



## 第二版：MySQL 50 道

### 目录

第二版：MySQL 50 道	1
1、MySQL 中有哪几种锁？	3
2、MySQL 中有哪些不同的表格？	3
1、MyISAM	4
2、Heap	4
3、Merge	4
4、INNODB	4
5、ISAM	4
3、简述在 MySQL 数据库中 MyISAM 和 InnoDB 的区别	4
4、MySQL 中 InnoDB 支持的四种事务隔离级别名称，以及逐级之间的区别？	5
1、read uncommitted：读到未提交数据	5
2、read committed：脏读，不可重复读	5
3、repeatable read：可重读	5
4、serializable：串行事物	5
5、CHAR 和 VARCHAR 的区别？	5
6、主键和候选键有什么区别？	5
7、MyISAM 表格将在哪里存储，并且还提供其存储格式？	6
8、MySQL 如何优化 DISTINCT？	6
9、如何显示前 50 行？	6
10、可以使用多少列创建索引？	6
11、NOW () 和 CURRENT_DATE () 有什么区别？	6
12、什么是非标准字符串类型？	7
1、TINYTEXT	7
2、TEXT	7
3、MEDIUMTEXT	7
4、LONGTEXT	7
13、myisamchk 是用来做什么的？	7
14、如果一个表有一列定义为 TIMESTAMP，将发生什么？	7



15、你怎么看到为表格定义的所有索引？	8
16、LIKE 声明中的%和_是什么意思？	8
17、列对比运算符是什么？	8
18、BLOB 和 TEXT 有什么区别？	8
19、MySQL_fetch_array 和 MySQL_fetch_object 的区别是什么？	9
20、什么是通用 SQL 函数？	9
21、MySQL 支持事务吗？	9
22、MySQL 里记录货币用什么字段类型好	10
23、MySQL 有关权限的表都有哪几个？	10
24、列的字符串类型可以是什么？	10
1、SET	11
2、BLOB	11
3、ENUM	11
4、CHAR	11
5、TEXT	11
25、MySQL 数据库作发布系统的存储，一天五万条以上的增量，预计运维三年,怎么优化？	11
26、存储时期	11
27、对于关系型数据库而言，索引是相当重要的概念，请回答有关索引的几个问题：	12
1、索引的目的是什么？	12
2、索引对数据库系统的负面影响是什么？	12
3、为数据表建立索引的原则有哪些？	12
4、什么情况下不宜建立索引？	13
28、解释 MySQL 外连接、内连接与自连接的区别	13
29、Myql 中的事务回滚机制概述	13
30、SQL 语言包括哪几部分？每部分都有哪些操作关键字？	14
31、完整性约束包括哪些？	14
32、什么是锁？	15
33、什么叫视图？游标是什么？	15
34、什么是存储过程？用什么来调用？	15
35、如何通俗地理解三个范式？	16
36、什么是基本表？什么是视图？	16



37、试述视图的优点？	17
38、 NULL 是什么意思	17
39、主键、外键和索引的区别？	17
40、你可以用什么来确保表格里的字段只接受特定范围里的值？	18
41、说说对 SQL 语句优化有哪些方法？（选择几条）	18
42、锁的优化策略	19
1、读写分离	19
2、分段加锁	19
3、减少锁持有的时间	19
4.多个线程尽量以相同的顺序去获取资源	19
43、索引的底层实现原理和优化	19
44、什么情况下设置了索引但无法使用	19
45、实践中如何优化 MySQL	19
1、SQL 语句及索引的优化	20
2、数据库表结构的优化	20
3、系统配置的优化	20
4、硬件的优化	20
46、优化数据库的方法	20
47、简单描述 MySQL 中，索引，主键，唯一索引，联合索引的区别，对数据库的性能有什么影响（从读写两方面）	20
48、数据库中的事务是什么？	21
49、SQL 注入漏洞产生的原因？如何防止？	22
50、为表中得字段选择合适得数据类型	22

我们的网站：<https://tech.souyunku.com>

**关注我们的公众号：搜云库技术团队，回复以下关键字**

回复：**进群** 邀请您进「技术架构分享群」

回复：**内推** 即可进：北京，上海，广州，深圳，杭州，成都，武汉，南京，

✧ 微信搜一搜

🔍 搜云库技术团队



郑州, 西安, 长沙「程序员工作内推群」

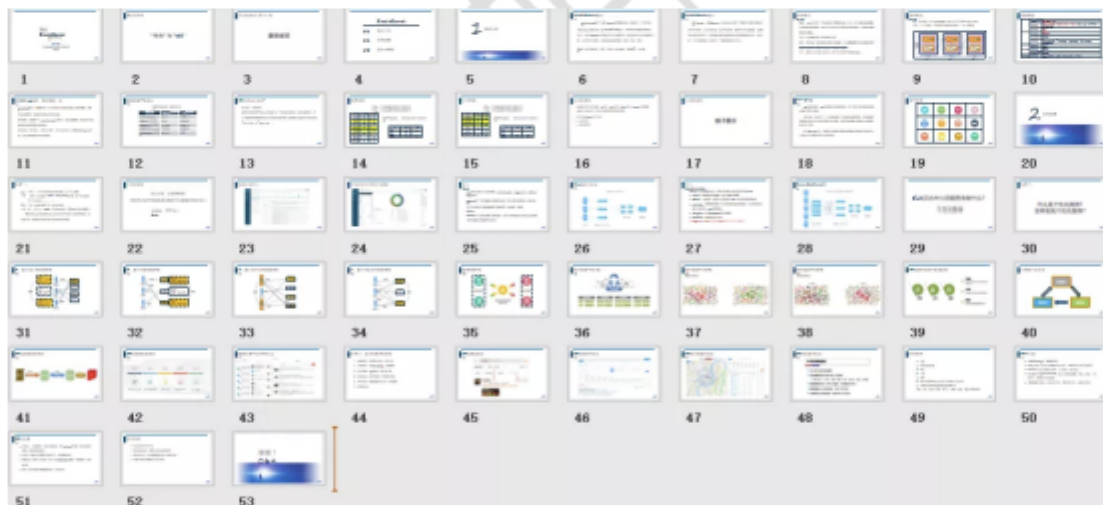
回复 **【1024】** 送 4000G 最新架构师视频

回复 **【PPT】** 即可无套路获取, 以下最新整理调优 PPT!

## 46 页《JVM 深度调优, 演讲 PPT》



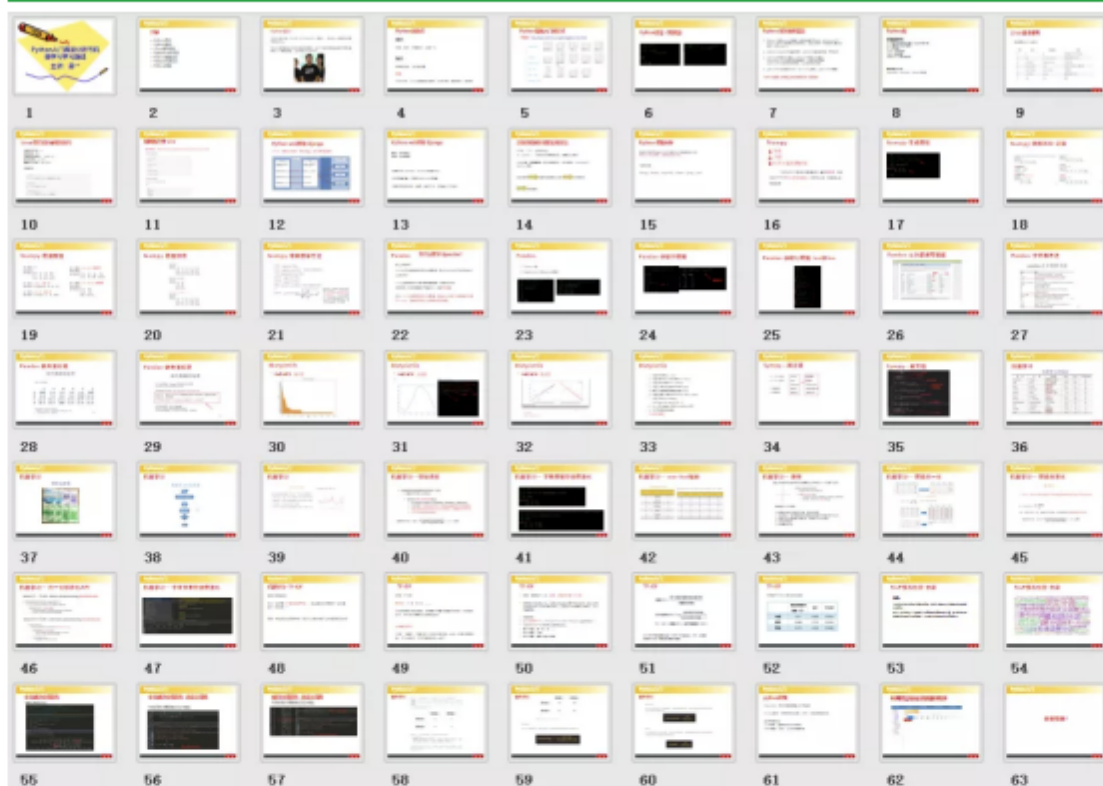
## 53 页《Elasticsearch 调优演讲 PPT》



## 63 页《Python 数据分析入门 PPT》

微信搜一搜

搜云库技术团队



微信扫一扫

<https://tech.souyunku.com>

技术、架构、资料、工作、内推  
专注于分享最有价值的互联网技术干货文章

## 1、MySQL 中有哪几种锁？



1、表级锁：开销小，加锁快；不会出现死锁；锁定粒度大，发生锁冲突的概率最高，并发度最低。

2、行级锁：开销大，加锁慢；会出现死锁；锁定粒度最小，发生锁冲突的概率最低，并发度也最高。

3、页面锁：开销和加锁时间界于表锁和行锁之间；会出现死锁；锁定粒度界于表锁和行锁之间，并发度一般。

## 2、MySQL 中有哪些不同的表格？

共有 5 种类型的表格：

1、MyISAM

2、Heap

3、Merge

4、INNODB

5、ISAM

## 3、简述在 MySQL 数据库中 MyISAM 和 InnoDB 的区别

**MyISAM:**

不支持事务，但是每次查询都是原子的；





支持表级锁，即每次操作是对整个表加锁；

存储表的总行数；

一个 MYISAM 表有三个文件：索引文件、表结构文件、数据文件；

采用非聚集索引，索引文件的数据域存储指向数据文件的指针。辅索引与主索引基本一致，但是辅索引不用保证唯一性。

### InnoDB:

支持 ACID 的事务，支持事务的四种隔离级别；

支持行级锁及外键约束：因此可以支持写并发；

不存储总行数：

一个 InnoDB 引擎存储在一个文件空间（共享表空间，表大小不受操作系统控制，一个表可能分布在多个文件里），也有可能为多个（设置为独立表空间，表大小受操作系统文件大小限制，一般为 2G），受操作系统文件大小的限制；

主键索引采用聚集索引（索引的数据域存储数据文件本身），辅索引的数据域存储主键的值；因此从辅索引查找数据，需要先通过辅索引找到主键值，再访问辅索引；最好使用自增主键，防止插入数据时，为维持 B+树结构，文件的大调整。

## 4、MySQL 中 InnoDB 支持的四种事务隔离级别名称，以及逐级之间的区别？

SQL 标准定义四个隔离级别为：



1、read uncommitted : 读到未提交数据

2、read committed: 脏读, 不可重复读

3、repeatable read: 可重读

4、serializable : 串行事物

## 5、CHAR 和 VARCHAR 的区别?

1、CHAR 和 VARCHAR 类型在存储和检索方面有所不同

2、CHAR 列长度固定为创建表时声明的长度, 长度值范围是 1 到 255 当 CHAR 值被存储时, 它们被用空格填充到特定长度, 检索 CHAR 值时需删除尾随空格。

## 6、主键和候选键有什么区别?

表格的每一行都由主键唯一标识, 一个表只有一个主键。

主键也是候选键。按照惯例, 候选键可以被指定为主键, 并且可以用于任何外键引用。

## 7、MyISAM 表格将在哪里存储, 并且还提供其存储格式?

每个 MyISAM 表格以三种格式存储在磁盘上:

- “.frm” 文件存储表定义





·数据文件具有 “.MYD” （MYData）扩展名

索引文件具有 “.MYI” （MYIndex）扩展名

## 8、MySQL 如何优化 DISTINCT?

DISTINCT 在所有列上转换为 GROUP BY，并与 ORDER BY 子句结合使用。

```
SELECT DISTINCT t1.a FROM t1,t2 where t1.a=t2.a;
```

## 9、如何显示前 50 行?

在 MySQL 中，使用以下代码查询显示前 50 行：

```
SELECT*FROM
```

```
LIMIT 0,50;
```

## 10、可以使用多少列创建索引?

任何标准表最多可以创建 16 个索引列。

## 11、NOW () 和 CURRENT\_DATE () 有什么区别?

NOW () 命令用于显示当前年份，月份，日期，小时，分钟和秒。



CURRENT\_DATE () 仅显示当前年份，月份和日期。

## 12、什么是非标准字符串类型？

1、TINYTEXT

2、TEXT

3、MEDIUMTEXT

4、LONGTEXT

## 13、myisamchk 是用来做什么的？

它用来压缩 MyISAM 表，这减少了磁盘或内存使用。

### MyISAM Static 和 MyISAM Dynamic 有什么区别？

在 MyISAM Static 上的所有字段有固定宽度。动态 MyISAM 表将具有像 TEXT，BLOB 等字段，以适应不同长度的数据类型。

MyISAM Static 在受损情况下更容易恢复。

## 14、如果一个表有一列定义为 TIMESTAMP，将发生什么？

每当行被更改时，时间戳字段将获取当前时间戳。

列设置为 AUTO INCREMENT 时，如果在表中达到最大值，会发生什么情况？



它会停止递增，任何进一步的插入都将产生错误，因为密钥已被使用。

怎样才能找出最后一次插入时分配了哪个自动增量？

LAST\_INSERT\_ID 将返回由 Auto\_increment 分配的最后一个值，并且不需要指定表名称。

## 15、你怎么看到为表格定义的所有索引？

索引是通过以下方式为表格定义的：

SHOW INDEX FROM <tablename>;

## 16、LIKE 声明中的%和\_是什么意思？

%对应于 0 个或更多字符，\_只是 LIKE 语句中的一个字符。

如何在 Unix 和 MySQL 时间戳之间进行转换？

UNIX\_TIMESTAMP 是从 MySQL 时间戳转换为 Unix 时间戳的命令

FROM\_UNIXTIME 是从 Unix 时间戳转换为 MySQL 时间戳的命令

## 17、列对比运算符是什么？

在 SELECT 语句的列比较中使用 =, <>, <=, <, >=, >, <<, >>, <=>, AND, OR 或 LIKE 运算符。



## 18、BLOB 和 TEXT 有什么区别？

BLOB 是一个二进制对象，可以容纳可变数量的数据。TEXT 是一个不区分大小写的 BLOB。

BLOB 和 TEXT 类型之间的唯一区别在于对 BLOB 值进行排序和比较时区分大小写，对 TEXT 值不区分大小写。

## 19、MySQL\_fetch\_array 和 MySQL\_fetch\_object 的区别是什么？

以下是 MySQL\_fetch\_array 和 MySQL\_fetch\_object 的区别：

MySQL\_fetch\_array() - 将结果行作为关联数组或来自数据库的常规数组返回。

MySQL\_fetch\_object - 从数据库返回结果行作为对象。

## 20、什么是通用 SQL 函数？

1、CONCAT(A, B) - 连接两个字符串值以创建单个字符串输出。通常用于将两个或多个字段合并为一个字段。

2、FORMAT(X, D)- 格式化数字 X 到 D 有效数字。

3、CURRDATE(), CURRTIME()- 返回当前日期或时间。

4、NOW () - 将当前日期和时间作为一个值返回。



5、MONTH ( ) , DAY ( ) , YEAR ( ) , WEEK ( ) , WEEKDAY ( ) – 从日期值中提取给定数据。

6、HOUR ( ) , MINUTE ( ) , SECOND ( ) – 从时间值中提取给定数据。

7、DATEDIFF (A, B) – 确定两个日期之间的差异, 通常用于计算年龄

8、SUBTIMES (A, B) – 确定两次之间的差异。

9、FROMDAYS (INT) – 将整数天数转换为日期值。

## 21、MySQL 支持事务吗?

在缺省模式下, MySQL 是 autocommit 模式的, 所有的数据库更新操作都会即时提交, 所以在缺省情况下, MySQL 是不支持事务的。

但是如果你的 MySQL 表类型是使用 InnoDB Tables 或 BDB tables 的话, 你的 MySQL 就可以使用事务处理, 使用 SET

AUTOCOMMIT=0 就可以使 MySQL 允许在非 autocommit 模式, 在非 autocommit 模式下, 你必须使用 COMMIT 来提交你的更改, 或者用 ROLLBACK 来回滚你的更改。

## 22、MySQL 里记录货币用什么字段类型好

NUMERIC 和 DECIMAL 类型被 MySQL 实现为同样的类型, 这在 SQL92 标准允许。他们被用于保存值, 该值的准确精度是极其重要的值, 例如与金钱有关的数据。当声明一个类是这些类型之一时, 精度和规模的能被(并且通常是)指定。



例如：

`salary DECIMAL(9,2)`

在这个例子中，9(precision)代表将被用于存储值的总的小数位，而 2(scale)代表将被用于存储小数点后的位数。

因此，在这种情况下，能被存储在 salary 列中的值的范围是从-9999999.99 到 9999999.99。

## 23、MySQL 有关权限的表都有哪几个？

MySQL 服务器通过权限表来控制用户对数据库的访问，权限表存放在 MySQL 数据库里，由 MySQL\_install\_db 脚本初始化。这些权限表分别 user, db, table\_priv, columns\_priv 和 host。

## 24、列的字符串类型可以是什么？

字符串类型是：

1、SET

2、BLOB

3、ENUM

4、CHAR

5、TEXT





## 25、MySQL 数据库作发布系统的存储,一天五万条以上的增量,预计运维三年,怎么优化?

- 1、设计良好的数据库结构,允许部分数据冗余,尽量避免 join 查询,提高效率。
- 2、选择合适的表字段数据类型和存储引擎,适当的添加索引。
- 3、MySQL 库主从读写分离。
- 4、找规律分表,减少单表中的数据量提高查询速度。
- 5、添加缓存机制,比如 memcached, apc 等。
- 6、不经常改动的页面,生成静态页面。
- 7、书写高效率的 SQL。比如 `SELECT * FROM TABEL` 改为 `SELECT field_1, field_2, field_3 FROM TABLE`。

## 26、存储时期

**Datetime:**以 YYYY-MM-DD HH:MM:SS 格式存储时期时间,精确到秒,占用 8 个字节得存储空间,datetime 类型与时区无关

**Timestamp:**以时间戳格式存储,占用 4 个字节,范围小 1970-1-1 到 2038-1-19,显示依赖于所指定得时区,默认在第一个列行的数据修改时可以自动得修改 timestamp 列得值

**Date:**(生日) 占用得字节数比使用字符串.datetime.int 储存要少,使用 date 只需要 3 个字节,存储日期月份,还可以利用日期时间函数进行日期间得计算



Time:存储时间部分得数据

注意:不要使用字符串类型来存储日期时间数据(通常比字符串占用得储存空间小,在进行查找过滤可以利用日期得函数)

使用 int 存储日期时间不如使用 timestamp 类型

## 27、对于关系型数据库而言，索引是相当重要的概念，请回答有关索引的几个问题：

### 1、索引的目的是什么？

快速访问数据表中的特定信息，提高检索速度

创建唯一性索引，保证数据库表中每一行数据的唯一性。

加速表和表之间的连接

使用分组和排序子句进行数据检索时，可以显著减少查询中分组和排序的时间

### 2、索引对数据库系统的负面影响是什么？

负面影响：

创建索引和维护索引需要耗费时间，这个时间随着数据量的增加而增加；索引需要占用物理空间，不光是表需要占用数据空间，每个索引也需要占用物理空间；当对表进行增、删、改、的时候索引也要动态维护，这样就降低了数据的维护速度。

### 3、为数据表建立索引的原则有哪些？

在最频繁使用的、用以缩小查询范围的字段上建立索引。



在频繁使用的、需要排序的字段上建立索引

#### 4、什么情况下不宜建立索引？

对于查询中很少涉及的列或者重复值比较多的列，不宜建立索引。

对于一些特殊的数据类型，不宜建立索引，比如文本字段（text）等

## 28、解释 MySQL 外连接、内连接与自连接的区别

先说什么是交叉连接：交叉连接又叫笛卡尔积，它是指不使用任何条件，直接将一个表的所有记录和另一个表中的所有记录一一匹配。

内连接 则是只有条件的交叉连接，根据某个条件筛选出符合条件的记录，不符合条件的记录不会出现在结果集中，即内连接只连接匹配的行。

外连接 其结果集中不仅包含符合连接条件的行，而且还会包括左表、右表或两个表中

的所有数据行，这三种情况依次称之为左外连接，右外连接，和全外连接。

左外连接，也称左连接，左表为主表，左表中的所有记录都会出现在结果集中，对于那些在右表中并没有匹配的记录，仍然要显示，右边对应的那些字段值以 NULL 来填充。右外连接，也称右连接，右表为主表，右表中的所有记录都会出现在结果集中。左连接和右连接可以互换，MySQL 目前还不支持全外连接。

## 29、Myql 中的事务回滚机制概述

事务是用户定义的一个数据库操作序列，这些操作要么全做要么全不做，是一个不可分割的工作单位，事务回滚是指将该事务已经完成的对数据库的更新操作撤销。



要同时修改数据库中两个不同表时，如果它们不是一个事务的话，当第一个表修改完，可能第二个表修改过程中出现了异常而没能修改，此时就只有第二个表依旧是未修改之前的状态，而第一个表已经被修改完毕。而当你把它们设定为一个事务的时候，当第一个表修改完，第二表修改出现异常而没能修改，第一个表和第二个表都要回到未修改的状态，这就是所谓的事务回滚

### 30、SQL 语言包括哪几部分？每部分都有哪些操作关键字？

SQL 语言包括数据定义(DDL)、数据操纵(DML)、数据控制(DCL)和数据查询 (DQL) 四个部分。

数据定义：Create Table,Alter Table,Drop Table, Craete/Drop Index 等

数据操纵：Select ,insert,update,delete,

数据控制：grant, revoke

数据查询：select

### 31、完整性约束包括哪些？

数据完整性(Data Integrity)是指数据的精确(Accuracy)和可靠性(Reliability)。

分为以下四类：

1、实体完整性：规定表的每一行在表中是惟一的实体。



2、域完整性：是指表中的列必须满足某种特定的数据类型约束，其中约束又包括取值范围、精度等规定。

3、参照完整性：是指两个表的主关键字和外关键字的数据应一致，保证了表之间的数据的一致性，防止了数据丢失或无意义的的数据在数据库中扩散。

4、用户定义的完整性：不同的关系数据库系统根据其应用环境的不同，往往还需要一些特殊的约束条件。用户定义的完整性即是针对某个特定关系数据库的约束条件，它反映某一具体应用必须满足的语义要求。

与表有关的约束：包括列约束(NOT NULL (非空约束))和表约束(PRIMARY KEY、foreign key、check、UNIQUE)。

## 32、什么是锁？

答：数据库是一个多用户使用的共享资源。当多个用户并发地存取数据时，在数据库中就会产生多个事务同时存取同一数据的情况。若对并发操作不加控制就可能读取和存储不正确的数据，破坏数据库的一致性。

加锁是实现数据库并发控制的一个非常重要的技术。当事务在对某个数据对象进行操作前，先向系统发出请求，对其加锁。加锁后事务就对该数据对象有了一定的控制，在该事务释放锁之前，其他的事务不能对此数据对象进行更新操作。

基本锁类型：锁包括行级锁和表级锁

## 33、什么叫视图？游标是什么？



答：视图是一种虚拟的表，具有和物理表相同的功能。可以对视图进行增，改，查，操作，视图通常是有一个表或者多个表的行或列的子集。对视图的修改不影响基本表。它使得我们获取数据更容易，相比多表查询。

游标：是对查询出来的结果集作为一个单元来有效的处理。游标可以定在该单元中的特定行，从结果集的当前行检索一行或多行。可以对结果集当前行做修改。一般不使用游标，但是需要逐条处理数据的时候，游标显得十分重要。

### 34、什么是存储过程？用什么来调用？

答：存储过程是一个预编译的 SQL 语句，优点是允许模块化的设计，就是说只需创建一次，以后在该程序中就可以调用多次。如果某次操作需要执行多次 SQL，使用存储过程比单纯 SQL 语句执行要快。可以用一个命令对象来调用存储过程。

### 35、如何通俗地理解三个范式？

答：第一范式：1NF 是对属性的原子性约束，要求属性具有原子性，不可再分解；

第二范式：2NF 是对记录的惟一性约束，要求记录有惟一标识，即实体的惟一性；

第三范式：3NF 是对字段冗余性的约束，即任何字段不能由其他字段派生出来，它要求字段没有冗余。。

范式化设计优缺点：

优点：

可以尽量得减少数据冗余，使得更新快，体积小





缺点:对于查询需要多个表进行关联,减少写得效率增加读得效率,更难进行索引优化

**反范式化:**

优点:可以减少表得关联,可以更好得进行索引优化

缺点:数据冗余以及数据异常,数据得修改需要更多的成本

### 36、什么是基本表? 什么是视图?

答:基本表是本身独立存在的表,在 SQL 中一个关系就对应一个表。视图是从一个或几个基本表导出的表。视图本身不独立存储在数据库中,是一个虚表

### 37、试述视图的优点?

答:(1) 视图能够简化用户的操作 (2) 视图使用户能以多种角度看待同一数据;(3) 视图为数据库提供了一定程度的逻辑独立性;(4) 视图能够对机密数据提供安全保护。

### 38、 NULL 是什么意思

答: NULL 这个值表示 UNKNOWN(未知):它不表示 "" (空字符串)。对 NULL 这个值的任何比较都会生产一个 NULL 值。您不能把任何值与一个 NULL 值进行比较,并在逻辑上希望获得一个答案。

使用 IS NULL 来进行 NULL 判断



### 39、主键、外键和索引的区别？

主键、外键和索引的区别

定义：

主键-唯一标识一条记录，不能有重复的，不允许为空

外键-表的外键是另一表的主键，外键可以有重复的，可以是空值

索引-该字段没有重复值，但可以有一个空值

作用：

主键-用来保证数据完整性

外键-用来和其他表建立联系用的

索引-是提高查询排序的速度

个数：

主键-主键只能有一个

外键-一个表可以有多个外键

索引-一个表可以有多个唯一索引

### 40、你可以用什么来确保表格里的字段只接受特定范围里的值？



答：Check 限制，它在数据库表格里被定义，用来限制输入该列的值。

触发器也可以被用来限制数据库表格里字段能够接受的值，但是这种办法要求触发器在表格里被定义，这可能会在某些情况下影响到性能。

## 41、说说对 SQL 语句优化有哪些方法？（选择几条）

- 1、Where 子句中：where 表之间的连接必须写在其他 Where 条件之前，那些可以过滤掉最大数量记录的条件必须写在 Where 子句的末尾.HAVING 最后。
- 2、用 EXISTS 替代 IN、用 NOT EXISTS 替代 NOT IN。
- 3、避免在索引列上使用计算
- 4、避免在索引列上使用 IS NULL 和 IS NOT NULL
- 5、对查询进行优化，应尽量避免全表扫描，首先应考虑在 where 及 order by 涉及的列上建立索引。
- 6、应尽量避免在 where 子句中对字段进行 null 值判断，否则将导致引擎放弃使用索引而进行全表扫描
- 7、应尽量避免在 where 子句中对字段进行表达式操作，这将导致引擎放弃使用索引而进行全表扫描

## 42、锁的优化策略

- 1、读写分离



2、分段加锁

3、减少锁持有的时间

4.多个线程尽量以相同的顺序去获取资源

不能将锁的粒度过于细化，不然可能会出现线程的加锁和释放次数过多，反而效率不如一次加一把大锁。

## 43、索引的底层实现原理和优化

B+ 树，经过优化的 B+ 树

主要是在所有的叶子结点中增加了指向下一个叶子节点的指针，因此 InnoDB 建议为大部分表使用默认自增的主键作为主索引。

## 44、什么情况下设置了索引但无法使用

1、以 “%” 开头的 LIKE 语句，模糊匹配

2、OR 语句前后没有同时使用索引

3、数据类型出现隐式转化（如 varchar 不加单引号的话可能会自动转换为 int 型）

## 45、实践中如何优化 MySQL

最好是按照以下顺序优化：



1、SQL 语句及索引的优化

2、数据库表结构的优化

3、系统配置的优化

4、硬件的优化

详细可以查看 [阿里 P8 架构师谈：MySQL 慢查询优化、索引优化、以及表等优化总结](#)

## 46、优化数据库的方法

1、选取最适用的字段属性,尽可能减少定义字段宽度,尽量把字段设置 NOTNULL,例如'省份'、'性别'最好适用 ENUM

2、使用连接(JOIN)来代替子查询

3、适用联合(UNION)来代替手动创建的临时表

4、事务处理

5、锁定表、优化事务处理

6、适用外键, 优化锁定表

7、建立索引

8、优化查询语句



## 47、简单描述 MySQL 中，索引，主键，唯一索引，联合索引的区别，对数据库的性能有什么影响（从读写两方面）

索引是一种特殊的文件(InnoDB 数据表上的索引是表空间的一个组成部分)，它们包含着对数据表里所有记录的引用指针。

普通索引(由关键字 KEY 或 INDEX 定义的索引)的唯一任务是加快对数据的访问速度。

普通索引允许被索引的数据列包含重复的值。如果能确定某个数据列将只包含彼此各不相同的值，在为这个数据列创建索引的时候就应该用关键字 UNIQUE 把它定义为一个唯一索引。也就是说，唯一索引可以保证数据记录的唯一性。

主键，是一种特殊的唯一索引，在一张表中只能定义一个主键索引，主键用于唯一标识一条记录，使用关键字 PRIMARY KEY 来创建。

索引可以覆盖多个数据列，如像 INDEX(columnA, columnB)索引，这就是联合索引。

索引可以极大的提高数据的查询速度，但是会降低插入、删除、更新表的速度，因为在执行这些写操作时，还要操作索引文件。

## 48、数据库中的事务是什么？

事务(transaction)是作为一个单元的一组有序的数据库操作。如果组中的所有操作都成功，则认为事务成功，即使只有一个操作失败，事务也不成功。如果所有操作完成，事务则提交，其修改将作用于所有其他数据库进程。如果一个操作失败，则事务将回滚，该事务所有操作的影响都将取消。





#### 事务特性：

- 1、原子性：即不可分割性，事务要么全部被执行，要么就全部不被执行。
- 2、一致性或可串性。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
- 3、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，
- 4、持久性。事务正确提交后，其结果将永久保存在数据库中，即使在事务提交后有了其他故障，事务的处理结果也会得到保存。

或者这样理解：

事务就是被绑定在一起作为一个逻辑工作单元的 SQL 语句分组，如果任何一个语句操作失败那么整个操作就被失败，以后操作就会回滚到操作前状态，或者是上有个节点。为了确保要么执行，要么不执行，就可以使用事务。要将有组语句作为事务考虑，就需要通过 ACID 测试，即原子性，一致性，隔离性和持久性。

## 49、SQL 注入漏洞产生的原因？如何防止？

SQL 注入产生的原因：程序开发过程中不注意规范书写 sql 语句和对特殊字符进行过滤，导致客户端可以通过全局变量 POST 和 GET 提交一些 sql 语句正常执行。

防止 SQL 注入的方式：

开启配置文件中的 `magic_quotes_gpc` 和 `magic_quotes_runtime` 设置

执行 sql 语句时使用 `addslashes` 进行 sql 语句转换



Sql 语句书写尽量不要省略双引号和单引号。

过滤掉 sql 语句中的一些关键词：update、insert、delete、select、\*。

提高数据库表和字段的命名技巧，对一些重要的字段根据程序的特点命名，取不易被猜到的。

## 50、为表中得字段选择合适得数据类型

字段类型优先级：整形>date,time>enum,char>varchar>blob,text

优先考虑数字类型，其次是日期或者二进制类型，最后是字符串类型，同级别得数据类型，应该优先选择占用空间小的数据类型