1. 事务管理用来确保数据的完整性和一致性。
2. Spring的核心事务管理抽象接口是PlatformTransactionManager。
3. 事务传播行为当事务方法被另一个事务方法调用时，必须指定事务应该如何传播。属性为：propagation=Propagation.REQUIRES(默认值，使用被调用当前的事务方法的事务)，propagation=Propagation.REQUIRES\_NEW(开启单独的事务)。
4. 并发事务所导致的问题有，脏读，不可重复读，幻读。
5. 使用isolation指定事务的隔离级别，常用的却只为READ\_COMMITTED(读已提交)
6. 默认情况下Spring声明式事务对所有的运行时异常进行回滚
7. 使用readOnly指定事务是否为只读，表示这个事务只读取数据但不更新数据。
8. 使用timeout指定强制回滚之前事务可以占用的时间。
9. 事务配置（注解方式）：
   1. 配置事务管理器，配置管理器中的dataSource属性
   2. 启动事务注解<tx:annotation-driven transaction-manager=”事务管理器Id”/>
   3. 在事务方法中加上事务注解@Transactional
10. 事务配置（xml方式）
    1. 配置相应的事务管理器
    2. 配置事务属性<tx:advice transaction-manager=”事务管理器Id”/> <tx:attributes /> <tx:method name=”” 其他属性。。。/>
    3. 配置事务切入点，把事务切入点和事务属性连接起来。<aop:config /> <aop:pointcut /> <aop:advisor />