# 一、读写、只读、DML、DDL 区别

在MySQL中，有不同的用户权限级别，包括读写权限、只读权限、DML权限和DDL权限。下面是它们之间的区别：

读写权限（Read/Write）：具有读写权限的用户可以执行查询（SELECT）和更新操作（INSERT、UPDATE、DELETE）等写操作。这些用户可以读取和修改数据库中的数据。

只读权限（Read-only）：具有只读权限的用户仅能执行查询操作（SELECT），不能进行任何写操作。他们可以读取数据库中的数据，但无法进行插入、更新或删除等写操作。

DML权限（Data Manipulation Language）：DML权限允许用户执行数据操作语言（Data Manipulation Language）的操作，例如INSERT、UPDATE和DELETE语句。这些权限允许用户修改数据表中的数据。

DDL权限（Data Definition Language）：DDL权限允许用户执行数据定义语言（Data Definition Language）的操作，例如CREATE、ALTER和DROP语句。这些权限允许用户创建、修改和删除数据库、数据表、视图、索引等数据库结构。

总结起来，读写权限允许用户读取和修改数据，只读权限只允许读取数据，DML权限允许执行数据操作语言（如INSERT、UPDATE、DELETE），而DDL权限允许执行数据定义语言（如CREATE、ALTER、DROP）。根据特定的需求和安全性要求，可以为MySQL用户分配适当的权限级别。

# **二、用户权限层级**

1)全局层级：全局权限适用于一个给定MySQL Server中的所有数据库，这些权限存储在mysql.user表中。

GRANT ALL ON \*.\* TO 'user'@'host'; # \*.\* 表示数据库库的所有库和表，对应权限存储在mysql.user表中

2)数据库层级：数据库权限适用于一个给定数据库中的所有目标，这些权限存储在mysql.db表中。

GRANT ALL ON mydb.\* TO 'user'@'host'; #mydb.\* 表示mysql数据库下的所有表，对应权限存储在mysql.db表中

3)表层级：表权限适用于一个给定表中的所有列，这些权限存储在mysql.tables\_priv表中。

GRANT ALL ON mydb.mytable TO 'user'@'host'; #mydb.mytable 表示mysql数据库下的mytable表，对应权限存储在mysql.tables\_priv表

4)列层级：列权限使用于一个给定表中的单一列，这些权限存储在mysql.columns\_priv表中。

GRANT ALL (col1， col2， col3) ON mydb.mytable TO 'user'@'host'; #mydb.mytable 表示mysql数据库下的mytable表， col1, col2, col3表示mytable表中的列名

5)子程序层级：CREATE ROUTINE、ALTER ROUTINE、EXECUTE和GRANT权限适用于已存储的子程序。这些权限可以被授予为全局层级和数据库层级。而且，除了CREATE ROUTINE外，这些权限可以被授予子程序层级，并存储在mysql.procs\_priv表中。

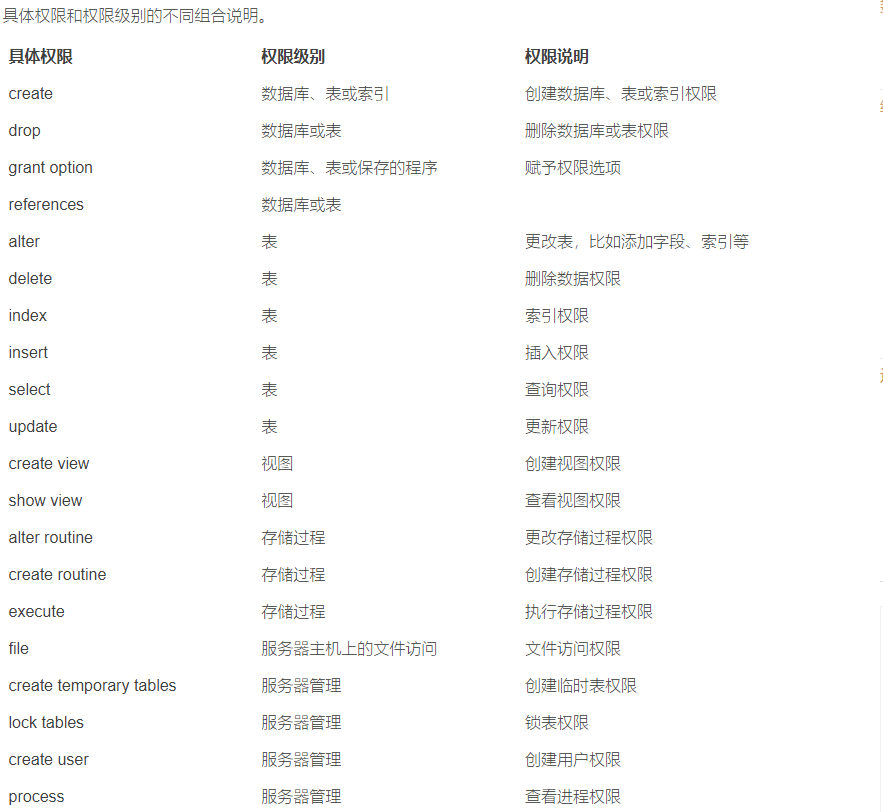
GRANT EXECUTE ON PROCEDURE mydb.myproc TO 'user'@'host'; #mydb.mytable 表示mysql数据库下的mytable表，PROCEDUR表示存储过程

# **三、MySQL权限简单分类**

1）数据权限 分为：库、表和字段三种级别

2）管理权限 主要是管理员要使用到的权限，包括：数据库创建，临时表创建、主从部署、进程管理等

3）程序权限 主要是触发器、存储过程、函数等权限。



# **四、用户管理实践**

## 1）用户创建

1、通过create user语句创建用户

create user 'USERNAME'@'HOST' identified by 'PASSWORD';

通过grant语句创建新用户

GRANT语句是添加新用户并授权它们访问MySQL对象的首选方法，其语法格式为：

grant all on DB\_NAME.TABLE\_NAME to 'USERNAME'@'HOST' identified by 'PASSWORD';

## 2）用户删除

mysql> drop user 'USERNAME'@'HOST';

# 删除MySQL默认的无用账户;

mysql> drop user 'root'@'localhost.localdomain';

# 删除MySQL默认的无用账户;

mysql> drop user 'root'@'127.0.0.1';

## 3）更改用户名

mysql> rename user OLD\_NAME to NEW\_NAME;

## 4）修改用户密码

通过mysqladmin工具

# 给root@localhost用户登录mysql设置密码为"redhat";

$ mysqladmin -u root -h localhost password "redhat"

# 修改root@localhost用户登录mysql数据库的密码;

$ mysqladmin -u root -h localhost password "new passwd" -p "old passwd"

通过直接修改mysql.user表的用户记录

# MySQL 5.6

mysql> update mysql.user set password=PASSWORD('redhat') where user='root';

mysql> flush privileges;

# MySQL 5.7

mysql> update mysql.user set authentication\_string=PASSWORD('redhat') where user='root';

mysql> flush privileges;

set password语句

mysql> set password for 'root'@'localhost'=PASSWORD('redhat');

mysql> flush privileges;

ALTER USER语句(MYSQL5.7版本)

mysql> use mysql

mysql> alter user root@'localhost' identified by '123456';

mysql> flush privileges;

## 5）查看用户权限

show grants for [username]@[hostname]

## 6）授权语句

在MySQL中，可以使用GRANT语句为用户授权不同级别的权限。以下是具体的授权语句示例：

1、授予读写权限：

GRANT INSERT, UPDATE, DELETE, SELECT ON database\_name.table\_name TO 'username'@'localhost';

2、授予只读权限：

GRANT SELECT ON database\_name.table\_name TO 'username'@'localhost';

3、授予DML权限：

GRANT INSERT, UPDATE, DELETE ON database\_name.table\_name TO 'username'@'localhost';

4、授予DDL权限：

GRANT CREATE, ALTER, DROP ON database\_name.\* TO 'username'@'localhost';