**题目 01：**

**一条 SQL 语句在 MySQL 中是如何执行的？**

MySQL 可以分为 Server 层和存储引擎层两部分。Server 层包括连接器、缓存器、分析器、优化器、执行器。存储引擎层包括各种类型的存储引擎，例如Innodb、Myism、memory等。

执行过程：

1、客户端先通过连接器将sql传入。

2、到缓存器中找是否已有缓存此sql的查询结果

3、如果没有找到缓存，分析器对sql语句进行词法分析、语法分析等。

4、则会使用优化器对sql进行执行计划生成、索引的选择。

5、接着会使用执行引擎。开始执行的时候，要先判断一下你对这个表 T有没有执行查询的权限，如果没有，就会返回没有权限的错误，如果命中查询缓存，会在查询缓存返回结果的时候，做权限验证。查询也会在优化器之前调用 precheck 验证权限。如果有权限，就打开表继续执行。打开表的时候，执行器就会根据表的引擎定义，去使用这个引擎提供的接口。

**题目 02**

**请解释一下你理解的事务是什么？**

**要点：**

**事务四大特性 ACID**

原子性：事务要么全部被执行，要么就全部不被执行。

一致性：事务的执行使得数据库从一种正确状态转换成另一种正确状态。

隔离性：在事务正确提交之前，不允许把该事务对数据的任何改变提供给任何其他事务。

持久性：事务正确提交后，其结果将永久保存在数据库中，即使在事务提交后。

**事务隔离级别**

隔离级别从低到高有读未提交、读已提交、可重复读和串行化。

读未提交：

假设事务1先于事务2，当事务1更新数据的时候，事务2可以读取事务1未提交的数据，但是不能更新事务1正在更新的数据。而如果事务1只是读数据，那么事务2既可以读数据，也可以更新数据。这种情况下无法规避脏读，不可重复读的问题。

读已提交：

在一个事务修改数据过程中，如果事务还没提交，其他事务不能读该数据，或者说只能读取committed的数据。这种情况下就避免了脏读，但是却不能避免不可重复读的问题。

可重复读：

确保同一事务的多个实例在并发读取数据时，看到同样的数据行。可能出现的问题-幻读。

串行化：

最高的隔离级别。强制事务排序，使之不可能相互冲突，解决幻读问题。

**事务会产生的并发问题**

如果以不受控制的方式允许并发事务交错执行，就有可能发生不期望的结果。当并发地写入和并发地读取就会得到非预期的数据。现象分别为：脏读、不可重复读、幻读和序列化异常。

脏读：当第一个事务读取了第二个事务中已经修改但还未提交的数据，包括INSERT、UPDATE、DELETE，当第二个事务不提交并执行ROLLBACK后，第一个事务所读取到的数据是不正确的，这种读现象称作脏读。

不可重复读：当一个事务第一次读取数据之后，被读取的数据被另一个已提交的事务进行了修改，事务再次读取这些数据时发现数据已经被另一个事务修改，两次查询的结果不一致，这种读现象称为不可重复读。

幻读：指一个事务的两次查询的结果集记录数不一致。例如一个事务第一次根据范围条件查询了一些数据，而另一个事务却在此时插入或删除了这个事务的查询结果集中的部分数据，这个事务在接下来的查询中，会发现有一些数据在它先前的查询结果中不存在，或者第一次查询结果中的一些数据不存在了，两次查询结果不相同，这种读现象称为幻读。幻读是受INSERT和DELETE影响的。

序列化异常：序列化异常是指成功提交的一组事务的执行结果与这些事务按照串行执行方式的执行结果不一致。

**事务的安全性、性能与隔离级别的关系**

安全性：串行化>可重复读>读已提交>读未提交

性能：串行化<可重复读<读已提交<读未提交