数据库使用最多的是 mysql 数据库，在实际项目中都有使用。

（DDL）：

掌握数据库基本的表创建，主外键，索引创建，值约束等。能根据业务需求设计合理的数据表。比如创建一张用户表，能根据字段选择合适的数据类型，user_id 类型使用 int，设置为主键，并设置自增。用户名使用 varchar，限制长度为 10。并设置为 not null。爱好之类的集合可以用 set，性别可以用 emun，每日签到使用布尔(tinyint)。

如果对用户表中，用户名字段会频繁的查询，表中的数据也非常的庞大，就可以为用户名字段创建一个普通的索引（单列索引），提高查询效率。

在实际项目中，表与表之间是有一定关联的，例如订单表和用户表，订单表中的用户 id 是依赖于用户表中的用户 id。可以将订单表中的用户 id 设置为外键，对该字段进行约束。

（DML）：

掌握基本的 sql 语句的增删改，使用 insert into 表名 values 在指定表中插多条数据。Values 中的值与字段顺序一一对应。delete from 表名 where 从表中删除符合条件的语句。update 表名 set where 语句用于更新某表指定字段的值。delete 和 updata 都必须写合理的 where 条件，否则将删除或更新整张表。

（DQL）：

简单的查询语句，比如查询年龄大于 15 并且姓李的用户，可以在 where 后面加上条件 age>15 and name LIKE “李%”;

稍复杂的查询，例如嵌套查询，要查询年龄大于平均年龄的用户有哪些，可以在 where 中嵌套一个子查询，age > select AVG(age) from user。在 select，from 中也都能嵌套子查询，再配合操作符>，<，=，in，not in，exists 等，聚合函数：count()，sum()，avg()，max()，min()等能解决很多查询问题。

还有一些查询，统计男女数量或者对应城市有多少人，可以使用 group by 根据字段分组，查询 city,使用 count(*)计算结果行数加上 group by city 对城市分组。

如果要筛选对应城市人数大于 10 的城市，对分出来的组进行筛选。只需要在 group by 后面加上 having count(*)>10 即可。

对筛选出来的城市，我们可以进行排序，有升序，降序排序，可以利用 order by 加上字段名和 DESC 或 ASC。如果要获得人数最多的前三个城市，那就用 DESC 降序，再使用 limit 3，取前三条数据就完成了。还有一些查询操作比如分页查询，也可以用 limit 实现。

还有表连接查询:有内连接，左连接，右连接。例如查询用户信息对应的订单信息可以使用内连接，inner join 连接这两个表，用这两表共有的字段建立连接关系，on user.user_id = order.user_id。如果要查询哪些用户没有购买过商品，可以使用左连接，用 left join 连接用户表和订单表。使用左连接的话，左边的表数据会保持完整，右表不存在的都是数据都是空值，再加上 where 语句筛选一下就能满足查询要求。右连接也一样，使用 right join 连接两表，右连接很少使用，一般可以用左连接替代。

当有多表的复杂查询时，可以使用视图来简化操作，使用 create view 视图名 as 加上复杂或嵌套的查询操作，使用时将视图名当做一个表来使用，大大简化了操作。还可以限制用户只能访问表的特定列。比如创建一个 userinfo 的视图，这个视图只查询用户的用户名，性别，爱好等，过滤掉密码等敏感信息。

存储过程

掌握存储过程的使用，在一些场景，比如对商品表中的数据进行分页查询，这个过程使用频繁使用而且多变的，我们可以创建一个存储过程来完成。使用 create procedure 加名字，使用 IN 定义传入参数： 第几页，类型为 int，每页数量，类型为 int，在这个存储过程中还用不上传出参数，就不用定义了。让后写上查询商品信息的 sql 语句，加上 limit，然后使用传入的参数即可。创建存储过程后，使用 call 加存储过程的名字，传入需要的参数就能使用了。还可以配合 declare，while，if 等流程控制语句实现更复杂的存储过程。

触发器

在一些时候，比如更改用户表信息时，可以使用触发器进行日志记录，使用 create trigger 加名字，加上 after，表示在触发后记录日志，然后 update on 用户表，在更新用户表后触发，再加上 for each row：对每一行修改都触发，然后在 Begin 和 end 之间写入插入日志表的 sql 语句，可以使用 new 和 old 关键字来获取新的数据和更新之前的数据进行详细的日志记录。

事务

还有对事务的使用，在对订单处理的时候，创建订单时需要同时减少库存，创建订单记录和支付记录，全部完成，这时候就可以使用事务，使用 START TRANSACTION 或 begin 语句开启事务，然后执行一系列的 sql 语句。其中一条执行失败，都会回滚到开启事务后的状态，没有问题使用 commit 结束事务，保存数据。或者使用 rollback 来结束，取消事务中的一系列操作。