**数据库事务的特性**

数据库事务的特性包括原子性、一致性、隔离性、持久性。

* 所谓原子性，就是指一个事务中的所有操作要么全成功，要么全失败；
* 所谓一致性就是指事务执行前后都必须处于一致性的状态；
* 所谓隔离性，是指开启的事务不能被其他的事务所影响，多个并发事务彼此隔离；
* 所谓持久性表示事务正确执行完后对于数据的改变是永久的。

其中，关于隔离性，数据库的隔离级别有四种，分别是读未提交，读已提交，可重复读、串行化。  
梳理四种隔离级别前，先来了解下事务并发时有3种常见的问题——脏读、不可重复读、幻读。

**事务并发的常见问题**

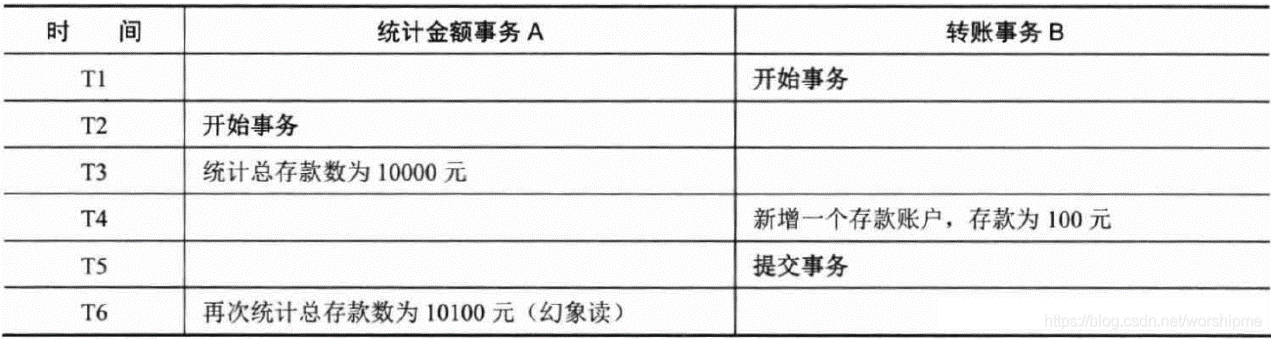
**脏读：**

所谓脏读就是A事务读取到了B事务还未提交的数据。  
图例分析：  


**不可重复读：**

不可重复读指的是A事务读取到了B事务已经提交的更改数据，在同个时间段内，两次查询结果不一致。  
图例分析：  


**幻读：**

幻读一般是在数据统计的事务上，A事务读取到了B事务提交的新增数据，导致统计结果前后不一致。  
图例分析：  


**数据库的隔离级别**

**四种隔离级别简介**

四种隔离级别，从程度从低到高依次是**读未提交，读已提交，可重复读、串行化**。隔离级别越大安全性越高，同时效率也越低，如果设置串行化，那就不存在并发操作了。四种隔离级别对于上述并发下可能出现的三个问题的控制情况可用一张图来描述：  


**常见数据库默认隔离级别**

不同的数据库默人的隔离级别有差别，Oracle、SQL Server 默认隔离级别是读已提交，Mysql的默认隔离级别是可重复读。

数据库事务的隔离级别有4个，由低到高依次为Read uncommitted(未授权读取、读未提交)、Read committed（授权读取、读提交）、Repeatable read（可重复读取）、Serializable（序列化），这四个级别可以逐个解决脏读、不可重复读、幻象读这几类问题。

* Read uncommitted(未授权读取、读未提交)： 写不让写  
  如果一个事务已经开始写数据，则另外一个事务则不允许同时进行写操作，但允许其他事务读此行数据。该隔离级别可以通过“排他写锁”实现。这样就避免了更新丢失，却可能出现脏读。也就是说事务B读取到了事务A未提交的数据。
* Read committed（授权读取、读提交）： 写不让读  
  读取数据的事务允许其他事务继续访问该行数据，但是未提交的写事务将会禁止其他事务访问该行。该隔离级别避免了脏读，但是却可能出现不可重复读。事务A事先读取了数据，事务B紧接了更新了数据，并提交了事务，而事务A再次读取该数据时，数据已经发生了改变。
* Repeatable read（可重复读取）： 读不让写  
  可重复读是指在一个事务内，多次读同一数据。在这个事务还没有结束时，另外一个事务也访问该同一数据。那么，在第一个事务中的两次读数据之间，即使第二个事务对数据进行修改，第一个事务两次读到的的数据是一样的。这样就发生了在一个事务内两次读到的数据是一样的，因此称为是可重复读。读取数据的事务将会禁止写事务（但允许读事务），写事务则禁止任何其他事务。这样避免了不可重复读取和脏读，但是有时可能出现幻象读。（读取数据的事务）这可以通过“共享读锁”和“排他写锁”实现。
* Serializable（序列化）：   
  提供严格的事务隔离。它要求事务序列化执行，事务只能一个接着一个地执行，但不能并发执行。如果仅仅通过“行级锁”是无法实现事务序列化的，必须通过其他机制保证新插入的数据不会被刚执行查询操作的事务访问到。序列化是最高的事务隔离级别，同时代价也花费最高，性能很低，一般很少使用，在该级别下，事务顺序执行，不仅可以避免脏读、不可重复读，还避免了幻像读。   
     
  隔离级别越高，越能保证数据的完整性和一致性，但是对并发性能的影响也越大。对于多数应用程序，可以优先考虑把数据库系统的隔离级别设为Read Committed。它能够避免脏读取，而且具有较好的并发性能。尽管它会导致不可重复读、幻读和第二类丢失更新这些并发问题，在可能出现这类问题的个别场合，可以由应用程序采用悲观锁或乐观锁来控制。大多数数据库的默认级别就是Read committed，比如Sql Server , Oracle。MySQL的默认隔离级别就是Repeatable read。