Python 中操作Mysql 步骤

#首先安装包-pymysql sudo pip install pymysql
#之后在程序中调用 from pymysql import *
''' connection 对象 用于建立与数据库的连接 创建对象：调用 connect()方法 ''' conn = connect(参数列表)
''' 参数列表：
host:连接 MySQL 主机，如果是本机则为”localhost“ port:连接 MySQL 主机端口，默认 3306
database:数据库名称
user：连接的用户名 password：连接的密码 charset:通信采用的编码方式，推荐采用 utf8
connection 对象方法 close() 关闭连接 commit() 提交 rollback() 回滚 cursor() 返回
cursor 对象，用于执行 sql 语句 例如：select,insert,update,delete
cs1 = conn.cursor()
cursor 对象方法
close() 关闭
execute(operation[,parameters])执行语句，返回受影响的行数，主要用于执行 insert、update、delete 语句，
也可以执行 create、alter、drop 等语句 fetchone()执行查询语句时，获取查询结果集的第一个行数据，返回一个元组 fetchall()执行查询时，获取结果集的所有行，一行构成一个元组，再将这些元组装入一个元组返回
cursor 对象属性
rowcount 只读属性，表示最近一次 execute()执行后受影响的行数 connection 获得当前连接对象
#例子
#创建 Connection 连接
conn = connect(host='localhost', port=3306, user='root', password='mysql', database='python1', charset='utf8')
#获得 Cursor 对象 cs = conn.cursor()
# 更新 #
sql = 'update students set name="刘邦" where id=6'
#删除
sql = 'delete from students where id=6'
#执行 select 语句，并返回受影响的行数：查询一条学生数据
sql = 'select id,name from students where id = 7'
# sql = 'SELECT id,name FROM students WHERE id = 7' count=cs.execute(sql)
#打印受影响的行数
print count

SQL 的 select 语句完整的执行顺序

1、from 子句组装来自不同数据源的数据；

2、where 子句基于指定的条件对记录行进行筛选；
3、group by 子句将数据划分为多个分组；
4、使用聚集函数进行计算；
5、使用 having 子句筛选分组；
6、计算所有的表达式；
7、select 的字段；
8、使用 order by 对结果集进行排序。
SQL语言不同于其他编程语言的最明显特征是处理代码的顺序。在大多数据库语言中，代码按编码顺序被处 理。但在 SQL语句中，第一个被处理的子句式 FROM，而不是第一出现的 SELECT。

SQL查询处理的步骤序号

(1) FROM <left_table>
(2) <join_type> JOIN <right_table>
(3) ON <join_condition>
(4) WHERE <where_condition>
(5) GROUP BY <group_by_list>
(6) WITH {CUBE | ROLLUP}
(7) HAVING <having_condition>
(8) SELECT
(9) DISTINCT
(10) ORDER BY <order_by_list>
(11) <TOP_specification> <select_list> 以上每个步骤都会产生一个虚拟表，该虚拟表被用作下一个步骤的输入。这些虚拟表对调用 者(客户端应用程 序或者外部查询)不可用。只有最后一步生成的表才会会给调用者。如果没有在 查询中指定某一个子句，将跳 过相应的步骤

逻辑查询处理阶段简介

1、 FROM：对 FROM 子句中的前两个表执行笛卡尔积(交叉联接)，生成虚拟表 VT1。
2、 ON：对 VT1 应用 ON 筛选器，只有那些使为真才被插入到 TV2。
3、 OUTER (JOIN):如果指定了 OUTER JOIN(相对于 CROSS JOIN 或 INNER JOIN)，保留表中未找到 匹配的行将作为外部行添加到 VT2，生成 TV3。如果 FROM 子句包含两个以上的 表，则对上一个联接生成 的结果表和下一个表重复执行步骤 1 到步骤 3，直到处理完所有的表 位置。
4、 WHERE：对 TV3 应用 WHERE 筛选器，只有使为 true 的行才插入 TV4。
5、 GROUP BY：按 GROUP BY 子句中的列列表对 TV4 中的行进行分组，生成 TV5。
6、 CUTE|ROLLUP：把超组插入 VT5，生成 VT6。
7、 HAVING：对 VT6 应用 HAVING 筛选器，只有使为 true 的组插入到 VT7。
8、 SELECT：处理 SELECT 列表，产生 VT8。
9、 DISTINCT：将重复的行从 VT8 中删除，产品 VT9。
10、 ORDER BY：将VT9中的行按ORDER BY子句中的列列表顺序，生成一个游标(VC10)。

11、 TOP：从 VC10 的开始处选择指定数量或比例的行，生成表 TV11，并返回给调用者。 where 子句中 的条件书写顺序

说一下 Mysql 数据库存储的原理

储存过程是一个可编程的函数，它在数据库中创建并保存。它可以有 SQL 语句和一些特殊的控制结构组成。

当希望在不同的应用程序或平台上执行相同的函数，或者封装特定功能时，存储过程是非常有 用的。数据库中 的存储过程可以看做是对编程中面向对象方法的模拟。它允许控制数据的访问方式。 存储过程通常有以下优点：

1、存储过程能实现较快的执行速度

2、存储过程允许标准组件是编程。

3、存储过程可以用流程控制语句编写，有很强的灵活性，可以完成复杂的判断和较复杂的运算。

4、存储过程可被作为一种安全机制来充分利用。

5、存储过程能够减少网络流量

如何对查询命令进行优化

a. 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。

b. 应尽量避免在 where 子句中对字段进行 null 值判断，避免使用!=或<>操作符，避免使用 or 连接条 件，或在 where 子句中使用参数、对字段进行表达式或函数操作，否则会导致权标扫描

c. 不要在 where 子句中的“=”左边进行函数、算术运算或其他表达式运算，否则系统将可能无 法正确使用 索引。

d. 使用索引字段作为条件时，如果该索引是复合索引，那么必须使用到该索引中的第一个字段作为 条件时才能保证系统使用该索引，否则该索引将不会被使用。

e. 很多时候可考虑用exists 代替 in。

f. 尽量使用数字型字段。

g. 尽可能的使用 varchar/nvarchar 代替 char/nchar

h. 任何地方都不要使用 select from t ，用具体的字段列表代替“”，不要返回用不到的任何字段。

i. 尽量使用表变量来代替临时表。

j. 避免频繁创建和删除临时表，以减少系统表资源的消耗。

k. 尽量避免使用游标，因为游标的效率较差。

l. 在所有的存储过程和触发器的开始处设置 SET NOCOUNT ON ，在结束时设置 SET NOCOUNT OFF。

m. 尽量避免大事务操作，提高系统并发能力。

n. 尽量避免向客户端返回大数据量，若数据量过大，应该考虑相应需求是否合理。

数据库的优化

1.优化索引、SQL 语句、分析慢查询；

2.设计表的时候严格根据数据库的设计范式来设计数据库；

3.使用缓存，把经常访问到的数据而且不需要经常变化的数据放在缓存中，能节约磁盘 IO

4.优化硬件；采用 SSD，使用磁盘队列技术(RAID0,RAID1,RDID5)等
5.采用 MySQL 内部自带的表分区技术，把数据分层不同的文件，能够提高磁盘的读取效率；
6.垂直分表；把一些不经常读的数据放在一张表里，节约磁盘 I/O；
7.主从分离读写；采用主从复制把数据库的读操作和写入操作分离开来；
8.分库分表分机器（数据量特别大），主要的的原理就是数据路由；
9.选择合适的表引擎，参数上的优化
10.进行架构级别的缓存，静态化和分布式；
11.不采用全文索引；
12.采用更快的存储方式，例如 NoSQL 存储经常访问的数据**。

Mysql 数据库如何分区、分表

分表可以通过三种方式：Mysql 集群、自定义规则和 merge 存储引擎。
分区有四类：
RANGE 分区：基于属于一个给定连续区间的列值，把多行分配给分区。
LIST 分区：类似于按 RANGE 分区，区别在于 LIST 分区是基于列值匹配一个离散值集合中的某个 值来进 行选择。
HASH 分区：基于用户定义的表达式的返回值来进行选择的分区，该表达式使用将要插入到表中的 这些行的列 值进行计算。这个函数可以包含 MySQL 中有效的、产生非负整数值的任何表达式。
KEY 分区：类似于按 HASH 分区，区别在于 KEY 分区只支持计算一列或多列，且 MySQL 服务器提供其 自身的哈希函数。必须有一列或多列包含整数值。

Sql 注入是如何产生的，如何防止

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。产生 Sql 注入。
下面是防止办法：
a. 过滤掉一些常见的数据库操作关键字，或者通过系统函数来进行过滤。
b. 在 PHP 配置文件中将 Register_globals=off;设置为关闭状态
c. SQL 语句书写的时候尽量不要省略小引号(tab 键上面那个)和单引号
d. 提高数据库命名技巧，对于一些重要的字段根据程序的特点命名，取不易被猜到的
e. 对于常用的方法加以封装，避免直接暴漏 SQL 语句
f. 开启 PHP 安全模式：Safe_mode=on;
g. 打开 magic_quotes_gpc 来防止 SQL 注入
h. 控制错误信息：关闭错误提示信息，将错误信息写到系统日志。
i. 使用 mysqli 或 pdo 预处理。
Mysql 数据库中怎么实现分页
select * from table limit (start-1)*limit,limit; 其中 start 是页码，limit 是每页显示的条数。

sql 语句怎么看效率

SQLServer2005-->新建一个查询-->输入语句 SELECT * FROM Person.Contact 执行(F5)-->Ctrl+L

优化数据库？提高数据库的性能

1.对语句的优化

用程序中，保证在实现功能的基础上，尽量减少对数据库的访问次数； 通过搜索参数，尽量减少对表的访问行 数,最小化结果集，从而减轻网络负担；

能够分开的操作尽量分开处理，提高每次的响应速度；在数据窗口使用 SQL 时，尽量把使用 的索引放在选择 的首列；算法的结构尽量简单；

在查询时，不要过多地使用通配符如 SELECT * FROM T1 语句，要用到几列就选择几列如：

SELECT COL1,COL2 FROM T1；

在可能的情况下尽量限制尽量结果集行数如：SELECT TOP 300 COL1,COL2,COL3 FROM T1,因为某些情 况下用户是不需要那么多的数据的。 不要在应用中使用数据库游标，游标是非常有用的工具，但比使用常规的、面向集的 SQL 语 句需要更大的开 销；按照特定顺序提取数据的查找。

2. 避免使用不兼容的数据类型 例如 float 和 int、char 和 varchar、binary 和 varbinary 是不 兼容的。

数据类型的不兼容可能使优化器无法执行一些本来可以进行的优化操作。 例如: SELECT name FROM employee WHERE salary > 60000

在这条语句中,如 salary 字段是 money 型的,则优化器很难对其进行优化,因为 60000 是个整型数。我 们应当在编程时将整型转化成为钱币型,而不要等到运行时转化。 若在查询时强制转换，查询速度会明显减慢。

3.避免在 WHERE 子句中对字段进行函数或表达式操作。 若进行函数或表达式操作，将导致引擎放弃使用索 引而进行全表扫描。

4.避免使用!=或＜＞、IS NULL 或 IS NOT NULL、IN ，NOT IN 等这样的操作符

5.尽量使用数字型字段

6.合理使用 EXISTS,NOT EXISTS 子句。

7.尽量避免在索引过的字符数据中，使用非打头字母搜索。

8.分利用连接条件

9.消除对大型表行数据的顺序存取

10. 避免困难的正规表达式

11. 使用视图加速查询

12. 能够用 BETWEEN 的就不要用 IN

13. DISTINCT 的就不用 GROUP BY

14. 部分利用索引

15. 能用 UNION ALL 就不要用 UNION

16. 不要写一些不做任何事的查询

17. 尽量不要用 SELECT INTO 语句

18. 必要时强制查询优化器使用某个索引
19. 虽然 UPDATE、DELETE 语句的写法基本固定，但是还是对 UPDATE 语句给点建议：
a) 尽量不要修改主键字段。
b) 当修改 VARCHAR 型字段时，尽量使用相同长度内容的值代替。 c) 尽量最小化对于含有 UPDATE 触发器的表的 UPDATE 操作。 d) 避免 UPDATE 将要复制到其他数据库的列。
e) 避免 UPDATE 建有很多索引的列。
f) 避免 UPDATE 在 WHERE 子句条件中的列。

提取数据库中倒数 10 条数据

select top (10) * from table1 order by id desc。

Mysql 数据库的操作

修改表-修改字段，重命名版：
alter table 表名 change 原名 新名 类型及约束；
alter table students change birthday birth datetime not null;
修改表-修改字段，不重名版本：
alter table 表名 modify 列名 类型和约束；
alter table students modify birth date not null
全列插入：
insert into 表名 values(...)
nsert into students values(0,"郭靖", 1,"内蒙","2017-6");
部分插入： 值的顺序与给出的列顺序对应：
insert into students(name, birthday) values("黄蓉","2017-8");
修改：update 表名 set 列 1=值 1，列 2=值 2.。。where
update students set gender=0, homwtown="古墓", where id = 5;
备份：mysqldump -uroot -p 数据库名 》 python.sql
恢复：mysql -uroot -p 数据库名 < python.sql

用select语句输出每个城市中心距离市中心大于 20km酒店数
select count（hotel）i from hotel_table where distance >20 group by city
请拿出 B 表中的 accd，(A 表中和 B 表中的一样的数据)
select * from B inner join on B.name = A.name
Mysql 怎么限制 IP 访问
grant all privileges on . to ‘数据库中用户名’@’ip 地址’ identified by ‘数据库密码’;

关系型数据库中，表和表之间有左连接，内连接，外连接，分别解释下他们的含义和区别

内连接查询：查询的结果为两个表匹配到的数据。 右接查询：查询的结果为两个表匹配到的数据， 右表特有的数据，对于左表中不存在的数据使用 null 填充 左连接查询：查询的结果为两个表匹配到的数据，左表特有的数据，对于右表中不存在的数据使用 null 填 充。

MongoDB

Mongo 数据库的一些基本操作命令

a) create database name; 创建数据库

b) use databasename; 选择数据库

c) drop database name 直接删除数据库，不提醒

d) show tables; 显示表

e) describe tablename; 表的详细描述

f) select 中加上 distinct 去除重复字段

g) mysqladmin drop databasename 删除数据库前，有提示。

h) 显示当前 mysql 版本和当前日期

i) select version(),current_date;