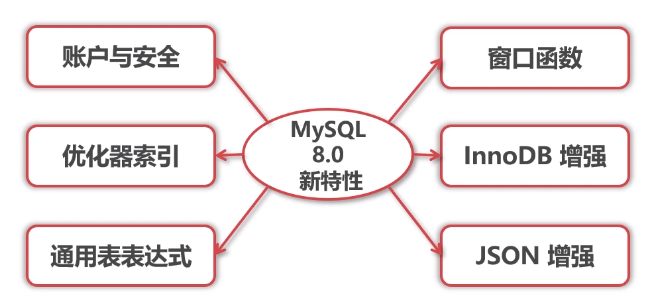
# 8.0新特性



# 账户与安全

## 用户创建和授权

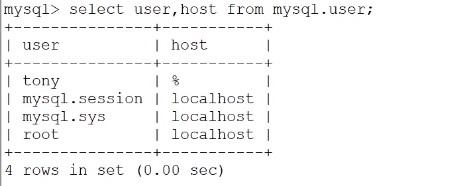
MySQL8.0用户创建和授权分开执行：

create user ‘tony’@’%’ identified by ‘Tony@2018’

grant all privileges on \*.\* to ‘tony’@’%’;

MySQL5.7用户创建和授权可以使用grant语句一次性完成：

grant all privileges on \*.\* to ‘tony’@’%’ identified by ‘passwd’;

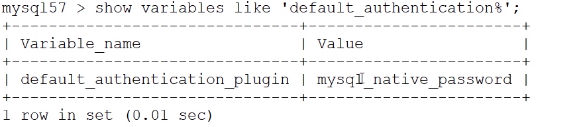


## 认证插件更新

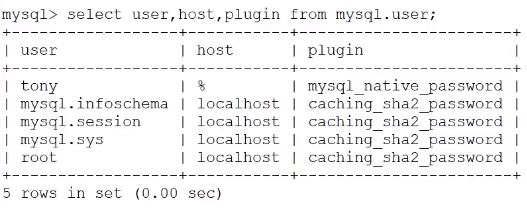
MySQL8.0中默认的身份认证插件是caching\_sha2\_password，替代了之前的mysql\_native\_password。

用户可以通过defalt\_authentication\_plugin或mysql.user表看到这种变化：

MySQL5.7：



MySQL8.0：



注：如果服务端mysql升级了，但是客户端没有升级，可能使用新的插件会导致连接错误。可以修改系统配置文件修改为原来的认证。

## 密码管理

MySQL8.0开始允许限制重复使用以前的密码：

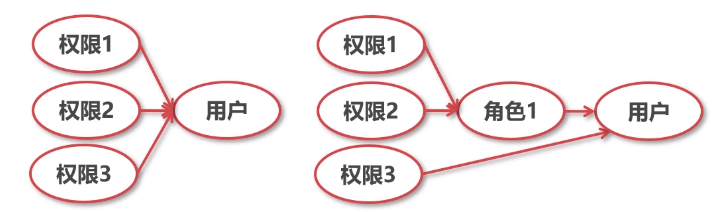
password\_history=3 //新密码不能和最近3次使用的相同

password\_reuse\_internal=90 //新密码不能和最近90天使用的密码相同

password\_require\_current=ON //修改密码时需要提供当前密码

## 角色管理

MySQL8.0提供了角色管理的新功能，角色是一组权限的集合。



# 优化器索引

## 隐藏索引

MySQL8.0开始支持隐藏索引（invisible index），不可见索引。

隐藏索引不会被优化器使用，但仍然需要进行维护。

应用场景：软删除、灰度发布。

软删除：比如我们需要测试不使用某个索引对系统性能的影响，以前的方案是删除当前索引，如果有影响，需要这个索引就必须重新建立，这样维护成本太高了。在8.0版本可以将这个待测试的索引设置为隐藏索引，这样实际查询中就不会使用这个索引了，测试后如果真需要删除再执行删除索引操作。

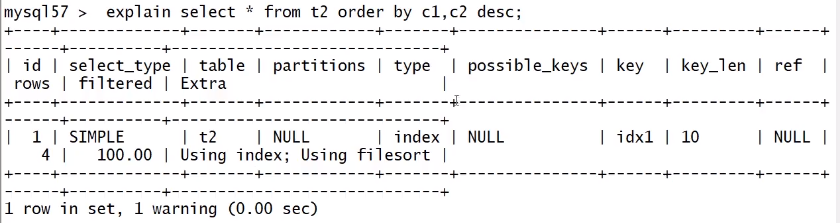
## 降序索引

MySQL8.0开始真正支持降序索引（descending index）。

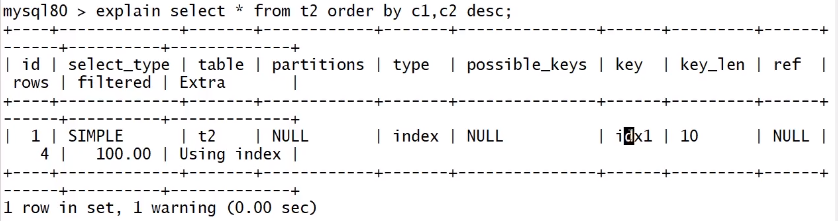
只有InnoDB存储引擎支持降序索引，只支持BTREE降序索引。

MySQL8.0不再对GROUP BY操作进行隐式排序。

MySQL5.7：



MySQL8.0：



## 函数索引

MySQL8/0/13开始支持在索引中使用函数（表达式）的值。

支持降序索引，支持JSON数据的索引。

函数索引基于虚拟列功能实现。

# 通用表达式

MySQL8.0开始支持通用表表达式（CTE），即WITH子句。

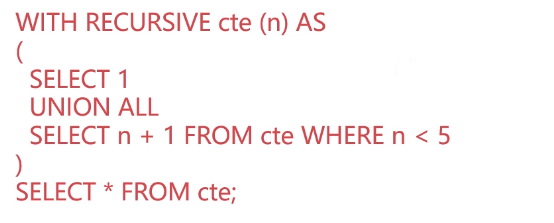
## 非递归CTE



注：CTE更加清晰，后面可以任意实用前面定义的CTE表达式，更加方便，类似增加了一个编程功能的扩展。

## 递归CTE

递归CTE在查询中引用自己的定义，实用RECURSIVE表示。



## 递归限制

递归表达式的查询中需要包含一个终止递归的条件：

cte\_max\_recursion\_depth：最大递归深度/调用次数

max\_execution\_time：SQL最长执行时间（没有定义最大递归深度的时候通过这个限制）

# 窗口函数

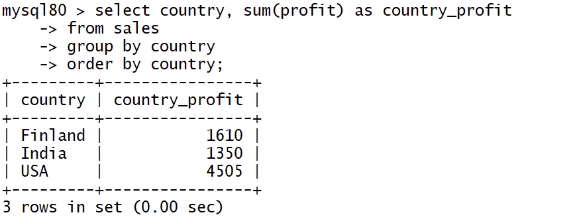
## 概述

MySQL8.0支持窗口函数（Window Function），也称分析函数。

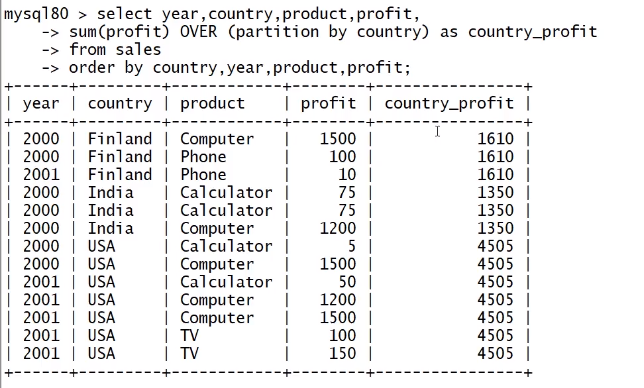
窗口函数与分组聚合函数类似，但是每一行数据都生成一个结果。

聚合窗口函数：SUM/AVG/COUNT/MAX/MIN等等。

常规操作：



窗口函数：



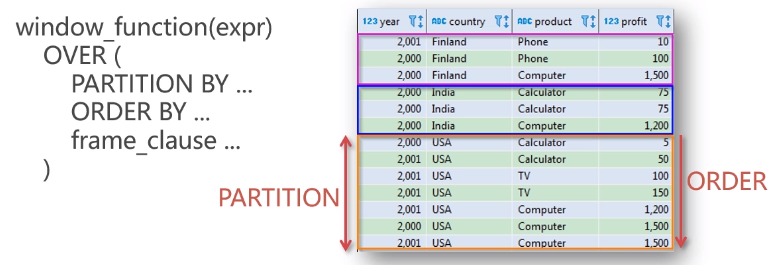
## 专用窗口函数

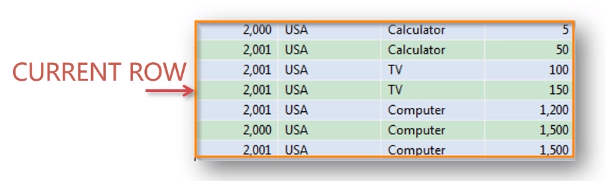
ROW\_NUMBER()/RANK()/DENSE\_RANK()/PERCENT\_RANK()

FIRST\_VALUE()/LAST\_VALUE()/LEAD()/LAG()

CUME\_DIST()/NTH\_VALUE()/NTILE()

## 窗口





# InnoDB增强

## 集成数据字典

MySQL8.0删除了之前版本的元数据文件，例如.frm，.opt等。

将系统表（mysql）和数据字典全部改为InnoDB存储引擎。

简化了INFORMATION\_SCHEMA的实现，提高了访问性能。

提供了序列化字典信息（SDI）的支持，以及ibd2sdi工具。

数据字典使用上的差异，例如innodb\_read\_only影响所有的存储引擎；数据字典表不可见，不能直接查询和修改。

## 原子DDL操作

MySQL8.0开始支持原子DDL操作，其中与表相关的原子DDL只支持InnoDB存储引擎。

一个原子DDL操作内容包括：更新数据字典，存储引擎层的操作，在binlog中记录DDL操作。

支持与表相关的DDL：数据库、表空间、表、索引的CREATE、ALTER、DROP以及TRUNCATE TABLE。

支持的其他DDL：存储程序、触发器、视图、UDF（用户自定义函数）的CREATE、DROP以及ALTER语句。

支持账户管理相关的DDL：用户和角色的CREATE、ALTER、DROP以及适用的RENAME，以及GRANT和REVOKE语句。

## 自增列持久化

MySQL5.7以及早期版本，InnoDB自增列计数器（AUTO\_INCREMENT）的值只存储在内存中。

MySQL8.0每次变化时将自增计数器的最大值写入redo log，同时在每次检查点将其写入引擎私有的系统表。

解决了长期以来的自增字段值可能重复的bug。

## 死锁检查控制

MySQL8.0（MySQL5.7.15）增加了一个新的动态变量（innodb\_deadlock\_detect），用于控制系统是否执行InnoDB死锁检查。

对于高并发的系统，禁用死锁检查可能带来性能的提高。

## 锁定语句选项

SELECT…FOR SHARE和SELECT…FOR UPDATE中支持NOWAIT、SKIP LOCKED选项：

对于NOWAIT，如果请求的行被其他事务锁定时，语句立即返回。

对于SKIP LOCKED，从返回的结果集中移除被锁定的行。

## 其他改进功能

# JSON增强

## 内联路径操作符

## JSON聚合函数

## JSON实用函数

## JSON合并函数

## JSON表函数