MySql启动、关闭

以管理员权限打开cmd

//启动

net start mysql80

//关闭

net stop mysql80

客户端连接

法一：MySQL提供的客户端命令行工具

法二：系统自带的命令行工具执行指令

mysql [-h 127.0.0.1] [-p 3306] -u root -p

-h：IP地址 //可省略

-p：端口 //可省略

简化为:

mtsql -u root -p

PS:在任意位置使用这种方式，得配置PATH环境变量

关系型数据库(RDBMS)

概念：建立在关系模型基础上，由多张相互连接的二维表组成的数据库。

特点：

A. 使用表存储数据，格式统一，便于维护。

B. 使用SQL语言操作，标准统一，使用方便。

数据模型

MySQL是关系型数据库，是基于二维表进行数据存储的

SQL通用语法

1). SQL语句可以单行或多行书写，以分号结尾。

2). SQL语句可以使用空格/缩进来增强语句的可读性。

3). MySQL数据库的SQL语句不区分大小写，关键字建议使用大写。

4). 注释：

单行注释：-- 注释内容 或 # 注释内容

多行注释：/\* 注释内容 \*/

SQL分类

SQL语句，根据其功能，主要分为四类：

DDL Data Definition Language 数据定义语言，用来定义数据库对象(数据库，表，字段)

DML Data Manipulation Language 数据操作语言，用来对数据库表中的数据进行增删改

DQL Data Query Language 数据查询语言，用来查询数据库中表的记录

DCL Data Control Language 数据控制语言，用来创建数据库用户、控制数据库 访问 权限

数据库操作

1). 查询所有数据库 show databases ;

2). 查询当前数据库 select database() ;

3). 创建数据库 create database [ if not exists ] 数据库名 [ default charset 字符集 ] [ collate 排序规则 ] ;

eg:

create database itcast;

create database if not extists itcast;

create database itheima default 1 charset utf8mb4;

4). 删除数据库 drop database [ if exists ] 数据库名 ;

5). 切换数据库 use 数据库名 ;

表操作

表操作-查询创建

1). 查询当前数据库所有表 show tables;

2). 查看指定表结构 desc 表名 ;

3). 查询指定表的建表语句 show create table 表名 ;

4). 创建表结构 CREATE TABLE 表名(

字段1 字段1类型 [ COMMENT 字段1注释 ],

字段2 字段2类型 [COMMENT 字段2注释 ],

字段3 字段3类型 [COMMENT 字段3注释 ],

......

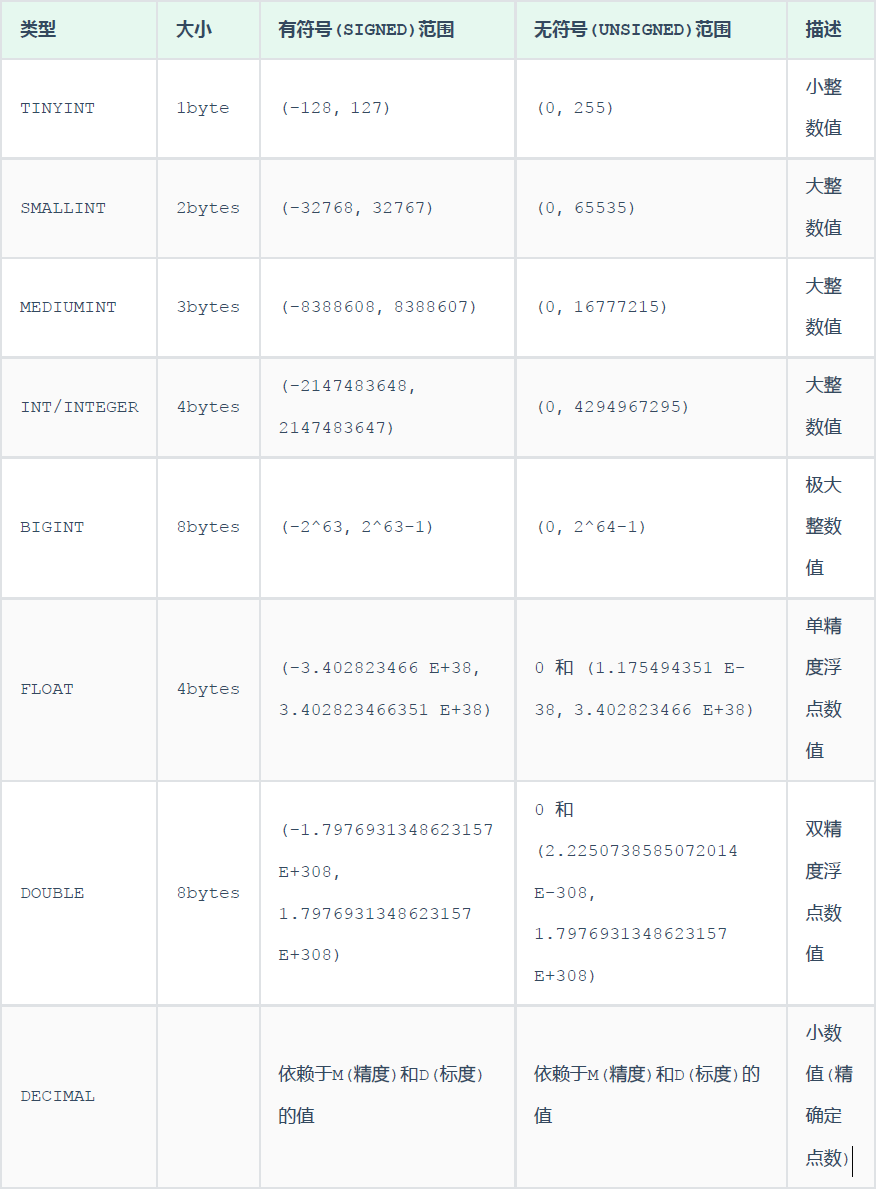
字段n 字段n类型 [COMMENT 字段n注释 ]

) [ COMMENT 表注释 ] ;

表操作-数据类型

MySQL中的数据类型有很多，主要分为三类：数值类型、字符串类型、日期时间类型。

1). 数值类型



2). 字符串类型



char 与 varchar 都可以描述字符串，char是定长字符串，指定长度多长，就占用多少个字符，和字段值的长度无关 。而varchar是变长字符串，指定的长度为最大占用长度 。相对来说char的性能会更高些。

3). 日期时间类型



表操作-修改

1). 添加字段

ALTER TABLE 表名 ADD 字段名 类型 (长度) [ COMMENT 注释 ] [ 约束 ];

2). 修改数据类型

ALTER TABLE 表名 MODIFY 字段名 新数据类型 (长度);

3). 修改字段名和字段类型

ALTER TABLE 表名 CHANGE 旧字段名 新字段名 类型 (长度) [ COMMENT 1 注释 ] [ 约束 ];

4). 删除字段

ALTER TABLE 表名 DROP 字段名;

5). 修改表名

ALTER TABLE 表名 RENAME TO 新表名;

表操作-删除

1). 删除表

DROP TABLE [ IF EXISTS ] 表名;

2). 删除指定表, 并重新创建表

TRUNCATE TABLE 表名;

DML

DML英文全称是Data Manipulation Language(数据操作语言)，用来对数据库中表的数据记录进行增、删、改操作。

添加数据（INSERT）

修改数据（UPDATE）

删除数据（DELETE）

添加数据

1). 给指定字段添加数据

INSERT INTO 表名 (字段名1, 字段名2, ...) VALUES (值1, 值2, ...);

2). 给全部字段添加数据

INSERT INTO 表名 VALUES 1 (值1, 值2, ...);

3). 批量添加数据

INSERT INTO 表名 (字段名1, 字段名2, ...) VALUES (值1, 值2, ...), (值1, 值2, ...), (值1, 值2, ...) ;

INSERT INTO 表名 VALUES (值1, 值2, ...), (值1, 值2, ...), (值1, 值2, ...) ;

注意事项:

插入数据时，指定的字段顺序需要与值的顺序是一一对应的。

字符串和日期型数据应该包含在引号中。

插入的数据大小，应该在字段的规定范围内。

修改数据

UPDATE 表名 SET 字段名1 = 值1 , 字段名 = 值2 , .... 1 [ WHERE 条件 ] ;

注意事项:

修改语句的条件可以有，也可以没有，如果没有条件，则会修改整张表的所有数据。

删除数据

删除数据的具体语法为：

DELETE FROM 表名 [ WHERE 条件 ] ;

注意事项:

DELETE 语句的条件可以有，也可以没有，如果没有条件，则会删除整张表的所有数

据。

DELETE 语句不能删除某一个字段的值(可以使用UPDATE，将该字段值置为NULL即

可)。

当进行删除全部数据操作时，datagrip会提示我们，询问是否确认删除，我们直接点击

Execute即可。

DQL

DQL 查询语句，语法结构如下：

SELECT

字段列表

FROM

表名列表

WHERE

条件列表

GROUP BY

分组字段列表

HAVING

分组后条件列表

ORDER BY

排序字段列表

LIMIT

分页参数

在基本查询的DQL语句中，不带任何的查询条件，查询的语法如下：

1). 查询多个字段

SELECT 字段1, 字段2, 字1 段3 ... FROM 表名 ;

SELECT \* FROM 表名 ;

注意 : \* 号代表查询所有字段，在实际开发中尽量少用（不直观、影响效率）。

2). 字段设置别名

SELECT 字段1 [ AS 别名1 ] , 字段2 [ AS 别名2 ] ... FROM 表名;

SELECT 字段1 [ 别名1 ] , 字段2 [ 别名2 ] ... FROM 表名;

3). 去除重复记录

SELECT DISTINCT 字段列表 FROM 表名;

条件查询

1). 语法

SELECT 字段列表 FROM 表1 名 WHERE 条件列表 ;

2). 条件

常用的比较运算符如下



常用的逻辑运算符如下



聚合函数

1). 介绍

将一列数据作为一个整体，进行纵向计算 。

2). 常见的聚合函数



3). 语法

SELECT 聚合函1 数(字段列表) FROM 表名 ;

注意 : NULL值是不参与所有聚合函数运算的。

分组查询

1). 语法

SELECT 字段列表 FROM 表名 [ WHERE 条件 ] GROUP BY 分组字段名 [ HAVING 分组后过滤条件 ];

2). where与having区别

执行时机不同：where是分组之前进行过滤，不满足where条件，不参与分组；而having是分组之后对结果进行过滤。

判断条件不同：where不能对聚合函数进行判断，而having可以。

注意事项:

分组之后，查询的字段一般为聚合函数和分组字段，查询其他字段无任何意义。

执行顺序: where > 聚合函数 > having 。

支持多字段分组, 具体语法为 : group by columnA, columnB

排序查询

排序在日常开发中是非常常见的一个操作，有升序排序，也有降序排序。

1). 语法

SELECT 字段列表 FROM 表名 ORDER BY 字段1 排序方式1 , 字段2 排序方式2 ;

2). 排序方式

ASC : 升序(默认值)

DESC: 降序

注意事项：

如果是升序, 可以不指定排序方式ASC ;

如果是多字段排序，当第一个字段值相同时，才会根据第二个字段进行排序 ;

分页查询

1). 语法

SELECT 字段列表 FROM 表名 LIMIT 起始索引, 查询记录数 ;

注意事项:

起始索引从0开始，起始索引 = （查询页码 - 1）\* 每页显示记录数。

分页查询是数据库的方言，不同的数据库有不同的实现，MySQL中是LIMIT。

如果查询的是第一页数据，起始索引可以省略，直接简写为 limit 10。

执行顺序



DCL

DCL英文全称是Data Control Language(数据控制语言)，用来管理数据库用户、控制数据库的访问权限。

管理用户

1). 查询用户

select \* from mysql.user;

其中 Host代表当前用户访问的主机, 如果为localhost, 仅代表只能够在当前本机访问，是不可以远程访问的。 User代表的是访问该数据库的用户名。在MySQL中需要通过Host和User来唯一标识一个用户。

2). 创建用户

CREATE USER '用户名'@'主机名' IDENTIFIED BY '密码';

3). 修改用户密码

ALTER USER '用户名'@'主机名' IDENTIFIED WITH mysql\_native\_password BY '新密码' ;

4). 删除用户

DROP USER '用户名'@'主机名' ;

注意事项:

在MySQL中需要通过用户名@主机名的方式，来唯一标识一个用户。

主机名可以使用 % 通配。

这类SQL开发人员操作的比较少，主要是DBA（ Database Administrator 数据库

管理员）使用。

权限控制



1). 查询权限

SHOW GRANTS FOR '用户名'@'主机名' ;

2). 授予权限

GRANT 权限列表 ON 数据库名.表名 TO '用户名'@'主机名';

3). 撤销权限

REVOKE 权限列表 ON 数据库名.表名 FROM '用户名'@'主机名';

注意事项：

多个权限之间，使用逗号分隔

授权时， 数据库名和表名可以使用 \* 进行通配，代表所有。