

计算机编程基础与实践

—结构化查询语言与MySQL

July 8, 2012

1 简介

- 基本概念
- SQL命令分类
- MySQL SQL命令测试环境

2 MySQL基本使用

- 连接登录
- 创建并选择数据库
- 创建数据表
- 增删改查

3 MySQL字符集问题

- 查看默认字符集
- 修改默认字符集

基本概念

结构化查询语言(Structured Query Language,简称SQL), 主要用途是构造各种数据库系统操作命令, 用来查询, 修改或删除数据库的各种数据。

这些命令中, 最重要的是:

- **SELECT**
- **INSERT**
- **UPDATE**
- **DELETE**

SQL命令分类

SQL命令可以分为以下3大类别:

- DML(Data Manipulation Language,数据处理语言):主要包括SELECT, INSERT, UPDATE, DELETE以及另外几个用来从数据表读出数据, 把数据存入数据表或是对数据表里的现有记录进行修改的命令。(重点)
- DDL(Data Definition Language,数据定义语言): 主要包括CREATE TABLE, ALTER TABLE等用来定义和改变数据库结构的命令。
- DCL(Data Control Language,数据控制语言): 主要包括GRANT, REVOKE以及几个用来帮助人们设置和调整数据库访问机制的SQL命令。

MySQL SQL命令测试环境

- ① 对于简单的SQL命令，最佳的测试环境是使用MySQL自带的命令解释器。
 - Unix/Linux操作系统中，可在shell中使用mysql命令来启动。
 - Windows中，可以通过菜单程序Program|MySQL|MySQL Server n.n|MySQL Command Client来启动它
- ② 对于又长又复杂的SQL命令，MySQL命令解释器使用不方便，最好选用一个第三方图形界面的MySQL客户端,比如MySQL Query Browser或者phpMyAdmin,进行SQL命令的编辑和调试。

连接登录

命令行输入:

```
mysql -h [HostName] -u [UserName] -p
```

系统询问密码:

Enter password:

输入密码后进入MySQL命令解释器:

```
mysql>
```

创建并选择数据库

创建数据库

```
mysql>CREATE DATABASE testdb;  
Query OK, 1 row affected (0.00 sec)
```

选择数据库

```
mysql>USE testdb;  
Database changed
```

注意:

- 在MySQL命令解释器中，需要对SQL命令末尾加上分号。
- 除了数据库和数据表的名字外，MySQL不区分字母的大小写。

*可使用SHOW DATABASES;来显示已有的数据库。

创建数据表

错误方式创建表

```
mysql>CREATE TABLE roster;  
ERROR 1113 (42000): A table must have at least 1 column
```

正确方式

```
mysql>CREATE TABLE IF NOT EXISTS roster(Id CHAR(3)  
PRIMARY KEY,Name VARCHAR(20), Intro VARCHAR(200));  
Query OK, 0 rows affected (0.00 sec)  
mysql>SELECT * FROM roster;  
Empty set (0.00 sec)
```

*可使用SHOW TABLES;来显示已有的数据表。

注意：数据表不能同名，所以CREATE TABLE后需要加IF NOT EXISTS,此时如果数据库中已存在同名的表，那么只会出现一个warning,否则会出现error!

增

往已创建的空表中添加记录

```
mysql>INSERT INTO roster(Id, Name, Intro)  
VALUES("007","xiayf","He is a man!");
```

Query OK, 1 row affected (0.00 sec)

```
mysql>INSERT INTO roster(Id, Name, Intro)  
VALUES("008","xiayz","He is a gentleman!");
```

Query OK, 1 row affected (0.00 sec)

查询结果:

```
mysql> SELECT * FROM roster;  
+-----+-----+-----+  
| Id   | Name  | Intro                |  
+-----+-----+-----+  
| 007  | xiayf | He is a boy!         |  
| 008  | xiayz | He is a boy too!    |  
+-----+-----+-----+  
2 rows in set (0.00 sec)
```

删

从表中删除一条记录

```
mysql> DELETE FROM roster WHERE Name = "xiayz";  
Query OK, 1 row affected (0.00 sec)
```

查询结果

```
mysql> SELECT * FROM roster;
```

+	—	+	—	-	+	—	—	—	+
	Id		Name			Intro			
+	—	+	—	-	+	—	—	—	+
	007		xiayf			He is a boy!			
+	—	+	—	-	+	—	—	—	+

1 row in set (0.00 sec)

改

查询表中所有记录

```
mysql> SELECT * FROM roster;
```

更改ID="009" 的人员的Name

```
mysql> UPDATE roster SET Name="xiayongqing"  
WHERE Id="009";
```

Query OK, 1 row affected (0.00 sec)

Rows matched: 1 Changed: 1 Warning: 0

```
mysql> SELECT * FROM roster;
```

Id	Name	Intro
007	xiayf	He is a boy!
008	xiayz	He is a boy too!
009	xiayq	He is my brother!

3 rows in set (0.00 sec)

```
mysql> UPDATE roster SET Name = "xiayongqing" WHERE Id = "009";  
Query OK, 1 row affected (0.00 sec)  
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> SELECT * FROM roster;
```

Id	Name	Intro
007	xiayf	He is a boy!
008	xiayz	He is a boy too!
009	xiayongqing	He is my brother!

3 rows in set (0.00 sec)

查

- SELECT Name FROM roster;
- SELECT Name FROM roster WHERE Id = "007";
- SELECT Id, Intro FROM roster WHERE Name = "xiayf";
- SELECT * FROM roster ORDER BY Name;
- SELECT Name, Intro FROM WHERE Name LIKE 'xia%';¹

¹字符%是代表任意字符串的通配符

默认字符集

查看默认字符集

```
mysql>SHOW VARIABLES LIKE 'character%';
```

结果

Variable_name	Value
character_set_client	utf8
character_set_connection	utf8
character_set_database	utf8
character_set_filesystem	binary
character_set_results	utf8
character_set_server	utf8
character_set_system	utf8
character_sets_dir	/usr/share/mysql/charsets/

修改默认字符集

- SET character_set_client = utf8;
- SET character_set_connection = utf8;
- ...

一般就算设置了表的默认字符集为utf8并且通过UTF-8编码发送查询，你会发现存入数据库的仍然是乱码。问题就出在这个connection连接层上。解决方法是在发送查询前执行下面这句：

```
SET NAMES 'utf8';
```

它相当于下面的三句指令：

- SET character_set_client = utf8;
- SET character_set_results = utf8;
- SET character_set_connection = utf8;

THANK YOU !

Proud to use \LaTeX and Beamer.