

# 数据库基础

mysql



软通大学  
ISOFTSTONE UNIVERSITY

# ● 本节任务

- ◆ 关系型数据库概念
- ◆ mysql的安装与客户端使用
- ◆ 数据库建表的基本操作
- ◆ 数据库表简单操作

# ● 本节目标

- ◆ 了解关系型数据库概念
- ◆ 学会mysql的安装与客户端使用
- ◆ 了解并掌握数据库表的建立，修改，删除等
- ◆ 掌握数据库表的增删改查



# 目录 CONTENTS

**1** 关系型数据库概念

mysql的安装与客户端使用 **2**

**3** 数据库建表的基本操作

数据库简单表操作 **4**

**5** 本章总结

# 01 关系型数据库



# ● 数据库介绍

## ◆ 数据库：

对大量信息进行管理的高效解决方案。  
按照数据结构来组织、存储和管理数据的库。

## ◆ 数据库系统（DBS,DATABASE SYSTEM）：

数据库（DB,DATABASE） + 数据库管理系统（DBMS,DATABASE MANAGEMENT SYSTEM)

## ◆ 关系型数据库

建立在关系模型上的数据库系统。

## ◆ 关系模型：

将日常生活中的实体（学生）和实体的属性（学生的学号，姓名等）保存到数据库中时，应该如何处理该实体结构。

- ◆ 数据结构可以规定，同类数据，结构一致，就是一个二维表格。
- ◆ 数据之间的关系可以设置。实体与实体之间的联系。

# ● 关系型数据库

关系型数据库，是指采用了关系模型来组织数据的数据库。

**关系模型**指的就是二维表格模型，而一个关系型数据库就是由二维表及其之间的联系组成的一个数据组织。

关系模型中的常用概念：

**关系**：可以理解为一张二维表，每个关系都具有一个关系名，就是通常说的表名。

**元组**：可以理解为二维表中的一行，在数据库中经常被称为记录。

**属性**：可以理解为二维表中的一列，在数据库中经常被称为字段。

**域**：属性的取值范围，也就是数据库中某一列的取值限制。

**关键字**：一组可以唯一标识元组的属性。数据库中常称为主键，由一个或多个列组成。

**关系模式**：指对关系的描述，其格式为：关系名（属性1，属性2，...，属性N）。在数据库中通常称为表结构。

# ● 主流关系型数据库

当前主流的关系型数据库：

Oracle、DB2、PostgreSQL、Microsoft SQL Server、Microsoft Access、MySQL、浪潮K-DB等。



---

## 02 mysql的安装和客户端使用



# • mysql安装

注意事项:

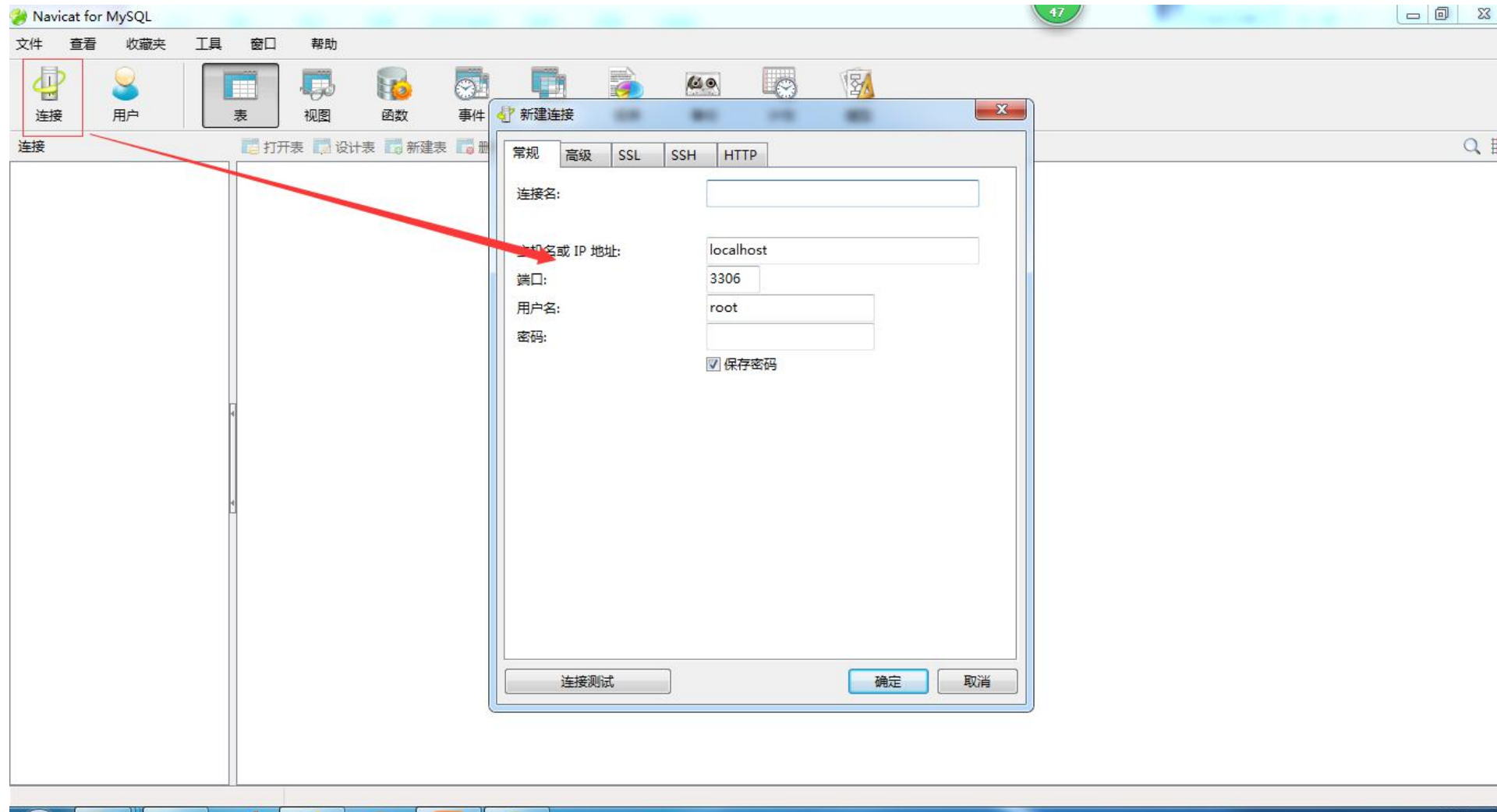
- 1: 选择安装包类型时, 选择custom;
- 2: 选择Enable TCP/IP NetWorking, 启用远程连接, , 数据库端口默认3306, 启动言责的语法配置Enable Strict Node;
- 3: 编码格式设置成utf-8;
- 4: 选择Install As Windows Service将mysql安装为windows服务, 还可以指定Service Name (服务标识名称), 选择Include Bin Dirextory in Windows Path将mysql的bin目录加入到Windows PATH (加入后, 就可以直接使用bin下的文件, 而不用指出目录名。
- 5: 输入root用户的密码, 并牢牢记住。选择Enable root access form remote machines启用root远程访问的功能。

安装失败解决方法:

打开c盘, 显示出隐藏文件夹ProgramData  
删除c:/ProgramData的MySQL文件夹, 重新安装。

# • 客户端工具Navicat for MySQL

绿色版客户端工具，双击Nacicat.exe打开，选择链接，输入ip和密码，测试成功点击确定；链接数据库成功。



## 03 数据库建表基本操作



# • mysql的基本操作（一）

启动MySQL服务器：net start MySQL

关闭MySQL服务器：net stop MySQL

进入mysql：mysql -uroot -p密码

数据库操作：

- ◆ 创建数据库

create database 数据库名；

- ◆ 查看数据库

show databases；

- ◆ 选择指定数据库

use 数据库名；

- ◆ 删除数据库

drop database 数据库名；

# mysql的基本操作（二）

MySQL数据表操作：

- 1、创建表  
create table 表名 (column\_name c
- 2、查看数据库中所有的表  
show tables; （前提是使用use data
- 3、查看数据表结构  
describe 表名;

create table语句的属性说明

属性	说明	属性	说明
column_name	字段名	Primary key	该列是否为主码
			列是否自动编号

```
graph LR; A[列类型] --> B[数值型]; A --> C[字符串型]; B --> D[整数型]; B --> E[小数型]; D --> D1[tinyint]; D --> D2[smallint]; D --> D3[mediumint]; D --> D4[int]; D --> D5[bigint]; E --> E1[浮点]; E --> E2[定点]; E1 --> E1a[float]; E1 --> E1b[Double]; E2 --> E2a[decimal]; C --> F[日期时间型]; C --> G[set]; C --> H[enum]; C --> I[blob]; C --> J[text]; C --> K[varchar]; C --> L[char]; F --> F1[year]; F --> F2[timestamp]; F --> F3[time]; F --> F4[date]; F --> F5[datetime];
```

<http://blog.csdn.net/itlover>



## • mysql的基本操作（三）

### 4、修改数据表结构

alter table 表名

add [column] create\_definition [first | after column\_name]      //添加新字段

add primary key (index\_col\_name,...)      //添加主码名称

alter [column] col\_name {set default literal | drop default}      //修改字段名称

change [column] old\_col\_name create\_definition      //修改字段名及类型

modify [column] create\_definition      //修改字段类型

drop [column] col\_name      //删除字段

drop primary key      //删除主码

rename [as] new\_tablename      //更改表名

### 5、删除指定数据表

drop table 表名;

## 04 数据库表操作





# ● 数据库表增、删、改

## ◆ 添加表数据

语法1: insert into 表名 values (值1, 值2, ...) (自增长的列应写null)

语法2: insert into 表名 (字段1, 字段2, ...) values (值1, 值2, ...)

语法3: insert into 表名 set 字段1=值1, 字段2=值2, ...

## ◆ 更新表数据

update 表名 set 字段1=值1 where 查询条件

若无查询条件，表中所有数据行都会被修改。

## ◆ 删除表数据

delete from 表名 where 查询条件

若无查询条件，表中所有数据行都会被删除。

TRUNCATE TABLE 表名

删除表中所有数据，而不记录单个行删除操作。

# • Truncate与delete注意事项

- ◆ TRUNCATE TABLE 在功能上与不带 WHERE 子句的 DELETE 语句相同：二者均删除表中的全部行。但 TRUNCATE TABLE 比 DELETE 速度快，且使用的系统和事务日志资源少。
- ◆ DELETE 语句每次删除一行，并在事务日志中为所删除的每行记录一项。TRUNCATE TABLE 通过释放存储表数据所用的数据页来删除数据，并且只在事务日志中记录页的释放。
- ◆ TRUNCATE TABLE 删除表中的所有行，但表结构及其列、约束、索引等保持不变。新行标识所用的计数值重置为该列的种子。如果想保留标识计数值，请改用 DELETE。如果要删除表定义及其数据，请使用 DROP TABLE 语句。
- ◆ 对于由 FOREIGN KEY 约束引用的表，不能使用 TRUNCATE TABLE，而应使用不带 WHERE 子句的 DELETE 语句。由于 TRUNCATE TABLE 不记录在日志中，所以它不能激活触发器。
- ◆ TRUNCATE TABLE 不能用于参与了索引视图的表。
- ◆ 对用TRUNCATE TABLE删除数据的表上增加数据时，要使用UPDATE STATISTICS来维护索引信息。
- ◆ 如果有ROLLBACK语句，DELETE操作将被撤销，但TRUNCATE不会撤销。

# ● 数据库表简单查询

## 1、查询表数据

`select * from 表名;`

## 2、限制查询记录数

`select * from 表名 limit[start] length`

start:表示从第几行记录开始输出，0表示第1行

## 3、条件查询

`select * from 表名 where`

# ● 主键与UNIQUE约束

mysql主键（primary key）的属性：

主键具有唯一性：是指一张表里只能有一个主键；

主键作用：主键primary key是为了唯一标识一个字段，使其唯一且不能为NULL，自动生成索引；

主键自增：如果把表的主键设为auto\_increment类型，数据库就会自动为主键赋值.

隐含定义：如果这些列没有被明确地定义为NOT NULL，MySQL会隐含地定义这些列。

主键其实也是索引，甚至在MySQL的术语里面“键”就等于“索引”，

UNIQUE约束的属性：

唯一性：定义了UNIQUE约束的字段中不能包含重复值，

可为空：在UNIQUE约束的字段上可以包含空值.

扩展：unique就是唯一,当你需要限定你的某个表字段每个值都唯一,没有重复值时使用.

比如说,如果你有一个student\_Info表,并且表中有个身份证（card）的column,那么你就可以指定该字段unique(card).

# ● 主键和UNIQUE约束的区别

关系：

主键=NOT NULL +UNIQUE键的结合；

NOT NULL和UNIQUE约束最好的结合。

如果这些列没有被明确地定义为NOT NULL，MySQL会隐含地定义这些列。

区别：

(1) 唯一性约束所在的列允许空值，但是主键约束所在的列不允许空值。

(2) 可以把唯一性约束放在一个或者多个列上，这些列或列的组合必须有唯一的。但是，唯一性约束所在的列并不是表的主键列。

(3) 唯一性约束强制在指定的列上创建一个唯一性索引。在默认情况下，创建唯一性的非聚簇索引，但是，也可以指定所创建的索引是聚簇索引。

(4) 建立主键的目的是让外键来引用。

(5) 一个表最多只有一个主键，但可以有很多唯一键

---

## 05 本章总结



# ● 本章总结

- ◆ 学会mysql的安装与使用
- ◆ 学会mysql控制台新建和删除数据库操作
- ◆ 学会新建、修改和删除表操作
- ◆ 学会在表中添加、修改和删除表数据
- ◆ 了解并掌握主键及UNIQUE约束