**数据库**

数据库（database）：依照某种数据模型进行组织并存放到储存器的数据集合

DBMS，Database Management System

数据库管理系统：用来操纵号和管理数据库的大型服务软件

DBS，Database System

数据库系统：即DB+DBNS，指带有数据库并整合了数据库管理软件的计算机系统

最著名、应用最广泛的开原数据库软件

服务默认端口：3306

安装mysql：

1. 安装包

mysql-community-client //

mysql-community-common

mysql-community-devel

mysql-community-embedded

mysql-community-embedded-compat

mysql-community-embedded-devel

mysql-community-libs

mysql-community-libs-compat

mysql-community-minimal-debuginfo

mysql-community-server

mysql-community-test

rpm -e --nodeps mariadb **//忽略依赖关系卸载包**

2.[root@client50 mysql-5.7.17]# rpm -Uvh mysql-community-\*.rpm

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

当有依赖关系的时候用yum list | grep

3.当出现上面yum关系依赖的时侯用yum查找相关依赖包

yum list | grep -i JOSN

再使用yum去安装依赖包，最后rpm -Uvh mysql-community-\*.rpm，安装就可以了

4.第一次初始化可能时间会比较长，服务会在 /var/lib/mysql/ 创建默认配置文件

systemctl start mysqld.server

端口号为3306

服务主配置文件：/etc/my.cnf

/var/lib/mysql/mysql.sock客户端访问数据库服务时传递数据，产生的套接字文件，启动服务默认生成

服务错误日志：/var/log/mysqld.log

pid-file进程号文件：/var/run/mysqld/mysqld.pid

进程所有者(管理用户)mysql，家目录是/var/lib/mysql

ls -ld /var/lib/mysql

drwxr-x--x 5 mysql mysql 4096 10月 15 10:19 /var/lib/mysql

文件目录/var/lib/mysql下的所有的文件权限都是mysql自己的

5.数据库安装好后, 默认只允许root账户在本机访问

安装mysql会生出初始密码

grep password /var/log/mysqld.log

2018-10-15T02:18:57.547715Z 1 [Note] A temporary password is generated for root@localhost: a14AG?na/w8-

本机访问数据库：mysql -h localhost -u root -p’a14AG?na/w8-’ //使用单引号对密码特殊字符进行屏蔽

1. 登陆进数据库后就只能使用：SQL命令（结构化查询语言）

取消命令的执行：\c

6.使用SQL命令修改密码

alter user root@"localhost" identified by "123qqq...A";

命令行，//模糊显示，只有修改完密码才允许查看

show variables like "%password%";

修改数据库服务密码策略，修改策略必须在修改密码之后才能修改

修改密码复杂度

set global validate\_password\_policy=0;

修改密码位数

set global validate\_password\_length=6;

此时修改简单密码成功

alter user root@"localhost" identified by '123456';

Query OK, 0 rows affected (0.00 sec)

永久设置

vim /etc/my.cnf

[mysqld]

validate\_password\_policy=0

validate\_password\_length=6

非交互式该密码：

mysqladmin -u root -p'old\_password' password "new\_password"

客户端把数据储存到服务器上的步

1. 客户端连接数据库服务器

[root@localhost ~]# mysql -h localhost -u root -p123456

1. 创建数据库

新建的数据库在/var/lib/mysql/都由一个同名的文件夹，所属组及所属者都是mysql

select database(); //显示当前所在的库

use 库名 //切换库

show databases; //显示已有的库

create database 库名; //创建新库

show table; //显示已有的表

drop database 库名; //删除库

create database teadb;

1. 创建数据表

每一个表在 /var/lib/mysql/库名/ 下对应者两个文件

create table 库名.表名(

字段名1 字段类型宽度) 约束条件,

字段名1 字段类型宽度) 约束条件,

...);

Use

create table stuinfo( id int, age int, name char(8) ); //建表

Drop table stuinfo;

show tables; //显示当前数据库里的表

desc stuinfo; //查看表结构

4.插入表记录

insert into teadb.stuinfo values(1,20,"xiaowu"),(2,19,"leifeng");

select \* from stuinfo; //查看表内容

update stuinfo set age=18 where name="leifeng"; //修改leifeng的age

delete from stuinfo where age='18'; //删除符合where的某一行

delete from stuinfo ; //删除表

5数据库表创建的误区

使用mkdir /var/lib/mysql/test 创建test库，mysql 无法在test库里创建表

原因：使用mkdir创建的文档所有者和所属者都属于root，这里mysql程序使用的是mysql用户，mysql用户对test没有写权限，将所有者和所属者改为mysql或者使其他人对test文件夹具有写权限就解决问题了

1. 解决中文插入数据的问题

show create 表名; //查看表创建时的语句

create table 学生表(姓名 char(20)) default charset=utf-8;

//使用utf-8编码方式进行表的创建，可以进行中文的插入

**Mysql数据类型**

数值型

字符型

枚举型

日期时间型

***数值型***

数值类型的宽度时显示私宽度

数值型默认的显示宽度是11位

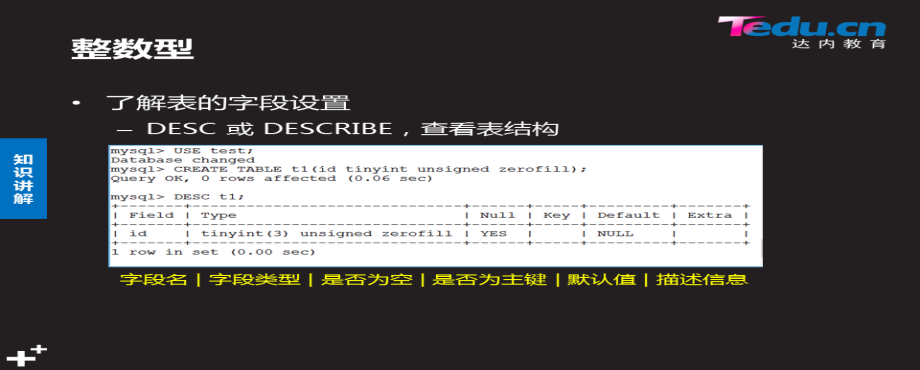
当数据的宽度小于显示宽度时，默认以空格在左边补位

当数据的宽度大于显示宽度时，以数据实际宽度显示

|  |  |  |
| --- | --- | --- |
| tinyint | 1 bit | 微小整数 |
| smallint | 2 bit | 小整数 |
| mediumint | 3 bit | 中整数 |
| int | 4 bit | 大整数 |
| bigint | 8 bit | 极大整数 |
| float | 4 bit | 单精度浮点数 |
| double | 8 bit | 双精度浮点数 |
| decimal | 对decimal(M,D)其中M为有效位数、D为小数位数、M应大于D、占用M+2bit | |
| unsigned | 无符号 |  |

表字段

Desc或者describe查看表结构



***字符型***

定长：char(字符数)

- 最大长度255字符数

- 不够指定字符数在右边用空格补齐

- 字符数段超出时，无法写入数据

变长：varchar(字符数)

- 不建议大量使用

- 根据存储数据的大小，去在磁盘开辟存储空间

- 最大存65532个字符

- 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

大文本类型：text/blob

- 字符数大于65535存储时使用

***日期时间类型***

日期时间，datetime

- 单用8个字节

- 范围：1000-01-01 ~ 9999-12-31

日期时间，timestamp

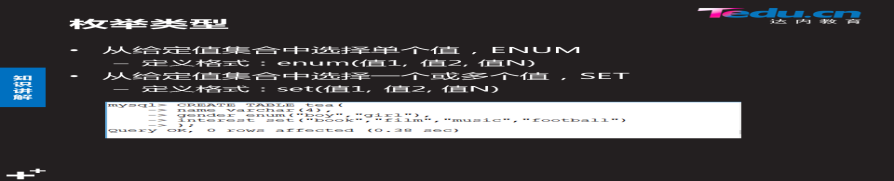
- 占用4个字节

未赋值时，结果以当前的系统时间为其赋值

时间函数

|  |  |
| --- | --- |
| now() |  |
| year() |  |
| month() |  |
| day() |  |
| sleep() |  |
| curdate() |  |
| curtime() |  |
| date() |  |
| time() |  |

***枚举型***

******