

目录

目录	1
Home	2
本文只是简单介绍了MyFat的配置和简单使用，更多功能，请移步到JSqlBox项目去了解，因为MyFat只是一个粘合剂，它本身不提供DAO功能，而是依赖于JSqlBox。	2
简介 Intro	2
优点 Advantages	2
与其它插件的一些区别	2
配置 Configuration	2
入门 First Example	2
示例 Demo	3
作者其它开源项目 Other Projects	3
期望 Futures	4
版权 License	4
关注我 About Me	4

Home

本文只是简单介绍了MyFat的配置和简单使用，更多功能，请移步到[jSqlBox](#)项目去了解，因为MyFat只是一个粘合剂，它本身不提供DAO功能，而是依赖于jSqlBox。

简介 | Intro

MyFat是MyBatis的DAO功能增强插件，利用全功能持久层工具jSqlBox来补足MyBatis缺少的功能。MyFat需Java8或以上。

优点 | Advantages

1. 无侵入性，透明式增强。对于使用MyBatis的项目只需要添加MyFat依赖即可，不用更改任何其它配置文件和代码，即使使用了其它插件如MyBatis-Plus等，依然可以用MyFat来增强。
2. 架构优秀。MyFat的源码只有7个类，开发时间只有两周，但实现的功能绝不比其它MyBatis插件少。这是因为它采用的是组合模式，即将另一个全功能持久层工具jSqlBox组合到MyBatis中，功能互补，而不是从头开发。内核和插件互不依赖，独立发展，提高了可维护性。
3. 功能全。以下为增强后的功能：
跨数据库分页、DDI生成、实体源码生成、函数变换、主键生成、多种SQL写法、DataMapper、ActiveRecord、Tail、实体越级关联查询、主从、分库分表等。引入了MyFat后，可以说只要你能想到的SQL写法，一定已经包括进来了。(这么多SQL写法，是因为架构导致的结果，而不是为了炫技)

与其它插件的一些区别

1. 目的不一样，其它插件是为MyBatis锦上添花，这个插件目的是让使用者了解jSqlBox的功能，抢一抢MyBatis的市场，希望有朝一日将MyBatis从项目中踢出去，因为MyBatis的主要功能如SqlMapper、用XML管理SQL、条件查询器，在作者看来都是反模式(这也是为什么jSqlBox缺少这些功能)，在jSqlBox中都有其它实现方式。
2. ActiveRecord可以只声明接口，不一定需要继承类，也不需要定义Mapper，更无侵入性
3. 没有专门的分页方法，但是所有SQL查询都可以分页，无侵入性
4. 支持的SQL写法更多，如参数内嵌式写法、实体越级关联查询等
5. 实体注解更符合JPA标准

配置 | Configuration

在pom.xml中加入以下内容即可，注意MyFat必须先于MyBatis加载：

```
<dependency>
  <groupId>com.github.drinkjava2</groupId>
  <artifactId>myfat</artifactId>
  <version>3.0.0</version> <!--或最新版-->
</dependency>

<dependency>
  <groupId>org.mybatis</groupId>
  <artifactId>mybatis</artifactId>
  <version>3.5.2</version> <!--或最新版-->
</dependency>
```

如果在Spring环境下,则上面的第二项MyBatis依赖要改成Spring的：

```
<dependency>
  <groupId>org.mybatis.spring.boot</groupId>
  <artifactId>mybatis-spring-boot-starter</artifactId>
  <version>2.1.0</version> <!--或最新版-->
</dependency>
```

总之MyFat必须先于MyBatis加载。

入门 | First Example

以下示例演示了MyFat的使用，至于SqlSessionFactory的配置，采用标准MyBatis的配置即可，完整的源码可见Demo目录下的myfattest:

```

SqlSession session = sqlSessionFactory.openSession(false);
try {
    // SqlSession有了增强的SQL方法，jSqlBox中的pinte系列方法可用
    Assert.assertEquals(50, session.iQueryForLongValue("select count(*) from user_tb where age>", ques(50),
        " and age <= ", ques(200)));

    Assert.assertEquals(100, session.eCountAll(User.class, " where age>0"));

    UserMapper mapper = session.getMapper(UserMapper.class);

    // 如果mapper继承于RootMapper<User>, 也有增强的pintea系方法了
    Assert.assertEquals(100, mapper.aCountAll(" where age>?", param(0), " and age<?", param(200)));

    Helper.pagin(2, 15); // 无侵入的分页
    try {
        List<User> users = mapper.getUsersOlderThan(50);
        Assert.assertEquals(15, users.size());
    } finally {
        Helper.clear();
    }

    new User().insert(session.ctx()); // ActiveRecord写法
    new Tail().putTail("id", "1").insert(session.ctx(), tail("user_tb")); // Tail相当于JFinal中的Record
    session.ctx().iExecute("insert into user_tb (id) values(?)", param('a'));
    session.iExecute("insert into user_tb (id) values(?)", param('b'));
    mapper.iExecute("insert into user_tb (id) values(?)", param('c'));
    mapper.insertOneUser("d");
} catch (Exception e) {
    System.out.println("Exception:" + e.getMessage());
} finally {
    session.close();
}

```

上例是手工进行SqlSession的获取和关闭，适用于纯MyBatis环境。如果在Spring或SpringBoot环境下，则有以下写法：

```

@Transactional
public void insertWithDiv0() {
    for (int i = 0; i < 100; i++)
        new Customer().putField("name", "Foo" + i).insert();
    Assert.assertEquals(15, new Customer().findAll(" where name<>'", pagin(2, 15)).size());

    customerMapper.insertByIdAndName("2", "Sam");
    customerMapper.iExecute("insert into customer (id,name) values (?,?)", param("3", "Tom"));
    iExecute("insert into customer (id,name) values (?,?)", param("4", "Cat"));
    new Customer().insert();
    Assert.assertEquals(104, new Customer().countAll());
    System.out.println(1 / 0);
}

```

Mapper由Spring注入，而且也不需要手工获取和管理SqlSession了。

示例二中 new Customer().insert(); 这种ActiveRedord写法，不需要传入一个ctx参数，它的前提是设定了一个缺省全局上下文(SqlBoxContext)，详见jSqlBox项目介绍及demo/myfat-springboot示例。

示例 | Demo

- [myfittest](#) 演示纯MyBatis环境下，手工获取SqlSesion进行操作
- [myfat-springboot](#) 演示MyFat在SpringBoot环境下的配置和使用

作者其它开源项目 | Other Projects

- [Java持久层工具 jSqlBox](#)
- [数据库方言工具 jDialects](#)
- [一个独立的声明式事务工具 jTransactions](#)
- [一个微型IOC/AOP工具 jBeanBox](#)
- [一个服务端布局工具 jWebBox](#)

[Home](#)

期望 | Futures

欢迎发issue提出更好的意见或提交PR，帮助完善MyFat

版权 | License

[Apache 2.0](#)

关注我 | About Me

[Github](#)

[码云](#)

