1. mysql/bin下命令的使用
   1. 实例启停（mysqld\_safe| mysqladmin）

* 启动实例

./mysqld\_safe --defaults-file=配置文件路径 --ledir=mysql主文件目录 &

* 指定配置文件，指定运行目录，并后台运行

./mysqld\_safe --defaults-file=/home/mysql/3201/cnf/my.3201.cnf --ledir=/home/mysql/3201/mysql/bin &

* 关闭

./mysqladmin -u root -p shutdown

* 1. 通过客户端连接服务端（mysql）
* 使用指定IP、端口的方式连接

./mysql -uroot -p -h127.0.0.1 -P3306

* 使用sock的方式连接(sock文件位置由配置文件指定)

./mysql -uroot -p --socket=/home/mysql/conn/mysql.3201.sock

* 1. 数据导出(mysqldump)

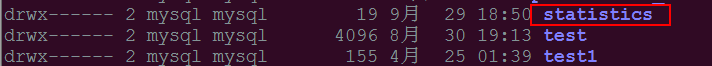
mysqldump -u 用户名 -p 数据库名 表名 -h主机地址/主机名 -P端口号 > 导出的文件名

1. Database
   1. 让database从本地存储链接到其他存储介质

创建Database

create database 库名;

mysql会在文件系统上创建目录



重命名该目录

mv statistics/ statistics\_bak

创建连接

ln -s 源地址 目标地址

源地址通常为存储上的地址，目标地址为database本地地址

1. 2018-01-23 总行MySQL培训课件学习
   1. 全局参数设置

key\_buffer\_size：

MyISAM索引缓冲，根据(key reads/key\_read\_requests)判断

innodb\_buffer\_pool\_size：

InnoDB数据、索引、日志缓冲最重要的引擎参数，根据(hit riatos和FILE I/O)判断

wait\_time\_out：

线程连接的超时时间，尽量不要设置的很大

max\_connections：

允许服务器最大连接数，尽量不要设置很大

thread\_concurrency：

线程并发利用数量(cpu+disk)\*2，根据（OS中显示的请求队列和tickets）判断

* 1. 线程参数设置

sort\_buffer\_size

获得更快的--ORDER BY, GROUP BY ,SELECT DISTINCT, UNION DISTINCT

read\_rnd\_buffer\_size

当根据键进行分类操作时，获得更快的——ORDER BY

join\_buffer\_size

Join连接使用全表扫描连接的缓冲大小，根据（Select\_full\_join）判断

read\_buffer\_size

全表扫描时，为查询预留的缓冲大小，根据(Select\_scan)判断

tmp\_table\_size

临时内存表超出设置，转化为磁盘表，根据(Created\_tmp\_disk\_table)判断

注意：线程参数设置的小影响性能，设置的大会导致服务器swap

* 1. 元数据信息
     1. information\_schema

数据表信息：

COLUMNS：所有表和视图中的列信息

ENGINES：所支持的存储引擎

SCHEMATA：所有的database使用的字符集和排序依据

TABLES：所有数据表信息

VIEWS：所有视图信息

分区信息：

PARTITIONS：表分区信息

权限信息：

COLUMN\_PRIVILEGES：列访问权限

SCHEMA\_PRIVILEGES：database的访问权限

TABLE\_PRIVILEGES：表访问权限

USER\_PRIVILEGES：用户的访问权限

变量和状态信息：

KEY\_COLUMN\_USAGE：主键、外键等约束

CLOBAL\_STATUS：数据库所有连接信息

GLOBAL\_VARIABLES：数据库参数信息

PLUGINS：插件信息

PROCESSLIST：mysql正在运行的进程信息

SESSION\_STATUS：连接状态信息

SESSION\_VARIABLES：连接参数值

InnoDB相关：

INNODB\_LOCKS：每个Innodb事务请求或拥有锁

INNODB\_LOCK\_WAITS：InnoDB事务锁定一条或多条记录

INNODB\_TRX：innodb中正在执行的每个事务

INNODB\_BUFFER\_POOL\_STATS:Innodb缓冲状态

* 1. show命令

SHOW DATABASES;

SHOW INDEX;

SHOW CREATE TABLE;

SHOW CHARACTER SET;

SHOW TABLES;

SHOW COLLATION;

SHOW OPEN TABLES;

SHOW PROCESSLIST;

SHOW COLUMNS;

SHOW TABLE STATUS;

SHOW ENGINE INNODB STATUS;——监控死锁

* 1. InnoDB

表空间tablespace，由系统表空间和独立表空间组成

段segment，包括数据段、索引段、回滚段等

区extent，每个区大小1M，默认情况下由连续的64个page组成

页page，默认16KB，可调整innodb\_page\_size大小4K、8K、16K

行row

INNODB\_BUFFER\_POOL\_SIZE：innodb的主要缓存区域，存放innodb表的数据、索引、undo信息，建议不超过物理内存的80%(独立数据库服务器)

INNODB\_FILE\_PER\_TABLE：是否采用独立表空间

* 1. 备份与恢复

按类型划分：

冷备份：停服务，不区分存储引擎，系统命令CP

暖备份：在线备份，需要上读锁，备份期间无法写入

热备份：在线备份，不影响业务读写，通过内置的备份工具

按备份方式划分：

* 逻辑备份(mysqldump)

备份语法：

mysqldump -u 用户名 -p database名称 表名称> 导出文件名

恢复语法：

mysql -u 用户名 -p < 导入文件名

将数据和表结构转换成SQL语句

可移植性好

造成热数据置换，影响服务性能

mysql服务必须运行

远程本地均可操作

备份与恢复速度比物理备份慢

备份的结果集通常大于实际数据文件

支持所有存储引擎

* 物理备份

拷贝数据文件

区分存储引擎

备份与恢复速度快于逻辑备份

必须本地操作

不一定能跨平台恢复

需要停机备份

* 1. 数据的导入导出

导出：

SELECT \* INTO OUTFILE '/tmp/data.txt' FROM TABLE;

导入：

LOAD DATA INFILE '/tmp/data.txt' INTO TABLE xxx;