

43_思考一下 JDBCTemplate 操作数据库存在的问题

1、开篇

上节课介绍了 JDBCTemplate 以及它对应的几种数据库操作方法，然后在互联网教育系统中加入了 TemplateBaseDao 类用来执行 JDBCTemplate 的相关操作。本节课，我们会尝试运行代码看看结果，并且思考一下 JDBC 还存在哪些问题，有哪些办法可以进行提升的。要介绍的内容包括：

- 执行 JDBCTemplate 的例子查看结果
- JDBCTemplate 还存在哪些问题

2、执行 JDBCTemplate 的例子查看结果

依旧是打包、发布、并且启动服务。直接通过服务的 Controller 作为入口对服务进行测试。如图 1 所示，通过 URL 地址 <http://106.15.94.79:8090/demo/template/all> 访问具体方法。随着在 postman 中点击“Send”按钮之后显示出查询结果。

在结果中显示出教室信息列表，其中包括教师 id、教师姓名、课程、分数、教学天数、教学次数、每天教学费用、以及描述和教师照片地址。

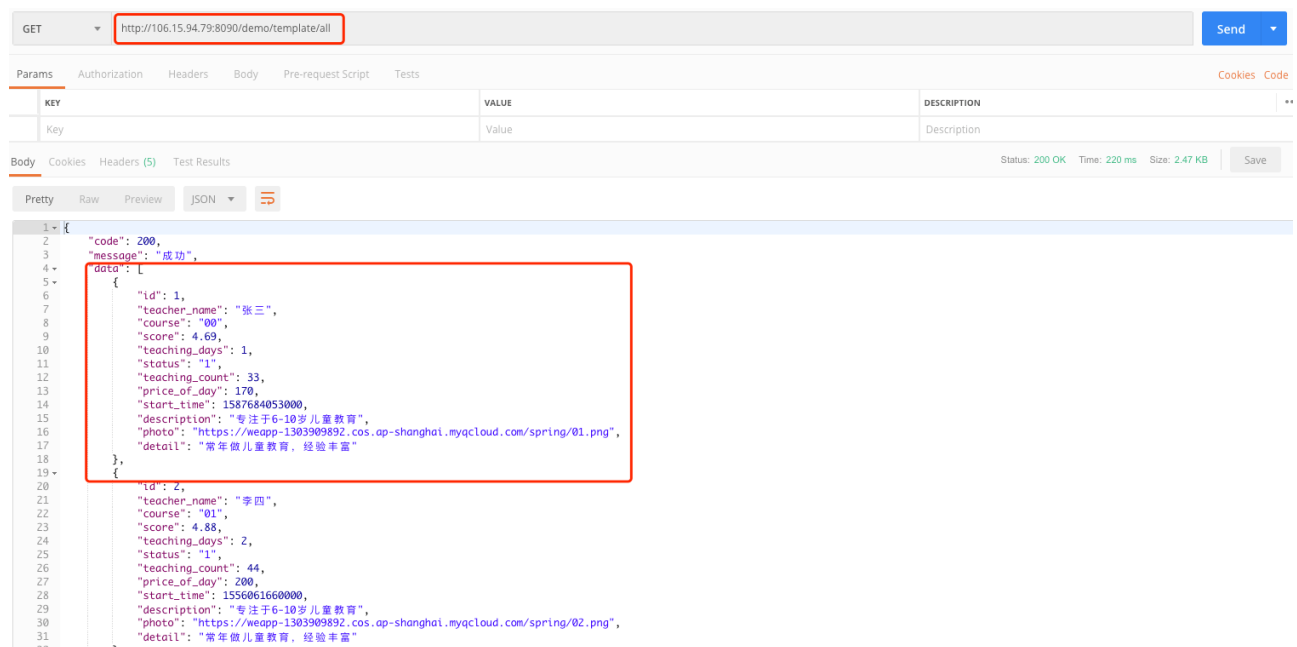


图 1 访问 JDBCTemplate 的 templateGetAll 方法

3、JdbcTemplate 还存在哪些问题

通过 JDBC、连接池、JdbcTemplate 几个阶段的改造以后，我们逐渐解决了数据库连接效率问题（连接池），手动打开关闭数据库连接（JdbcTemplate）。但是作为 JdbcTemplate 还是需要将 SQL 语句硬编码到 java 代码中，如果 SQL 语句修改，需要重新编译 java 代码。而且依旧需要输入 SQL 语句以及对应的 Params 作为参数。

如果能够将数据库对应的实体类与查询语句对应。同时针对使用最频繁的查询语句而言，如果针对不同的条件匹配不同的参数就更好了。因此我们需要一个更加强大的数据库持久层，同时具备更加灵活的 SQL 查询方式，不仅能够根据不同的数据库平台调整 SQL 的语法，也可以根据业务的调整手动调整 SQL 内容，而不用重新编译程序。

为了避免 JdbcTemplate 的这些问题，我们引出 Mybatis 的数据库访问方式。

MyBatis 是支持 SQL 查询，存储过程和高级映射的优秀持久层框架。MyBatis 消除了几乎所有的 JDBC 代码和参数的手工设置以及结果集的检索。MyBatis 使用简单的 XML 或注解用于配置和原始映射，将接口和 Java 的 POJOs（Plain Ordinary Java Objects，普通的 Java 对象）映射成数据库中的记录。

4、总结

本节课介绍了 JdbcTemplate 测试的结果，同时引出 JDBC 存在的问题，需要将 SQL 语句硬编码到 java 代码中，如果 SQL 语句修改，需要重新编译 java 代码等等。为了解决这些问题我们会使用 Mybatis 替代 JdbcTemplate 完成数据库访问的工作。下节课，我们分析 Mybatis 原理，同时对项目进行继续改造，使用 Mybatis 完成数据库的查询工作。我们下期见，拜拜。