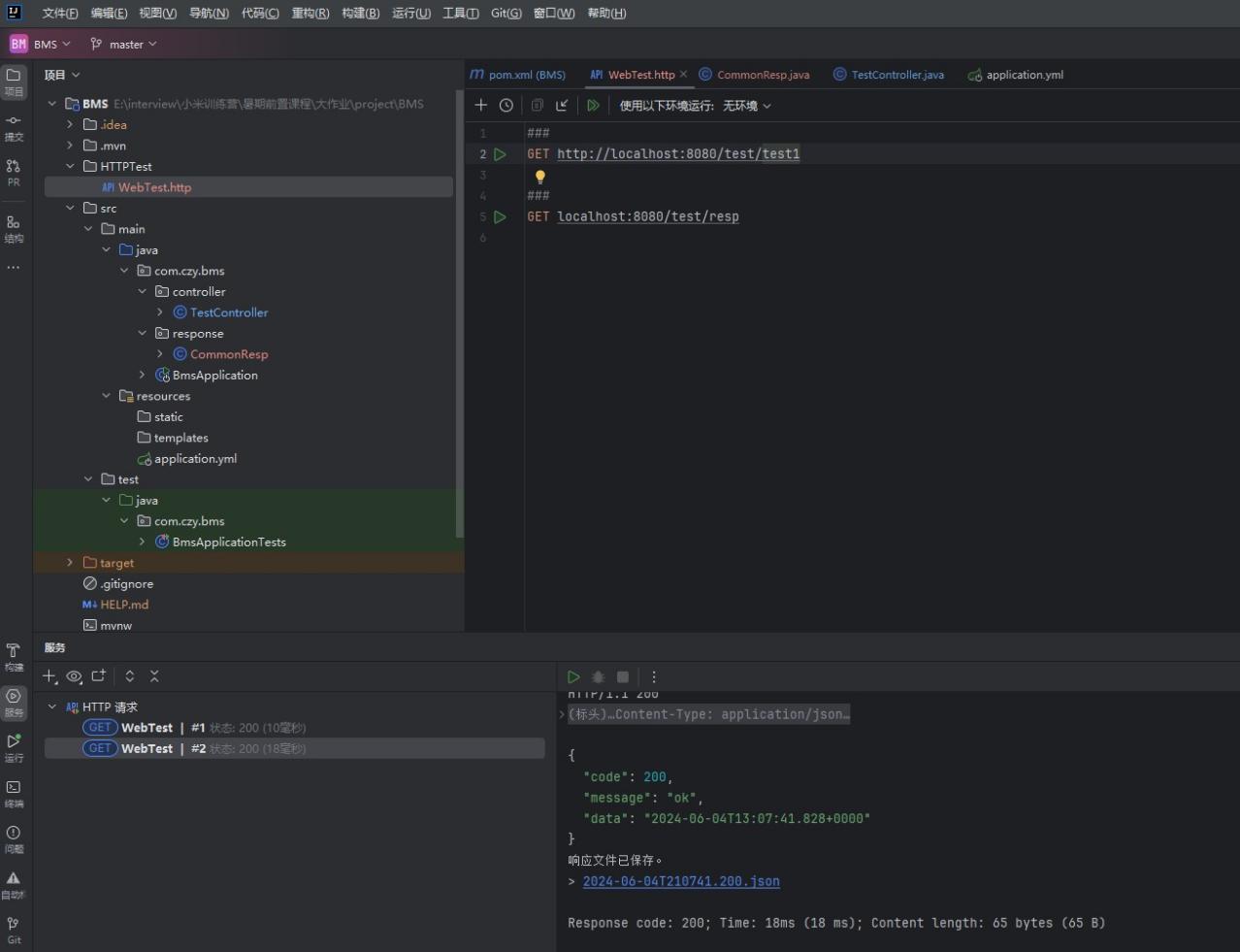


图2.1 封装返回结果测试结果图

1. 配置时间格式及时区
2. 测试结果

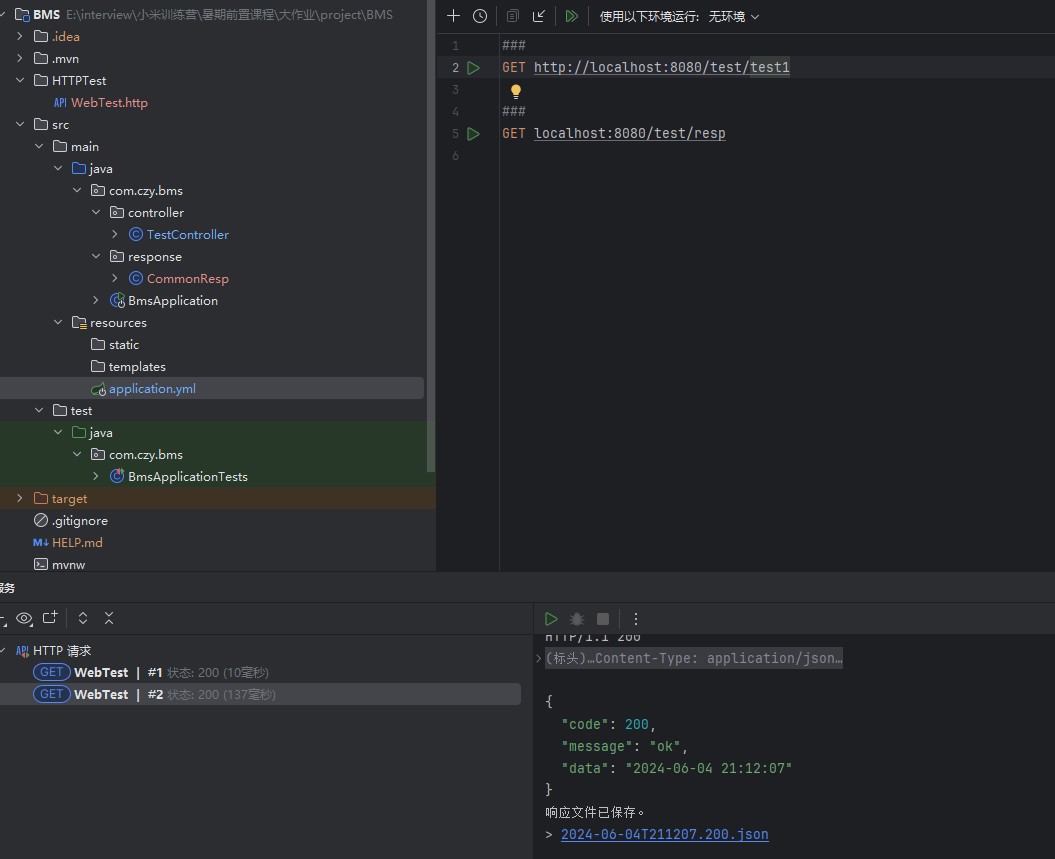


图2.2 时间格式及时区测试结果图

1. 测试以json数组的形式返回数据
   1. 添加json-simple依赖
   2. 测试结果

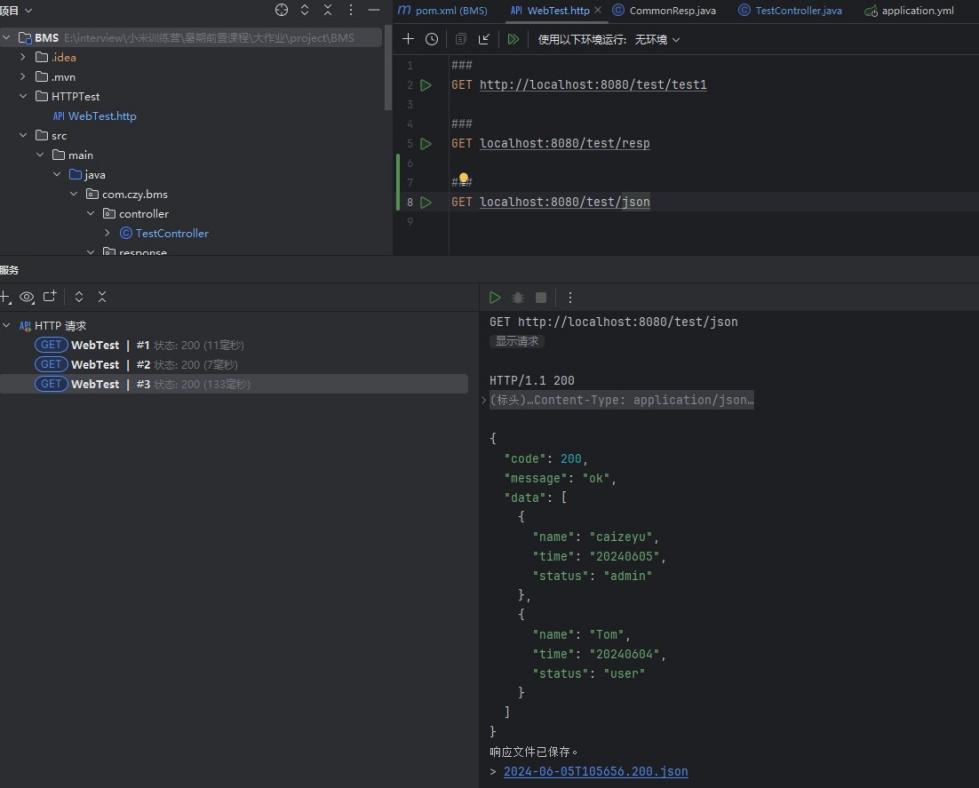


图2.3 以json数组格式返回数据测试结果图

三、连接数据库

1、创建数据库

2、IDEA连接数据源

3、测试数据源连接

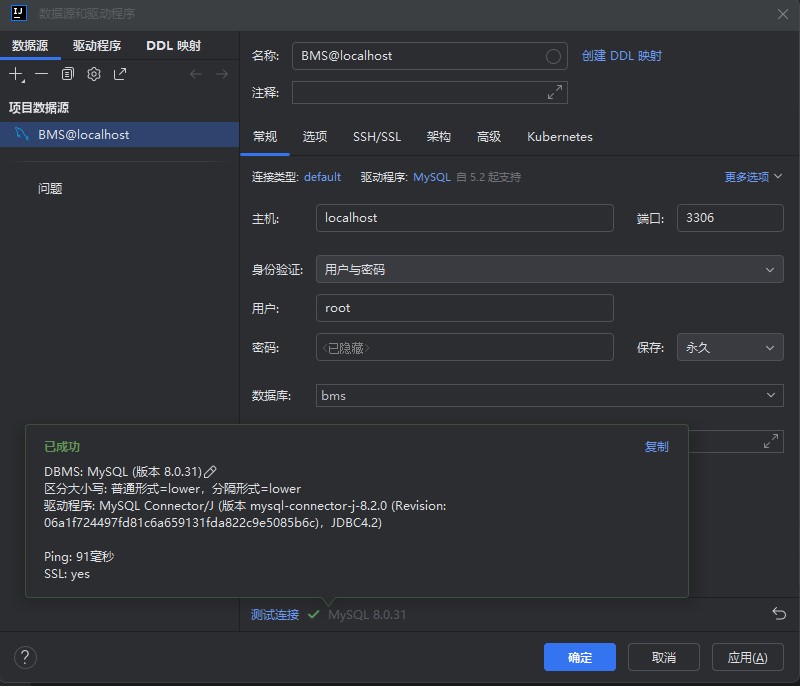


图3.1 IDEA连接数据源测试结果图

4、运行SQL脚本

（1）汽车表vehicle：



图3.2 创建汽车表

（2）规则表battery\_rules

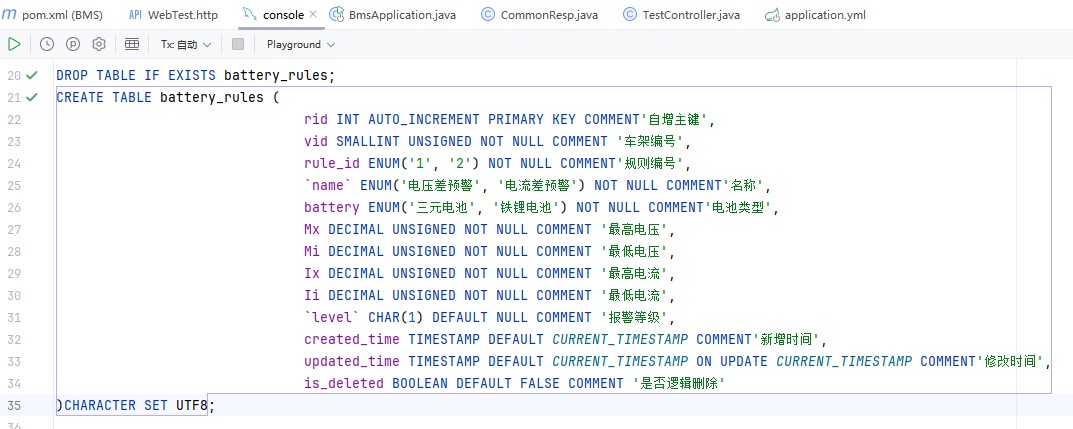


图3.3 创建规则表

四、集成Mybatis

1、引入mysql和mybatis依赖

2、配置数据源

3、定义持久层

（1）mapper包下的VehicleMapper接口

（2）resources/mapper文件夹下的VehicleMapper.xml文件

（3）在application.yml文件中配置classpath

（4）在启动类中添加@MapperScan注解

4、定义应用层

（1）service包下的VehicleService接口

（2）sevice/Impl文件夹下的VehicleServiceImpl类

5、定义控制层、编写测试方法，查询Vehicle表中的记录数量

6、测试结果

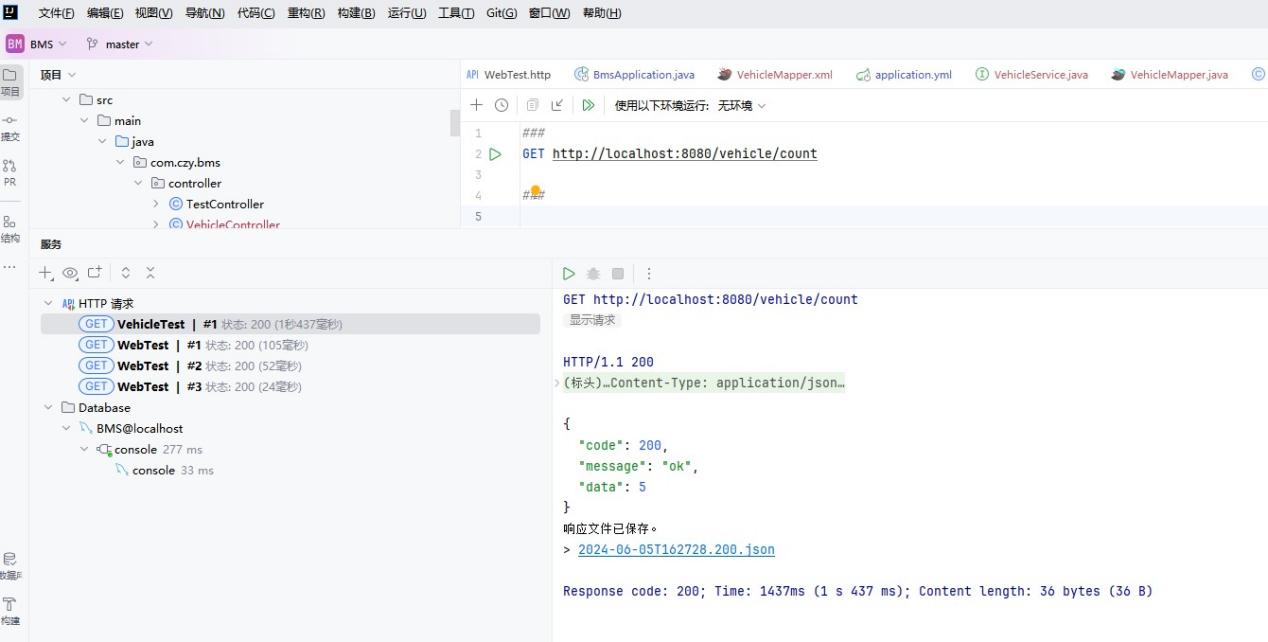


图3.4 Mybatis集成测试结果