

@Transactional注解

Springboot默认集成事物，只要在方法上加@Transactional即可。

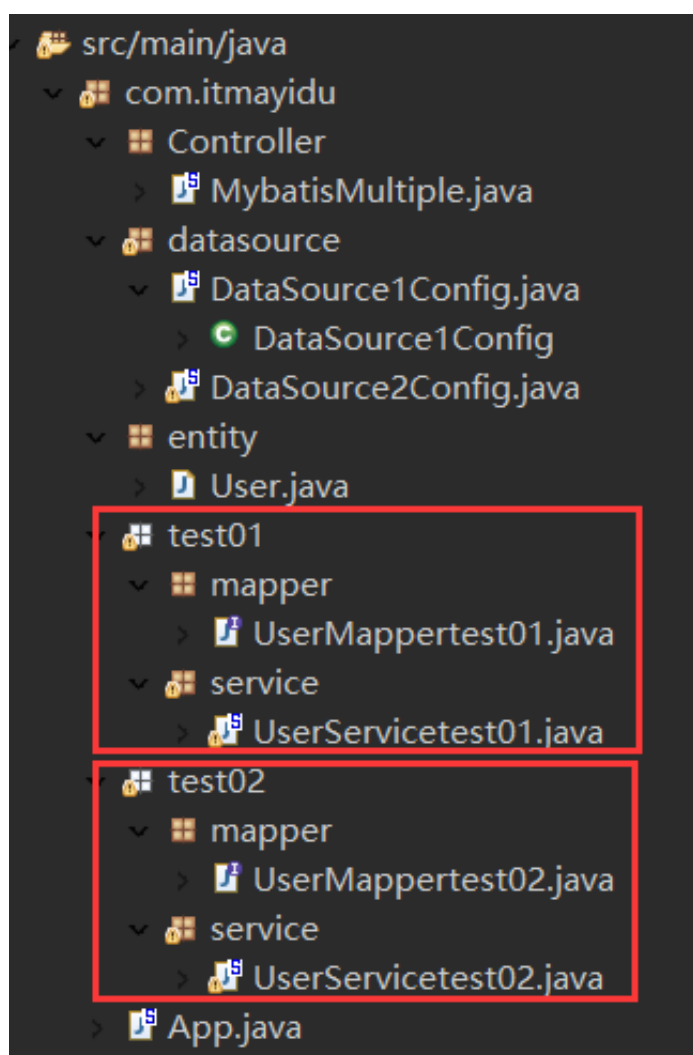
这个注解是事务的注解应用于Springboot的service层的方法上@Transactional

多数据源时要指定事物管理器：@Transactional(transactionManager = "xxxxxxxxx")

SpringBoot-整合多数据源

以包为不同的数据源，使用分包方式拆分数据源

1、分包



2、在application.properties手动指定多个数据源

```
spring.datasource.test1.driver-class-name=com.microsoft.sqlserver.jdbc.SQLServerDriver
spring.datasource.test1.url=jdbc:sqlserver://localhost:1433;DatabaseName=test01
spring.datasource.test1.username=sherry
spring.datasource.test1.password=xiaoai
server.port=2222
spring.datasource.test2.driver-class-name=com.microsoft.sqlserver.jdbc.SQLServerDriver
spring.datasource.test2.url=jdbc:sqlserver://localhost:1433;DatabaseName=test02
spring.datasource.test2.username=sherry
spring.datasource.test2.password=xiaoai
```

3、配置不同的数据源DataSource1Config、DataSource2Config.....（重点）见附件：[DataSource1Config.java](#)

使用jta+atomikos解决分布式事务

即是java中对事务处理的api，为Java平台提供增值服务的并且开源类事务管理器

1、引入依赖

```
<dependency>
```

```
<groupId>org.springframework.boot</groupId>
```

```
<artifactId>spring-boot-starter-jta-atomikos</artifactId>
```

```
</dependency>
```

2、在application.properties手动指定多个数据源，有要求

```
#automatic+jta
mysql.datasource.test1.url=jdbc:mysql://localhost:3306/test1
mysql.datasource.test1.username=root
mysql.datasource.test1.password=.....
mysql.datasource.test1.minPoolSize=3
mysql.datasource.test1.maxPoolSize=25
mysql.datasource.test1.maxLifetime=20000
mysql.datasource.test1.borrowConnectionTimeout=30
mysql.datasource.test1.loginTimeout=30
mysql.datasource.test1.maintenanceInterval=60
mysql.datasource.test1.maxIdleTime=60

mysql.datasource.test2.url=jdbc:mysql://localhost:3306/test2
mysql.datasource.test2.username=root
mysql.datasource.test2.password=.....
mysql.datasource.test2.minPoolSize=3
mysql.datasource.test2.maxPoolSize=25
mysql.datasource.test2.maxLifetime=20000
mysql.datasource.test2.borrowConnectionTimeout=30
mysql.datasource.test2.loginTimeout=30
mysql.datasource.test2.maintenanceInterval=60
mysql.datasource.test2.maxIdleTime=60
```

固定格式

3、配置bean类

```

@ConfigurationProperties(prefix="mysql.datasource.test1")
public class DBConfig1 {
    private String url;
    private String username;
    private String password;
    private int minPoolSize;
    private int maxPoolSize;
    private int maxLifetime;
    private int borrowConnectionTimeout;
    private int loginTimeout;
    private int maintenanceInterval;
    private int maxIdleTime;
    public String getUrl() {
        return url;
    }
    public void setUrl(String url) {
        this.url = url;
    }
    public String getUsername() {
        return username;
    }
    public void setUsername(String username) {
        this.username = username;
    }
    public String getPassword() {
        return password;
    }
}

```

4、配置数据源管理Bean 见附件: [test1MybatisConfig.java](#)

热部署 Devtools

1、引入依赖

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<optional>true</optional>

</dependency>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

<configuration>

<fork>true</fork> <!-- 如果没有该配置，devtools不会生效 -->

</configuration>

</plugin>

2、yml配置

devtools:

livereload:

enabled: true #是否支持livereload

port: 35729

restart:

enabled: true #是否支持热部署