

对象模型与关系数据库的映射框架

目前主要有两种对象模型和关系数据库的映射框架——Hibernate、MyBatis

Hibernate

一种全自动框架对象关系模型持久化框架，对数据库的操作提供完成封装，开发效率较高

我们只需要设计系统的对象模型，Hibernate会根据对象模型去生成对应的表结构、把关于对象的操作转换为对关系数据库的SQL操作

这样做开发效率较高，但其生成的SQL语句并不是最优的，效能较差

MyBatis

一种半自动框架对象关系模型持久化框架

通过采用XML和注解的方式，把对象模型和关系数据库的映射写成配置信息，通过配置信息把对象模型转换为关系数据库的表，把数据库操作映射为接口中的方法，从而省去关于JDBC、参数设置、结果处理等操作，让我们只需要去管理对象模型和关系数据库的映射，以及怎样把对象模型的操作转换为对数据库的操作

因为是半自动框架，所以我们需要手动写SQL语句、手动处理映射，这样做开发效率较低，但性能可调优

对象关系映射

对象模型中的一些概念：类、对象、属性、继承关系、关联关系等

关系数据库中的一些概念：表、记录、字段、外键等

这些概念之间存在一些相似性，我们可以将相似的部分作为映射，如：

类——表

对象——记录

（实例化对象在内存中的id——记录中的主键）

属性——字段

关联关系——外键

MyBatis配置信息

用XML文件写

用注解写

用注解写在映射的接口前

MyBatis标记

MyBatis在xml和注解中都提供了一系列的标签、注解、标记来描述对象模型和关系数据库的映射

一些主要的标记：

- insert：把新增对象的操作映射为数据库中的insert操作
- update：把对对象模型的修改映射为数据库中的update操作
- delete：销毁对象映射为数据库中的delete操作
- select：最常用，会返回很多不同类型的结果集，如果返回结果集可以简单映射为一个类，就可以直接把类作为返回的类型；如果是复杂的结果集，就可以用resultMap标记把结果集映射成为一组有关联的复杂的对象
- 传参：如果是简单的一个参数，可以直接传过去，MyBatis会负责把这个参数映射成为SQL语句中的参数；如果是多个的、复杂的参数，需要定义一个parameterMap，把多个参数放到parameterMap中，传到MyBatis里，MyBatis从中拿到对应参数映射到SQL中的参数上
- sql：这个标注主要用来定义重用的SQL代码的，需要多次用到的SQL代码可以用sql标记，然后在需要的地方用include标签引进（类似于xml文件中的<properties>）