

MySQL数据库基本操作

登录

退出

得到版本

登录同时打开指定数据库

命令行结束符

查看命令行手册

常用SQL语句

SQL语句语法规范

数据库相关操作

创建数据库

查看已有数据库

查看指定数据库的详细信息

修改指定数据库的编码方式

打开指定数据库

得到当前打开的数据库

删除指定的数据库

数据表相关操作

数据表

创建数据表

查看当前数据库下已有数据表

查看指定数据表的详细信息

查看表结构

删除指定的数据表

插入一条记录

查询表中所有记录

表结构相关操作

MySQL中的数据类型

数值型

字符串类型

日期时间类型

MYSQL存储引擎

MyISAM存储引擎

Inodb存储引擎

MySQL数据库基本操作

登录

```
mysql -uroot -p
```

退出

```
exit
```

quit

\q

得到版本

mysql -V 或者 mysql --version

登录同时打开指定数据库

mysql -uroot -p -D db_name

命令行结束符

默认使用 ; 或者 \g 来结束

查看命令行手册

help create database

可以通过 help

常用SQL语句

- SELECT USER() : 得到登录的用户
- SELECT VERSION() : 得到Mysql的版本信息
- SELECT NOW() : 得到当前的日期时间
- SELECT DATABASE() : 得到当前打开的数据库
- SHOW WARNINGS; : 查看上一步操作产生的警告信息

SQL语句语法规范

- 常用的MySQL的关键字我们需要大写，库名、表名、字段名称等使用小写
- SQL语句支持折行操作，拆分的时候不能把完整单词拆开
- 数据库名称、表名称、字段名称不要使用MySQL的保留字，如果必须要使用，需要用反引号 `` 将其扩起来
- mysql中的注释
 - #注释内容
 - --注释内容

数据库相关操作

创建数据库

CREATE DATABASE db_name 或者 CREATE SCHEMA db_name

检测数据库名称是否存在，不存在则创建 CREATE DATABASE [IF NOT EXISTS] db_name;

在创建数据库的同时制定编码方式 `CREATE DATABASE [IF NOT EXISTS] db_name [DEFAULT] CHARACTER SET [=]charset;`

注意：数据库名称最好取的有意义，名称不要包括特殊字符或者是MySQL关键字

查看已有数据库

```
show databases;
```

查看指定数据库的详细信息

```
SHOW CREATE DATABASE db_name;
```

修改指定数据库的编码方式

```
ALTER DATABASE db_name[DEFAULT]CHARACTER SET [=]charset;
```

打开指定数据库

```
USE db_name;
```

得到当前打开的数据库

```
SELECT DATABASE()|SCHEMA();
```

删除指定的数据库

```
DROP DATABASE db_name;
```

如果数据库存在则删除 `DROP DATABASE [IF EXISTS] db_name;`

数据表相关操作

数据表

- 数据表是数据库最重要的组成部分之一，数据是保存在数据表中
- 数据表由行(`row`)和列(`col`)来组成
- 每个数据表中至少有一列，行可以有零行一行或者多行
- 表名要求唯一，不要包含特殊字符，取名要有意义
- 完整性约束条件
 - `unsigned` :无符号，没有负数，从0开始
 - `zerofill` :零填充，当显示长度不够的时候可以使用前补0的效果填充至指定长度
 - `NOT NULL` :非空约束，也就是插入值的时候这个字段必须要给值
 - `DEFAULT` :默认值，如果插入记录的时候没有给字段赋值，则使用默认值
 - `PRIMARY KEY` :主键，标识记录的唯一性，值不能重复，一个表只能有一个主键，自动禁止为空
 - 复合主键

- ```

1 CREATE TABLE table_name(
2 id INT UNSIGNED,
3 courseId INT UNSIGNED,
4 username VARCHAR(20),
5 email VARCHAR(50),
6 PRIMARY KEY(id ,courseId)
7);
8
9 INSERT table_name(id,courseId,username,email)
10 VALUES(1,1,'king','939744682@qq.com');
```

- 复合主键赋值时必须同时相同才代表数据重复，通过两个复合主键都可以定义到同一条元素
  - UNIQUE KEY :唯一性，一个表中可以有多个字段是唯一索引，同样的值不能重复，null除外
  - AUTO\_INCREMENT :自动增长，只能用于数值列，而且配合索引所有，默认起始值1开始，每次增长1，已有编号的最大值加1
  - FOREIGN KEY :外键约束

## 创建数据表

```

1 CREATE TABLE[IF NOT EXISTS] tbl_name(
2 字段名称 字段类型[完整约束条件]
3 字段名称 字段类型[完整约束条件]
4
5)ENGINE=存储引擎 CHARSET=编码方式;
```

## 查看当前数据库下已有数据表

```
SHOW TABLES ;
```

```
SHOW [FULL] TABLES[{FROM|IN}db_name][LIKE 'pattern'|WHERE expr];
```

## 查看指定数据表的详细信息

```
SHOW CREATE TABLE tbl_name;
```

## 查看表结构

```
DESC tbl_name;
```

```
DESCRIBE tbl_name;
```

```
SHOW COLUMNS FROM tbl_name;
```

## 删除指定的数据表

```
DROP TABLE[IF EXISTS] tbl_name;
```

## 插入一条记录

```
INSERT [INTO] tbl_name(id,username,...) VALUES(1,"KING",.....);
```

## 查询表中所有记录

```
select * from tbl_name;
```

## 表结构相关操作

- 添加字段

- - 1 `alter table tbl_name`
  - 2 `add 字段名称 字段属性[完整性约束条件][FIRST|AFTER 字段名称];`

- 删除字段

- - 1 `alter table tbl_name`
  - 2 `drop 字段名称;`

- 添加默认值

- - 1 `alter table tbl_name`
  - 2 `alter 字段名称 set default 默认值;`

- 删除默认值

- - 1 `alter table tbl_name`
  - 2 `alter 字段名称 drop default ;`

- 修改字段类型、字段属性

- - 1 `alter table tbl_name`
  - 2 `modify 字段名称 字段类型[字段属性][first|after 字段名称];`

- 修改字段名称、字段类型、字段属性

- - 1 `alter table tbl_name`
  - 2 `change 原字段名称 新字段名称 字段类型 字段属性[first|after 字段名称];`

- 添加主键

- - 1 `alter table tbl_name`
  - 2 `add primary key(字段名称);`

- 删除主键

- ```
1 alter table tbl_name
2 drop primary key;
```

- 添加唯一

- ```
1 alter table tbl_name
2 add unique key|index index_name (字段名称);
```

- 删除唯一

- ```
1 alter table tbl_name
2 drop index_name;
```

- 修改数据表名称

- ```
1 alter table tbl_name
2 rename [to|as] new_tbl_name;
```

- ```
1 rename table tbl_name to new_tbl_name;
```

MySQL中的数据类型

[Mysql的数据类型](#)

数值型

- 整数型
- 浮点型
- 定点型

字符串类型

- CHAR(M):M个字节
- VARCHAR(M):1+1个字节,其中1<=M且0<=M<=65535
- CHAR效率高于VARCHAR,CHAR相当于拿空间换时间, VARCHAR拿时间换空间
- CHAR默认进行存储数据的时候, 后面会用空格填充到指定长度, 而在检索的时候会去掉后面空格
- VARCHAR在保存的时候不进行填充, 尾部的空格会留下
- TEXT列不能有默认值,检索的时候不存在大小写转换

日期时间类型

- Time: 3个字节

- DATE:3个字节
- DATETIME:8个字节
- TIMESTAMP:4个字节
- YEAR:1个字节

MYSQL存储引擎

MyISAM存储引擎

- 默认MyISAM的表会在磁盘中产生3个文件
 - `.frm`：表结构文件
 - `.MYD`：数据文件
 - `.MYI`：索引文件
- 可以在创建表的时候指定数据文件和索引文件的存储位置，只有MyISAM表支持

- 1 | `DATA DIRECTORY [=]` 数据保存的绝对路径

- 1 | `INDEX DIRECTORY [=]` 索引文件保存的绝对路径

- MyISAM单表最大支持的数据量 2^{64} 次方条记录
- 每个表最多可以建立64个索引
- 如果是复合索引，每个复合索引最多包含16个，索引值最大长度是1000B
- MyISAM引擎的存储格式
 - 定长(FIXED 静态)
 - 动态(DYNAMIC)
 - 压缩(COMPRESSED)

Inodb存储引擎