

2. 数据库与表的操作



1. 创建、查看、删除数据库

创建数据库

命令: CREATE DATABASE <database_name>;

例: CREATE DATABASE jxgl;

```
mysql> create database jxgl;  
Query OK, 1 row affected (0.00 sec)  
  
mysql>
```



1. 创建、查看、删除数据库

查看数据库

命令: SHOW DATABASES;

```
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| jxgl |  
| mysql |  
| oqmd_v1_2 |  
| performance_schema |  
| sys |  
+-----+  
6 rows in set (0.01 sec)  
  
mysql>
```



1. 创建、查看、删除数据库

删除数据库

命令： DROP DATABASE <database_name>;

例： DROP DATABASE jxgl;

```
mysql> drop database jxgl;  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> show databases;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| mysql |  
| oqmd_v1_2 |  
| performance_schema |  
| sys |  
+-----+  
5 rows in set (0.00 sec)  
  
mysql> _
```



1. 创建、查看、删除数据库

使用数据库

命令：USE <database_name>;

例：USE jxgl;

```
mysql> CREATE DATABASE jxgl;  
Query OK, 1 row affected (0.00 sec)  
  
mysql> use jxgl;  
Database changed  
mysql>
```



2. 创建、查看、修改、删除基本表

创建基本表

命令:

```
CREATE TABLE <表名>
```

```
(
```

```
列名1 数据类型1 [列级完整性约束条件1],
```

```
列名2 数据类型2 [列级完整性约束条件1],
```

```
...
```

```
[表级完整性约束条件]
```

```
);
```



2. 创建、查看、修改、删除基本表

查看当前数据库中所有表：

命令： `show tables;`

查看某个表的结构：

命令： `desv table_name;`

查看某个表中的所有值：

命令： `select * form table_name;`



2. 创建、查看、修改、删除基本表

MySQL中主要的数据类型：整型

类型	大小	范围（有符号）	范围（无符号）	用途
TINYINT	1 byte	(-128, 127)	(0, 255)	小整数值
SMALLINT	2 bytes	(-32 768, 32 767)	(0, 65 535)	大整数值
MEDIUMINT	3 bytes	(-8 388 608, 8 388 607)	(0, 16 777 215)	大整数值
INT或INTEGER	4 bytes	(-2 147 483 648, 2 147 483 647)	(0, 4 294 967 295)	大整数值
BIGINT	8 bytes	(-9,223,372,036,854,775,808, 9 223 372 036 854 775 807)	(0, 18 446 744 073 709 551 615)	极大整数值

2. 创建、查看、修改、删除基本表

MySQL中主要的数据类型：浮点型与小数

类型	大小	范围（有符号）	范围（无符号）	用途
FLOAT	4 bytes	(-3.402 823 466 E+38, -1.175 494 351 E-38), 0, (1.175 494 351 E-38, 3.402 823 466 351 E+38)	0, (1.175 494 351 E-38, 3.402 823 466 E+38)	单精度(浮点数值)
DOUBLE	8 bytes	(-1.797 693 134 862 315 7 E+308, -2.225 073 858 507 201 4 E-308), 0, (2.225 073 858 507 201 4 E-308, 1.797 693 134 862 315 7 E+308)	0, (2.225 073 858 507 201 4 E-308, 1.797 693 134 862 315 7 E+308)	双精度(浮点数值)
DECIMAL	M+2(MYSQL>=3.23)	依赖于M和D的值	依赖于M和D的值	小数值

2. 创建、查看、修改、删除基本表

MySQL中主要的数据类型：浮点型与小数

DECIMAL(M,D): M指定小数点左边和右边可以存储的十进制数字的最大个数，最大精度38。D指定小数点右边可以存储的十进制数字的最大个数。小数位数必须是从0 到 M之间的值。



2. 创建、查看、修改、删除基本表

MySQL中主要的数据类型：文本

类型	大小	用途
CHAR	0-255 bytes	定长字符串
VARCHAR	0-65535 bytes	变长字符串
TINYBLOB	0-255 bytes	不超过 255 个字符的二进制字符串
TINYTEXT	0-255 bytes	短文本字符串
BLOB	0-65 535 bytes	二进制形式的长文本数据
TEXT	0-65 535 bytes	长文本数据
MEDIUMBLOB	0-16 777 215 bytes	二进制形式的中等长度文本数据
MEDIUMTEXT	0-16 777 215 bytes	中等长度文本数据
LOBLOB	0-4 294 967 295 bytes	二进制形式的极大文本数据
LONGTEXT	0-4 294 967 295 bytes	极大文本数据

2. 创建、查看、修改、删除基本表

MySQL中主要的数据类型：时间

类型	大小 (bytes)	范围	格式	用途
DATE	3	1000-01-01/9999-12-31	YYYY-MM-DD	日期值
TIME	3	'-838:59:59'/'838:59:59'	HH:MM:SS	时间值或持续时间
YEAR	1	1901/2155	YYYY	年份值
DATETIME	8	1000-01-01 00:00:00/9999-12-31 23:59:59	YYYY-MM-DD HH:MM:SS	混合日期和时间值
TIMESTAMP	4	1970-01-01 00:00:00/2038 (结束时间是第2147483647秒，北京时间2038-1-19 11:14:07，格林尼治时间 2038年1月19日 凌晨 03:14:07)	YYYYMMDD HHMMSS	混合日期和时间值，时间戳

2. 创建、查看、修改、删除基本表

MySQL中约束条件: not null

not null 保证字段的值不能为空

MySQL中约束条件: default

default 表示默认约束，保证字段总会有值，即使没有插入值，都会有默认值

MySQL中约束条件: unique

unique 保证唯一性但是可以为空，比如座位号



2. 创建、查看、修改、删除基本表

MySQL中约束条件: check

check 为检查性约束, 例: s_sex CHAR(1) CHECK(s_sex='男' OR s_sex='女')

MySQL中约束条件: primary key

primary key 表示主键约束, 同时保证唯一性和非空

MySQL中约束条件: foreign key

unique 为外键约束, 用于限制两个表的关系, 保证从表该字段的值来自于主表相关联的字段的值



2. 创建、查看、修改、删除基本表

P79 例3.1:

```
mysql> create table s
-> (sno char(5) primary key,
-> sn varchar(8) not null,
-> sex char(2) not null check(sex in ('男','女')),
-> age int not null check(age>0),
-> dept varchar(20),
-> constraint sn_u unique(sn))
-> ;
Query OK, 0 rows affected (0.01 sec)
mysql>
```



2. 创建、查看、修改、删除基本表

P79 例3.2:

```
mysql> create table c  
      -> (cno char(5) not null primary key,  
      -> cn varchar(20) not null,  
      -> ct int check(ct>=1));  
Query OK, 0 rows affected (0.01 sec)
```



2. 创建、查看、修改、删除基本表

P79 例3.3:

```
mysql> create table sc
-> (sno char(5) not null,
-> cno char(5) not null,
-> score numeric(3,0),
-> constraint s_c_p primary key (sno, cno),
-> constraint s_f foreign key (sno) references s(sno),
-> constraint c_f foreign key (cno) references c(cno));
Query OK, 0 rows affected (0.01 sec)

mysql>
```

