1. 事务
2. ACID
3. 原子性Atomicity

现实世界中一个不可分割的操作可能对应着数据库中若干条不同的操作，数据库中一条操作也可能被分解为若干个步骤。原子性的意思是某些操作是不可分割的整体，要做全做，要么全不做。

1. 一致性Consistency

如果数据库中的数据符合现实世界中的约束，这些数据就是符合一致性的。数据库本身能解决一部分一致性需求，如主键，唯一索引等等，也可以通过定义触发器来自定义约束条件，保证数据的一致性。更多的需要开发者在业务代码中检查数据的一致性。

1. 隔离性Isolation

保证某次的状态转换不会影响到其他的状态转换，这个规则称为隔离性。

1. 持久性Durability

当一个状态转换完成后，这个转换的结果将会永久保留，这个规则称为持久性。

1. 事务

把需要保证原子性，一致性，隔离性，持久性的一个或多个数据库操作称为事务。根据这些操作所执行的不同阶段事务被分为了几个状态：

1. 活动的

事务对应的数据库操作正在执行过程中。

1. 部分提交的

当事务的最后一个操作执行完成，但是结果还没有刷新到磁盘中。

1. 失败的

当事务处于活动的或者部分提交的状态时，可能遇到了某些错误而无法继续执行。

1. 中止的

当回滚操作执行完毕后，数据库恢复到了执行事务之前的状态。

1. 提交的

当一个处于部分提交状态的事务将修改后的数据都刷新到磁盘中。

1. MySQL中事务的语法
2. 开启事务

两种语法开启事务：

1. BEGIN [WORK];

开启事务后，可以继续写若干条语句。

1. START TRANSACTION;

后面还可以跟随几个修饰符，如：

READ ONLY: 标识当前事务是只读事务，不允许修改普通表数据（临时表可以）。

READ WRITE: 读写事务。

WITH CONSISTENT SNAPSHOT: 启动一致性读。

1. 提交事务

开启事务后就可以编写需要放到该事务中的语句了，当最后一条语句写完后就可以提交该事务了，使用COMMIT语句提交即可。

1. 手动终止事务

ROLLBACK语句代表回滚一个事务。

1. 自动提交

MySQL中有一个系统变量autocommit，用来自动提交事务，默认“on”开启的，如果不显示地开启一个事务，那么每一条语句都算作是一个独立的事务，这种特性称为事务的自动提交。

1. 隐式提交

有些特殊的语句会导致事务自动提交，如下所示：

1. 定义或修改数据库对象的数据定义语言（DDL）

所谓的数据库对象，就是指数据库，表，视图，存储过程等这些东西，当使用CREATE ,ALTER, DROP等语句修改这些数据库对象时，就会隐式地提交前面语句所属的事务。

1. 隐式使用或修改mysql数据库中的表
2. 事务控制或关于锁定的语句

在一个事务没有提交时就重新开启另一个事务，此时会隐式地提交上一个事务。使用LOCK TABLES, UNLOCK TABLES 等锁定的语句也会隐式地提交前面语句所属的事务。

1. 保存点

在事务对应的数据库语句中打几个保存点，在调用ROLLBACK语句时可以指定回滚到哪个点，免得直接回滚一整个事务。定义保存点的语法如下：

SAVEPOINT 保存点名称;

回滚到某个保存点：

ROLLBACK TO 保存点名称;

如果不加保存点名称，直接回滚到事务执行之前的状态。

删除某个保存点：

RELEASE SAVEPOINT 保存点名称;

1. 支持事务的存储引擎

只有InnoDB和NDB支持事务，如果某个事务包含的语句要修改某个表中的数据，但是该表使用的存储引擎不支持事务，那么对该表所做的修改将无法进行回滚。