



链滴

自定义持久层框架的设计思路

作者: [terwergreen](#)

原文链接: <https://ld246.com/article/1661598310252>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

核心要点

1. 解析配置文件

- 数据库配置信息
- sql 的封装

2. 构建 SqlSessionFactory, 注意这里的 openSession 方法

3. 拿到 SqlSession

- 定义 SqlSession 基本方法
- 封装具体的执行逻辑, Executor

Execute 的 query 方法就对应 jdbc 操作

- 优化, 使用 JDK 动态代理避免 statementId 的硬编码

4. 将 SqlSession 的操作封装到 DAO 层

使用端 (项目)

- 引入自定义持久层框架的 jar 包
- 提供两部分配置信息
 - 数据库配置信息
 - sql 配置信息: sql 语句、参数类型、返回值类型
- 使用配置文件来提供这两部分配置信息:

(1) `sqlMapConfig.xml` : 存放数据库配置信息, 存放 `mapper.xml` 的全路径

(2) `mapper.xml` : 存放 sql 配置信息

自定义持久层框架本身 (工程)

本质是对 JDBC 代码进行封装

- 加载配置文件

根据配置文件的路径记载成字节输入流, 存储到内存中

创建 Resources 类

方法: `InputStream getResourceAsStream(String path)`

- 创建两个 JavaBean (容器对象): 存放的是配置文件解析出来的内容

Configuration: 核心配置类, 存放 `sqlMapConfig.xml` 解析出来的内容

MappedStatement: 映射配置类, 存放 `mapper.xml` 解析出来的内容

- 解析配置文件: dom4j

创建类: `sqlSessionFactoryBuilder` 类, 方法 `build(InputStream in)`

1、使用 dom4j 解析配置文件, 将解析出来的内容封装到容器对象中

2、创建 SqlSessionFactory 对象，生产 SqlSession（会话对象），工厂模式

- 创建 SqlSessionFactory 以及实现类 DefaultSqlSessionFactory

`openSession()`: 生产 SqlSession

- 创建 SqlSession 接口及实现类 DefaultSqlSession

定义对数据库的 CRUD 操作:

```
selectList()
selectOne()
update()
delete()
```

- 创建 Executor 实现类以及实现类 SimpleExecutor

`query(Configuration configuration, MappedStatement mappedStatement, Object... params)`

执行 JDBC 代码

文章更新历史

2022/05/08 feat:新增 Kotlin 版代码实现。

2022/03/14 feat:初稿。