

一、表的概念

由生活中的仓库来理解数据库

仓库 -> 数据库

例：

JD 松江仓库

1. 电子产品
2. 日用品
3. 化妆品
4. ...

结论：仓库的下边，其实是有其他东西，这些东西归类，就叫做表

二、引用数据库和查看数据库中的表

指定要操作的仓库：

use 仓库名;

```
use student;      # 仓库名为student
```

查看仓库中的表：

show 表名;

```
show student;
```

三、创建&查看表

(1) 通过“create”关键字可以创建表：

```
create table 表名(  
关键字1 数据类型1,  
关键字2 数据类型2,  
...  
关键字n 数据类型n  
);
```

```
mysql> create table student(      # 这里创建了表student，用半括号，将表中的元素(即字段)写在后边的行中  
-> id int,                        #将表中的元素一行一个列出来是很好的习惯  
-> name varchar(30),              #mysql中的字符串用数据类型varchar，括号内为长度  
-> age int                        #反括号和分号不能丢  
-> );  
Query OK, 0 rows affected (0.01 sec)
```

(2) 通过“show tables”查看已创建的表：

show tables;

```
mysql> show tables;
+-----+
| Tables_in_shek_school |
+-----+
| student                |
+-----+
1 row in set (0.00 sec)
```

(3) 如果是一些比较具体的信息，还可以通过"show create"关键字进行操作：

show create table 表名;

```
mysql> show create table student;
+-----+-----+
| Table | Create Table
+-----+-----+
| student | CREATE TABLE `student` (
  `id` int(11) DEFAULT NULL,
  `name` varchar(30) DEFAULT NULL,
  `age` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=gbk |
+-----+-----+
1 row in set (0.00 sec)
```

四、创建表 (进阶)

如果是要装逼或者是在企业中的一些比较专业的操作，则应像如下代码一样进行创建表：

```
# 解释放在下文中
mysql> create table if not exists teacher(
  -> id int auto_increment primary key comment '主键id',
  -> name varchar(30) not null comment '老师的名字',
  -> phone varchar(20) comment '电话号码',
  -> address varchar(100) default '暂时未知' comment '住址'
  -> )engine=innodb;
Query OK, 0 rows affected (0.01 sec)
```

- id ,name, age 称为字段
- auto_increment 自动增长
- primarily key 主键 最主要的靠它来区分学生这张表
- comment 注释
- not null 不能为空
- default 默认值
- innodb 一种数据库引擎

五、查看表结构

"desc"可以将表的结构展示出来：

desc 表名;

```
mysql> desc teacher;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id    | int(11)       | NO   | PRI | NULL    | auto_increment |
| name  | varchar(30)   | NO   |     | NULL    |                |
| phone | varchar(20)   | YES  |     | NULL    |                |
| address | varchar(100) | YES  |     | 暂时未知 |                |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

六、删除表

先创建一些比较垃圾的表以供练习，这里创建了俩垃圾表之后的样子：

```
mysql> show tables;
+-----+
| Tables_in_shek_school |
+-----+
| dsbs                   |
| stu                   |
| student                |
| teacher                |
+-----+
4 rows in set (0.00 sec)
```

跟数据库的创建删除其实差不多，都是用的"drop"关键字：

drop table if exists 表名;

drop table if exists 表名1, 表名2, ...,表名n;

```
mysql> drop table if exists dsbs, stu, saogang;
Query OK, 0 rows affected, 1 warning (0.01 sec)
```

我们再看：

```
mysql> show tables;
+-----+
| Tables_in_shek_school |
+-----+
| student                |
| teacher                |
+-----+
2 rows in set (0.00 sec)
```

OK，删除成功。

七、修改表

1. 增加字段:

先创建一个表"saogang":

```
mysql> create table saogang(  
    -> id int);  
Query OK, 0 rows affected (0.01 sec)
```

(1) 常规添加

向表中添加一个字段(field), 使用"alter"字段:

alter table 表名 add 字段 字段数据类型;

```
mysql> alter table saogang add length int;  
Query OK, 0 rows affected (0.01 sec)  
Records: 0 Duplicates: 0 Warnings: 0  
  
mysql> alter table saogang add name varchar(20);  
Query OK, 0 rows affected (0.01 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc saogang;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type          | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| id    | int(11)       | YES  |     | NULL    |       |  
| length | int(11)       | YES  |     | NULL    |       |  
| name  | varchar(20)   | YES  |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
3 rows in set (0.00 sec)
```

(2) 插入到指定位置

使用"add"、"after"关键字:

alter table 表名 add 字段名 字段数据类型 after 目标字段名;

```
mysql> alter table saogang add gender varchar(1) after length;  
Query OK, 0 rows affected (0.01 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc saogang;  
+-----+-----+-----+-----+-----+-----+  
| Field | Type          | Null | Key | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| id    | int(11)       | YES  |     | NULL    |       |  
| length | int(11)       | YES  |     | NULL    |       |  
| gender | varchar(1)    | YES  |     | NULL    |       |  
| name  | varchar(20)   | YES  |     | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
4 rows in set (0.00 sec)
```

如果想放在开头，使用"first"关键字：

alter table 表名 add 字段名 字段数据类型 first;

```
mysql> alter table saogang add height int first;
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc saogang;
```

Field	Type	Null	Key	Default	Extra
height	int(11)	YES		NULL	
id	int(11)	YES		NULL	
length	int(11)	YES		NULL	
gender	varchar(1)	YES		NULL	
name	varchar(20)	YES		NULL	

5 rows in set (0.00 sec)

2. 删除字段

还是使用"drop"关键字：

alter table 表名 drop 字段名;

```
mysql> alter table saogang drop height;
Query OK, 0 rows affected (0.01 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc saogang;
```

Field	Type	Null	Key	Default	Extra
id	int(11)	YES		NULL	
length	int(11)	YES		NULL	
gender	varchar(1)	YES		NULL	
name	varchar(20)	YES		NULL	

4 rows in set (0.00 sec)

3. 修改字段

(1) 改字段名字+类型：

使用"change"关键字：

alter table 表名 change 原字段名 新字段名 新数据类型;

```
mysql> alter table saogang change length depth int(11);
Query OK, 0 rows affected (0.00 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> desc saogang;
```

Field	Type	Null	Key	Default	Extra
-------	------	------	-----	---------	-------

```

+-----+-----+-----+-----+-----+
| id    | int(11) | YES | | NULL | | |
| depth | int(11) | YES | | NULL | | |
| gender | varchar(1) | YES | | NULL | | |
| name   | varchar(20) | YES | | NULL | | |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

(3) 只改类型

使用"modify"关键字:

alter table 表名 modify 字段名 新数据类型;

```

mysql> alter table saogang modify gender varchar(4);
Query OK, 0 rows affected (0.00 sec)
Records: 0 Duplicates: 0 Warnings: 0

```

```
mysql> desc saogang;
```

```

+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| id    | int(11)       | YES  |     | NULL    |       |
| depth | int(11)       | YES  |     | NULL    |       |
| gender | varchar(4)    | YES  |     | NULL    |       |
| name   | varchar(20)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

4. 修改表名

使用"rename to"关键字:

alter table 表名 rename to 新表名;

```

mysql> alter table student rename to students;
Query OK, 0 rows affected (0.01 sec)

```

```
mysql> show tables;
```

```

+-----+
| Tables_in_shek_school |
+-----+
| saogang                |
| students               |
| teacher                |
+-----+
3 rows in set (0.00 sec)

```

