

一、数据源DataSource

1.数据源

数据源是数据库连接的来源，通过DataSource接口获取

2.类型

通用型数据源 (javax.sql.DataSource)

主要使用场景：通用型数据库，本地事务，一般通过Socket方式连接

分布式数据源 (javax.sql.XADataSource)

主要使用场景：通用数据库，分布式事务，一般通过Socket方式连接

嵌入式数据源

(org.springframework.jdbc.datasource.embedded.EmbeddedDatabase) 嵌入式数据库比如浏览器中历史数据

主要使用场景：本地文件系统数据库 如：HSQL、H2、Derby等。

枚举：

org.springframework.jdbc.datasource.embedded.EmbeddedDatabaseType

二、事务Transaction

事务用于提供数据完整性，并在并发访问下确保数据视图的一致性

事务的三个重要概念

- 自动提交模式 (Auto-commit mode)

概念：默认情况下，当独立的sql语句执行后，当前事务将会自动提交

触发时机

- DML执行后
- DDL执行后
- Select查询后结果集关闭后
- 存储过程执行后（如果执行返回结果集的话，待其关闭后）

- 事务隔离级别 (Transaction isolation levels)

概念：事务隔离级别决定事务中执行语句中的数据可见性

事务并发可能的影响

- 脏读
- 不可重复读
- 幻读

- 保护点 (Savepoints)

保护点是在事务中创建，提供细粒度事务控制

使用场景

- 部分事务回滚
- 选择性释放

三、JDBC 4.0(JSR-221)

核心接口

- 驱动接口：java.sql.Driver
- 驱动管理：java.sql.DriverManager
- 数据源：javax.sql.DataSource
- 数据连接：java.sql.Connection
- 执行语句：java.sql.Statement
- 查询结果集：java.sql.ResultSet

元数据接口：

可以使用元数据接口，来看数据源是否支持事务 Metadata。元数据就是描述内容数据的信息。

四、Spring Boot整合

1.spring在properties文件中配置了数据库连接的基本信息之后，可以在自动装配好DataSource进行引入。

2.java1.7之后不用对数据库连接 Connection 进行关闭，因为都实现了 AutoCloseable接口。

3.这里也可直接注入spring框架中的JdbcTemplate进行数据库的操作。

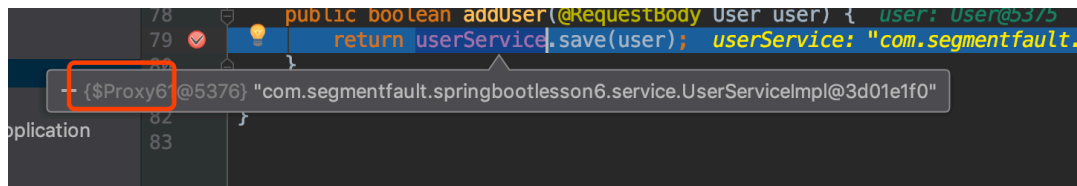
DataSourceConfig中有关于数据库连接池的配置。这里要清楚一个概念：这些class文件下下来代码之后，有些会报错，这个是因为你的项目中可能没有

对应的包的依赖，所以会报错。jar包中的代码不用担心编译错误，不会运行到即可。

使用数据库连接池：dbcp 进行配置。（现在dbcp已经被淘汰）

4. 事务

有事务注解的service类在controller中注入是一个代理对象。有接口的方式



消息中间件

- (1) 一定要保证消息可达。
- (2) 消息的可序性。如果A消息和B消息有依赖关系的时候要保证有序

jdbc中的connection和orm的session其实是一个东西。但是ORM框架中的session有两种：有状态和无状态的。其中无状态的和connection一样，不保持状态。

编程式的事务方式 注入PlatformTransActionManager 进行提交事务、回滚事务等。