



链滴

# Spring 事务

作者: [helly](#)

原文链接: <https://ld246.com/article/1511267049500>

来源网站: [链滴](#)

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

## Spring事务的隔离级别

ISOLATION\_DEFAULT // 默认值, 指底层数据库的默认隔离级别

ISOLATION\_READ\_UNCOMMITTED

ISOLATION\_READ\_COMMITTED

ISOLATION\_REPEATABLE\_READ

ISOLATION\_SERIALIZABLE

## Spring事务的传播行为

PROPAGATION\_REQUIRED: 如果当前存在事务, 则加入该事务; 如果当前没有事务, 则创建一个事务。

PROPAGATION\_REQUIRES\_NEW: 创建一个新的事务, 如果当前存在事务, 则把当前事务挂起。

PROPAGATION\_SUPPORTS: 如果当前存在事务, 则加入该事务; 如果当前没有事务, 则以非事务方式继续运行。

PROPAGATION\_NOT\_SUPPORTED: 以非事务方式运行, 如果当前存在事务, 则把当前事务挂起。

PROPAGATION\_NEVER: 以非事务方式运行, 如果当前存在事务, 则抛出异常。

PROPAGATION\_MANDATORY: 如果当前存在事务, 则加入该事务; 如果当前没有事务, 则抛出异常。

PROPAGATION\_NESTED: 如果当前存在事务, 则创建一个事务作为当前事务的嵌套事务来运行; 果当前没有事务, 则该取值等价于TransactionDefinition.PROPAGATION\_REQUIRED。

## Spring事务超时

@Transactional(timeout=x) // 单位是秒

## Spring只读事务

@Transactional(read-only=true)

在这个事务里只能有数据库读操作, 不能有写操作, 否则会抛出异常, 然后事务回滚。

true readonly只读

false read/write读写

## Spring事务回滚

如果在事务中抛出RuntimeException, 事务会回滚。

## 自动提交

数据库默认操作模式是autocommit自动提交模式。这就表示除非显式地开始一个事务, 否则每个查都被当做一个单独的事务自动执行。

spring会将底层数据库连接的自动提交特性设置为false。

当一个连接关闭时，如果有未提交的事务应该如何处理？JDBC规范没有提及，C3P0默认的策略是回任何未提交的事务。

## Spring编程式事务管理

TransactionTemplate

## Spring声明式事务管理

@Transactional(value="", read-only=true/false, timeout=x, isolation=xx.xx, propagation=xx.x, ...)

@Transactional 可以作用于接口、接口方法、类以及类方法上。但是 Spring 建议不要在接口或者接口方法上使用该注解，因为这只有在使用基于接口的代理时它才会生效。

```
\<tx:annotation-driven transaction-manager="txManager" />
\<bean id="transactionManager" class="org.springframework.jdbc.datasource.DataSourceTr
nsactionManager">
  <!-- 传入dataSource -->
  \<property name="dataSource" ref="dataSource" />
\</bean>
```