一　数据库

DB,DataBase：依照某种数据模型进行组织并存放到存储器的数据集合

数据库：

DBMS

数据库管理系统：用来操作和管理数据的大型服务软件

DBS

数据库系统：DB+DBMS,指带有数据库并整合了数据库管理软件的计算机

1，软件

－－关系型

闭源 Oracle DB2 sql server(windows)

开源 mysql mariadb

跨平台 oracle DB2 mysql mariadb(mysql姐妹篇)

Mysql特点：

适合中小型企业的关系型数据库服务软件

支持多种操作系统(windows unix linux)

支持多种开发语言

可移植性好

端口号：3306

进程号：mysqld

主配置文件：／etc/my.cnf

数据库文件：/var/lib/mysql/

进程所有者，所属组：mysql

　utf8 一个汉字3个字符,以前的mysql

现在的版本mysql ==utf8　==> char(3) ==> 存储3个汉字

－－非关系型

Redis MongoDB memcache

２，数据库

－－数据库的命令规则

可以使用数字，字母，下环线，但不能纯数字

区分大小写，具有唯一性

不可以使用指令关键字，特殊字符

－－mysql数据库类型

1. 数值类型



##整数　1个字节＝8 byte

Tinyint (1个字节)

Smallint(2个字节)

Int(4个字节)

Bigint(8个字节)

##浮点型

单精度fload（4个字节）双精度double(8个字节)

Fload(总宽度，小数位数)，double(总宽度，小数位数)　#不写默认的

Fload(5,2) #5位数，保留小数两位数

1. 字符

##char

char(字符数)

最大长度255

不够指定字符数时在右变用空格补齐

字符数断超出时，无法写入数据

存储数据速度快

##varchar

varchar(字符数)

最大长度65532

按数据实际大小分配存储空间

字符数段超出时，无法写入数据

存储数据速度较char慢

##text/blob

　　字符数大于65535存储时使用

1. 枚举

##从给定值集合中选择单个值，ENUM

定义格式enum(值１，值２，值n)

##从给定值集合中选择一个或多个值，SET

定义格式：set(值１，值２，值n)

1. 日期

##datetime

　　默认为null

占用8个字节

范围：1000-01-01　00：00：00：.000000

　　　~9999-12-31　23：59：59.999999

##timestamp

　　默认当前系统的时间

占用4个字节

范围：1970-01-01 00:00:00.000000

~2038-01-19 03:14:07.999999

##date

占用4个字节

范围：0001-01-01 ~ 9999-12-31

##year

占用1个字节

范围：1901－2155

##time

占用3个字节

格式：HH:MM:SS

1. 时间函数



３，搭建mysql数据库

－－安装，启服务

１）删除mariadb数据库

[root@mysql-50 jluo]# rpm -e --nodeps mariadb-server 　#--nodeps忽略依赖

[root@mysql-50 jluo]# rpm -e --nodeps mariadb

[root@mysql-50 jluo]# rm -rf /var/lib/mysql/\*

[root@mysql-50 jluo]# rm -rf /etc/my.conf

２）虚拟机192.168.4.1解压mysql-5.7.17.tar 整合包

[root@mysql-50 jluo]# tar -xvf mysql-5.7.17.tar //解压mysql整合包

[root@mysql-50 jluo]# yum -y install perl-JSON　　　　#提前安装依赖包

[root@mysql-50 jluo]# rpm -Uvh \*.rpm　　　　　　　　　#Ｕ更新安装

1. 启动MySQL数据库服务并设置开机自启

[root@mysql-50 jluo]# systemctl start mysqld

[root@mysql-50 jluo]# systemctl enable mysqld

[root@mysql-50 jluo]# systemctl status mysqld

● mysqld.service - MySQL Server

Loaded: loaded (/usr/lib/systemd/system/mysqld.service; enabled; vendor preset: disabled)

Active: active (running) since 五 2018-11-16 10:23:36 CST; 40s ago

Docs: man:mysqld(8)

http://dev.mysql.com/doc/refman/en/using-systemd.html

Main PID: 2548 (mysqld)

CGroup: /system.slice/mysqld.service

└─2548 /usr/sbin/mysqld --daemonize --pid-file=/var/run/mysql...

11月 16 10:23:08 mysql-50 systemd[1]: Starting MySQL Server...

11月 16 10:23:36 mysql-50 systemd[1]: Started MySQL Server.

[root@mysql-50 jluo]# ls /var/lib/mysql　　　#第一次初始化的mysql文件

auto.cnf ib\_buffer\_pool mysql public\_key.pem

ca-key.pem ibdata1 mysql.sock server-cert.pem

ca.pem ib\_logfile0 mysql.sock.lock server-key.pem

client-cert.pem ib\_logfile1 performance\_schema sys

client-key.pem ibtmp1 private\_key.pem

－－连接MySQL服务器，修改密码

命令：

mysql -h 数据库服务器ip／主机名　-u 用户名　-p密码　［库名］

１）查看随机生成的root管理密码

[root@mysql-50 jluo]# grep 'temporary password' /var/log/mysqld.log

2018-11-16T02:23:18.210657Z 1 [Note] A temporary password is generated for root@localhost: R4MzjtCyY!%.

初始密码：R4MzjtCyY!%.

２）使用客户端命令mysql连接到MySQL服务器，修改密码

[root@mysql-50 jluo]# mysql -uroot -p'R4MzjtCyY!%.'

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

ERROR 1819 (HY000): Your password does not satisfy the current policy requirements

mysql> alter user root@localhost identified by "LOL.jluo14.com" ; #修改密码

Query OK, 0 rows affected (0.00 sec)

提示：必须要修改密码之后才能操作数据库．

1. 修改密码策略

mysql> show variables like "%password%";

．．．．

| validate\_password\_length | 8 |

| validate\_password\_mixed\_case\_count | 1 |

| validate\_password\_number\_count | 1 |

| validate\_password\_policy | MEDIUM |

| validate\_password\_special\_char\_count | 1 |

+---------------------------------------+--------+

14 rows in set (0.01 sec)

##临时设置密码规则，重启后无效

mysql>set global validate\_password\_policy=0;　　#验证长度

mysql>set global validate\_password\_length=6;　　#密码长度

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

Query OK, 0 rows affected (0.00 sec)

##永久有效

[root@mysql-50 jluo]# vim /etc/my.cnf

[mysqld]

validate\_password\_length=6

validate\_password\_policy=0

[root@mysql-50 jluo]# systemctl restart mysqld

４，数据库sql命令

－－常见的sql操作指令

DDL：数据定义语言（create ,alter,drop）

DML：数据操作语言（insert,update,delete,select）

DCL：数据控制语言（grant,revoke）

DTL：数据事物语言（commit,rollback,savepoint）

－－修改MySQL服务的默认字符集

1. 建表时指定字符集

mysql> create table teacher(name char(10),age int ,password char(20))default charset=utf8;　　#建表时指定字符集utf8

1. 修改配置文件

#vim /etc/my.cnf

character\_set\_server=utf8

－－约束条件

１）设置字段默认值

mysql> create table t4(

-> name char(10) not null,

-> age tinyint not null default 25,

-> likes set("eat","sleep","game","zhang") default "eat,sleep");

1. 设置表主键,外键，Primary key

一个表中可以有多个索引字段

##键值类型

Index：普通索引

Unique：唯一索引

Fulltext：全文索引

Primary key：主键

Foreign key：外键

##操作index索引

Create index　索引名　on　表名(字段名);

Drop index 索引名 on　表名  
例子：

mysql> show index from stutab\G;　　　#查看表的索引信息,每一行一个

mysql> create index age on stutab(age);　　#方式一

mysql> create table t3(　　　　　　　　　　　#方式二

-> name char(10)not null,

-> age tinyint not null default 25,

-> likes set("eat","sleep","game") default "eat,sleep",

-> index(name),index(age));

mysql> alter table stutab drop index age;　　　#删除index（age）索引

##primary key

注意：

一个表中只能有一个primary key 字段

对应的字段不允许有重复，且不允许为空

如果有多个字段都作为primary key,称为复合主键，必须一起创建

主键字段的key标识是pri

通常与auto\_increment连用　　　#设置主键自曾

经常把表中能够唯一标识记录的字段设置为主键字段

建表的时候指定主键字段

　　Primary key(字段名)

Alter table 表名　add primary key(字段名);　#在已有表中设置主键

Alter table 表名 drop primary key; #移除主键

例子：

mysql> create table t4 (　　　　　　　#方式一

-> id char(12) primary key ,

-> name char(10) not null,

-> class char(10) not null

-> );

mysql> create table t5( id char(12), class char(10) not null, name char(10) not null ,primary key(id));　　　　　　　　#方式二

mysql> alter table stutab add primary key(stu\_num);　#给已存在的表（没有主键）新建主键

mysql> alter table stutab drop primary key; #删除表中主键

提示：移除主键前，如果有自曾属性，必须先去掉

创建复合主键

mysql> create table t6( clientip char(15), serport smallint unsigned, status enum("allow","deny"), primary key(clientip,serport) );

##foreign key

让当前的字段的值在另一个表中字段值的范围内选择

使用外键的条件：

表的存储引擎必须是innodb

字段类型要一致

被参照字段必须要是索引类型的一种（primary key）

基本用法：

Foreign key(表a的字段名) references 表b(字段名)　［on update|delete cascade］;

同步更新｜同步删除　默认没有这两个功能，需要自己启用

Alter table 表名 drop foreign key 约束名; #删除外键

mysql> show create table gz;　　　　#查看当时建表的命令

| gz | CREATE TABLE `gz` (

`gz\_id` int(11) DEFAULT NULL,

`pay` float(7,2) DEFAULT NULL,

KEY `gz\_id` (`gz\_id`),

CONSTRAINT `gz\_ibfk\_1` FOREIGN KEY (`gz\_id`) REFERENCES `yg` (`id`) ON DELETE CASCADE ON UPDATE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8 |

例子：

mysql> create table gz( gz\_id int , pay float(7,2), foreign key(gz\_id) references yg(id) on update cascade on delete cascade )engine=innodb;　　#添加外键

mysql> alter table gz add primary key(gz\_id);　　　　　#添加主键

mysql> alter table gz drop foreign key gz\_ibfk\_1;　　　　#删除外键

－－修改alter

1. 修改表结构

基本用法：

Alter table 表名 执行动作 [after 字段名 | first]

Add 添加字段

Modify 修改字段类型

Change 修改字段名

Drop　删除字段

Rename 修改表名

After 字段名１：添加的新字段在字段１的后面

First：添加的新字段在表的最前面

提示：不加字段顺序，默认在后面字段追加．

例子：

##添加字段

mysql> alter table t4 add email varchar(30) default "jluo@jluo.com",

-> add tel char(11),add stu\_num char(9) first;

mysql> alter table t4 add class char(7) not null default "nsd1808" after name;

##修改字段

mysql> alter table t4 stu\_num char(10) not null;

提示：修改字段时，若与已有数据类型或属性有冲突则修改失败

修改字段时，不修改的数据要原样写入

##删除字段

mysql> alter table stutab drop email;

##修改表名

mysql> alter table t4 rename to stutab;

##修改字段名

mysql> alter tale t4 change name t4\_name char(10);

－－基本操作

1. 数据库操作

格式：

show databases;　　#查看数据库信息

use 库名;　　　　　#切换库

select database();　　#显示当前所在的库

show tables;　　　　　#显示已有的表

create database 新库名;　#创建新库

drop database 库名;　　#删除库

desc(desctibe) 库名.表名 #查看表结构

例子：

使用 CHANGE 子句, 语法有很大的不同。 在 CHANGE 关键字之后，紧跟着的是你要修改的字段名，然后指定新字段名及类型。尝试如下实例：

mysql> ALTER TABLE testalter\_tbl CHANGE i j BIGINT;

mysql> ALTER TABLE testalter\_tbl CHANGE j j INT;

mysql> create database studb;

mysql> use studb;

mysql> create table student(name char(10),age int);

mysql> desc student　　　　　#查看表结构

变化

[root@mysql-50 ~]# ls /var/lib/mysql/studb/

db.opt student.frm student.ibd

db.opt：数据库studb配置文件

student.frm student.ibd：表student信息文件

1. Table

格式：

mysql> create table teacher(name char(10),age int ,password char(20))default charset=utf8;　　#建表时指定字符集utf-8

1. Select

格式：

1. Insert

格式：

1. Update

格式：

1. Delete

格式：

－－授权