**16-数据库**

1. **mysql使用**

1.1 **安装管理操作**

安装

|  |
| --- |
| Python 安装 apt-get install mysql-server   设置密码 mysqladmin -u root password "new\_password"; 登录 mysql -u root -p  上述bug可以这样登录  sudo mysql -u root或 sudo mysql 然后设置密码 ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY '密码';  退出 quit |

修改sql密码，给远程权限

|  |
| --- |
| Python 账户密码 /etc/mysql/[debian](https://so.csdn.net/so/search?q=debian&spm=1001.2101.3001.7020).cnf文件，查看debian-sys-maint帐号密码   为root更改密码，localhost，只允许本地 ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY 'password'; 修改远程更改密码，允许外部tcp登录，默认没有远程账户进行下面操作 ALTER USER 'root'@'%' IDENTIFIED WITH mysql\_native\_password BY 'password';  查看远程账户 SELECT user, host FROM mysql.user WHERE user = 'root';  创建远程账户 CREATE USER 'root'@'%' IDENTIFIED BY 'password';  给权限给远程，访问特定数据库 GRANT ALL PRIVILEGES ON yourdatabase.\* TO 'yourusername'@'%'; 也可以全部给 GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'%' WITH GRANT OPTION; 用于重新载入授权表（用户账户和权限信息） FLUSH PRIVILEGES; # |

解决远程无法登录

|  |
| --- |
| Bash vi /etc/mysql/mysql.conf.d/mysqld.cnf 修改地址绑定将 bind-address = 127.0.0.1 改为 bind-address = 0.0.0.0 service mysql restart |

1.2 **命令语句操作**

1.2.1 **基本命令**

SQL语句都是以；结尾

请注意这些命令都是大写，MySQL不区分大小写

|  |
| --- |
| Python 显示数据库 show databases; 显示表 show tables;  创建数据库 create database mytest; 删除数据库 drop database mytest; 选择数据库  use mytest;    创建表  mysql> **CREATE** **TABLE** runoob\_tbl(  -> runoob\_id **INT** **NOT** **NULL** **AUTO\_INCREMENT**,  -> runoob\_title **VARCHAR**(100) **NOT** **NULL**,  -> runoob\_author **VARCHAR**(40) **NOT** **NULL**,  -> submission\_date **DATE**,  -> **PRIMARY** **KEY** ( runoob\_id )  -> )ENGINE=InnoDB **DEFAULT** CHARSET=utf8; Query OK, 0 **ROWS** affected (0.16 sec) 检查删除表格 drop table if exists my\_tbl; 删除表格 drop table my\_tbl;  插入数据 mysql> insert into my\_tbl  -> (title, author, date)  -> values  -> ("学习 MySQL", "旭宝of", now()); Query OK, 1 row affected, 1 warning (0.02 sec) 或 mysql> insert into my\_tbl (title, author, date) values ("学习 C++", NULL, now()); Query OK, 1 row affected, 1 warning (0.01 sec)  查询数据 查看所有数据 mysql> SELECT \* FROM my\_tbl; 特定列所有行 SELECT title, date FROM my\_tbl; 按条件查询行 SELECT \* FROM users WHERE id = 1; 按照某列升序（字符串也可以排序） SELECT \* FROM my\_tbl ORDER BY title; 按照某列降序 SELECT \* FROM my\_tbl ORDER BY title DESC; 限制多少行返回 SELECT \* FROM my\_tbl LIMIT 10; 条件组合（可以使用各种条件运算符（如 =, <, >, <=, >=, !=），逻辑运算符（如 AND, OR, NOT），以及通配符（如 %）等） mysql> SELECT \* FROM my\_tbl WHERE id = 2 or id <= 1; 以2为开头的通配符LIKE '2%' SELECT \* FROM my\_tbl WHERE date LIKE '2%'; 条件中筛选 SELECT \* FROM my\_tbl WHERE date IN ('2024-03-12', '2024-03-11')  更新数据 mysql> UPDATE my\_tbl SET title = '学习Python', date = '2024-03-12' where id = 2; Query OK, 1 row affected (0.01 sec) Rows matched: 1 Changed: 1 Warnings: 0  删除数据 DELETE FROM my\_tbl WHERE date='2024-03-12'; Query OK, 1 row affected (0.01 sec)   选择数据库带表格操作 select \* from mytest.my\_tbl |

表格实验数据

|  |
| --- |
| Python mysql> SELECT \* FROM my\_tbl; +----+--------------+----------+------------+ | id | title | author | date | +----+--------------+----------+------------+ | 1 | 学习 MySQL | 旭宝of | 2024-03-11 | | 2 | 学习 C++ | NULL | 2024-03-11 | | 3 | 学习 C++ | NULL | 2024-03-11 | +----+--------------+----------+------------+ |

1.2.2 **命令行解释**

**AUTO\_INCREMENT** 定义列为自增的属性，一般用于主键，数值会自动加 1。

**PRIMARY KEY** 关键字用于定义列为主键。 您可以使用多列来定义主键，列间以逗号 **,** 分隔。

**ENGINE** 设置存储引擎，**CHARSET** 设置编码。

**LIKE** 子句是在 MySQL 中用于在 WHERE 子句中进行模糊匹配的关键字。它通常与通配符一起使用，用于搜索符合某种模式的字符串。

**LIKE** 子句中使用百分号 **%**字符来表示任意字符，类似于UNIX或正则表达式中的星号 **\***。

如果没有使用百分号 **%**, LIKE 子句与等号 **=** 的效果是一样的。

**UNION** 操作符用于连接两个以上的 SELECT 语句的结果组合到一个结果集合，并去除重复的行。

1.2.3 **mysqladmin工具**

|  |
| --- |
| Python 用mysqladmin工具删除表格 mysqladmin -u your\_username -p drop your\_table |

1.3 **数据库位置**

/var/lib/mysql/

2. **可视化工具安装**

2.1 **命令行安装**

可能不成功，可以使用第二种方式

sudo apt-get install mysql-workbench

2.2 **安装包安装**

下载安装包

https://downloads.mysql.com/archives/workbench/

安装

dpkg -i mysql-workbench-community\_8.0.29-1ubuntu20.04\_amd64.deb

2.3 **使用方法**

PK(Primary Key) 主键，一个数据表中只能有一个字段是主键，我们一般通过主键来索引某一条数据

NN(Not Null) 非空，该键的值不能是空的

UQ(Unique) 值唯一，我们已用户名来举例，A起名叫张三后，其余所有人就不能叫张三了

B(Binary) 二进制数据

UN(UNsigned) 无符号(非负数)

ZF(zero fill) 填充0 例如字段内容是1，字段类型是int(4),则内容显示为0001

AI(Auto Increment) 值自动增长，把这个勾上写入数据的时候不用刻意去传id，每写一条数据id自动增加(如果你想给一个指定的ID也是可以的)

G(Generated) 勾上之后表示该列是基于其他列计算产生的列

3. **SQL数据类型**

数值类型



**点击图片可查看完整电子表格**

日期类型



**点击图片可查看完整电子表格**

字符串类型



**点击图片可查看完整电子表格**

枚举



**点击图片可查看完整电子表格**

空间数据类型

GEOMETRY,

POINT,

LINESTRING,

POLYGON,

MULTIPOINT,

MULTILINESTRING,

MULTIPOLYGON,

GEOMETRYCOLLECTION: 用于存储空间数据（地理信息、几何图形等）。

正则表达式匹配的字符类

* .：匹配任意单个字符。
* ^：匹配字符串的开始。
* $：匹配字符串的结束。
* \*：匹配零个或多个前面的元素。
* +：匹配一个或多个前面的元素。
* ?：匹配零个或一个前面的元素。
* [abc]：匹配字符集中的任意一个字符。
* [^abc]：匹配除了字符集中的任意一个字符以外的字符。
* [a-z]：匹配范围内的任意一个小写字母。
* \d：匹配一个数字字符。
* \w：匹配一个字母数字字符（包括下划线）。
* \s：匹配一个空白字符。

|  |
| --- |
| Python mysql> SELECT name FROM person\_tbl WHERE name REGEXP '^st'; |

4. **参考网址**

MySQL工具在社区可以下载

1. [MySQL 查询数据 | 菜鸟教程](https://www.runoob.com/mysql/mysql-select-query.html)
2. [Python 连接 MySQL 数据库实例\_w3cschool](https://www.w3cschool.cn/mysql/mysql-de5j3g5o.html)

[mydb.py](https://li.feishu.cn/wiki/SBkOw4l7AitTfok6D3TcBVGtnoh?fromScene=spaceOverview)

1. 登录失败解决

5. **python**

python3 可以使用mysql-connector-python模块连接、操作数据库

|  |
| --- |
| Python pip install mysql-connector-python |

使用远程连接，mysql需要单独创建一个用户，并对用户赋予对应database的权限

|  |
| --- |
| SQL create user user1 identified by "密码"; grant all privileges on mydb.\* to user1@'%'; show grants for user1; |

python连接mysql并查询方法

|  |
| --- |
| Python import mysql.connector def query\_from\_mysqldb(table, tags:list, fields:list):  db = mysql.connector.connect(host="127.0.0.1", user="user1", passwd="user1", db="mydb")  key\_str = ",".join(tags + fields)  try:  cursor = db.cursor()  sql = "SELECT {} FROM {}".format(key\_str, table)  cursor.execute(sql)  results = cursor.fetchall()  return results  finally:  db.close() |