6.14 DBA1

1. 搭建数据库服务器

服务器 cpu 内存 存储

装包： 源码、RPM、包的来源、软件的版本

主流的数据库软件：

关系性数据库管理系统：(RDBMS)

Oracle DBA Mysql mariadb

非关系数据库管理系统：（Nosql）

Redis Monyodh

软件的版本：

商业、开源、跨平台

1.安装软件包

系统自带mariadb包，再次安装，会冲突，所以选择升级软件包

[root@host50 ~]# rpm -qa | grep -i mysql //查看系统的软件包

root@host51 ~]# tar -xf mysql-5.7.17.tar

[root@host50 ~]# rpm -Uvh mysql-community-\*.rpm

-Uvh 升级软件包，不会发生冲突

安装软件包，会报错，用yum安装

[root@host50 ~]# yum -y install perl-JSON

报错信息为

警告：mysql-community-client-5.7.17-1.el7.x86\_64.rpm: 头V3 DSA/SHA1 Signature, 密钥 ID 5072e1f5: NOKEY

错误：依赖检测失败：

perl(JSON) 被 mysql-community-test-5.7.17-1.el7.x86\_64 需要

安装完成后,在此运行

[root@host50 ~]# rpm -Uvh mysql-community-\*.rpm

查看软件包

[root@host50 ~]# rpm -qa | grep -i mysql

mysql-community-libs-5.7.17-1.el7.x86\_64

mysql-community-embedded-compat-5.7.17-1.el7.x86\_64

mysql-community-embedded-devel-5.7.17-1.el7.x86\_64

mysql-community-server-5.7.17-1.el7.x86\_64

mysql-community-test-5.7.17-1.el7.x86\_64

php-mysql-5.4.16-42.el7.x86\_64

mysql-community-common-5.7.17-1.el7.x86\_64

mysql-community-devel-5.7.17-1.el7.x86\_64

mysql-community-libs-compat-5.7.17-1.el7.x86\_64

mysql-community-embedded-5.7.17-1.el7.x86\_64

mysql-community-client-5.7.17-1.el7.x86\_64

mysql-community-minimal-debuginfo-5.7.17-1.el7.x86\_64

修改配置文件 /etc/my.cnf

默认的数据库目录 /var/lib/mysql

默认的端口是3306

查看进程 [root@host50 mysql]# ps -C mysqld

启动服务

[root@host50 mysql]# systemctl start mysqld

[root@host50 mysql]# systemctl status mysqld

[root@host50 mysql]# systemctl enable mysqld

查看服务起没起的三种方法：

端口（mysqld） 进程（ps -C mysqld）状态（ status）

查看他所有的进程

[root@host50 mysql]# ps aux | grep mysqld

默认的所属组和所有人是mysql

[root@host51 ~]# ls -ld /var/lib/mysql

1. 数据库服务器的基本使用

把数据存储到服务器上的过程

1. 连接数据库 （命令连接 图形工具）
2. 选择库 （文件夹）
3. 建表 （文件）
4. 插入记录 （行）
5. 断开连接

Which mysql 查看有没有安装client

首次登陆密码会随机生成，存储在错误日志文件里

[root@host50 mysql]# cd /var/log

[root@host50 log]# grep -i password /var/log/mysqld.log

默认随机密码是不能在数据库中操作，所以要更改数据库密码，默认密码长度是8位

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

Validate\_password\_policy=0 全局设置

mysql> set global validate\_password\_length=6; // 修改密码长度 默认值是 8 个字

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

设置永久的要更改配置文件

[root@host50 log]# vim /etc/my.cnf

[mysqld]

validate\_password\_policy=0

validate\_password\_length=6

修该配置文件后，先停掉服务，在重起

数据库的注意事项：

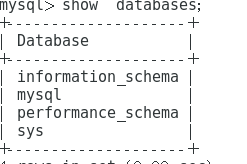
命令不区分大小写

每条命令以分号结尾

不支持tab健补齐

\c 可废弃当前编辑写错的操作指令

数据库默认有四个库：不可删除



数据库的命名规则：

不可使用纯数字、特殊字符

区分大小写，具有唯一性（不能创建相同的）

库命令：

Show databases; 显示已有的库

Use 库名；

Select database（）； 显示当前所在的库

Create database 库名； 创建新库

Show tables； 显示已有的库

Drop database 库名； 删除库

在系统命令在创建一个库，数据库中也会多出一个库，但他只有读的权限，不能创建表,他的所有人和所属组是root

表命令：

Desc 表名； //察看表结构

Select \* from 表名； //查看表记录

Drop table 表名； 删除表

创建表，语法格式

Create table 库名.表名（

字段名1 字段类型（宽度） 约束条件，

字段名2 字段类型（宽度） 约束条件

）；

mysql> create table t1 (

-> name char(15),

-> class char(7)

-> );

插入

Insert into 表名 values( 值列表 ); 插入表记录

mysql> insert into t1 values("toutou","NSD1803"),("dalao","NSD1803");

修改表

Update 表名 set 字段 = 值 ; // 修改表记录

mysql> update t1 set class="NSD1809";

删除表

Delete from 表名 ; // 删除表记录

mysql> delete from t1;

例：

mysql> create table 学生信息表(

-> 姓名 char(15),

-> 班级 char(7)

-> ) DEFAULT CHARSET=utf8;

mysql> insert into 学生信息表 values("toutou","NSD1803"); 插入表

查看表信息

mysql> show create table 学生信息表;

Mysql默认的字符是latin1，不支持中文，改成utf8，支持中文

Mysql 数据类型

常见信息种类、数值类型、字符类型、日期时间类型、枚举类型、约束条件

默认数值类型使用的范围是有符号的，使用无字符要用：

无符号：unsigned



整数型

使用UNSIGNED修饰时，对应的字段只保存整数

数值不够指定宽度时,在左边填空格补位

宽度仅是显示宽度,存数值的大小由类型决定

使用关键字 ZEROFILL 时,填 0 代替空格补位

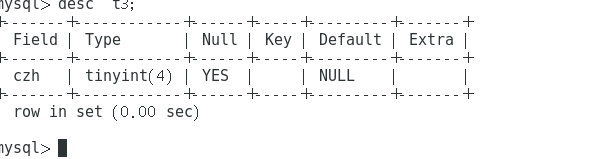
数值超出范围时,报错

mysql> create table t3(

-> czh tinyint

-> );

mysql> insert init t3 values(88);



浮点型

定义格式: float( 总宽度 , 小数位数 )

当字段值与类型不匹配时,字段值作为 0 处理

数值超出范围时,仅保存最大 / 最小值

整数.小数 存储最大值为999.00（1000）

类型

M 总位数 n小数

Flocat （m，n）

Double（m，n）

mysql> create table t4(

-> pay double(5,2)

mysql> insert into t4 values(44.22)

字符类型 最大是65532

定长 char（字符数） 最大长度为255字符

指定字符不能超过，超过不能写入数据

不指定字符时，默认是1

字符不够时，自动会补空格

变长 varchar（字符数）

按数据大小分配储存空间，会比较慢

不能超出字符数，超出无法写入数据

不能不指定字符数

mysql> create table t5(

-> name char(15),

-> email varchar(30)

-> );

mysql>insert into t5 values("aa","aa@163.com"),("lucy","lucy@sina.com.cn");

枚举类型

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)

Enum只可选一个值

Set选多个值

mysql> create table t6(

-> name char(15),

-> age tinyint unsigned,

-> pay float(7,2),

-> sex enum("boy","girl","no"),

-> likes set("woman","money","gamy","eat") -> );

mysql> desc t6;

mysql> insert into t6 values("toutou",17,10000,"2","eat,gamy");

mysql> select \* from t6;

日期时间类型

关于日期时间字段

– 当未给 TIMESTAMP 字段赋值时,自动以当前系统时

间赋值,而 DATETIME 字段默认赋值为 NULL

• YEAR 年份的处理

– 默认用 4 位数字表示

– 当只用 2 位数字赋值时, 01~69 视为 2000~2069 ,

而 70~99 视为 1970~1999

mysql> create table t7(

-> name char(15),

-> syear year,

-> birthday date,

-> up\_class time,

-> party datetime

-> );

mysql> insert into t7 values("toutou",2000,20001114,123000,20180618213000);

