**C++操作利用MSQL API连接和操作数据库**

1. **C++连接和操作MySQL的方式**

在Windows平台，我们可以使用ADO、ODBC或者MySQL API进行连接和操作。[ADO](http://baike.baidu.com/view/43014.htm" \t "http://baike.baidu.com/_blank) (ActiveX Data Objects，[ActiveX数据对象](http://baike.baidu.com/view/185276.htm" \t "http://baike.baidu.com/_blank)）是Microsoft提出的一个用于存取数据源的COM组件。它提供了程序语言和统一数据访问方式OLE DB的一个中间层，也就是Microsoft提出的应用程序接口（[API](http://baike.baidu.com/view/16068.htm" \t "http://baike.baidu.com/_blank)）用以实现访问关系或非关系数据库中的数据。

ODBC(Open DataBase Connection)开放式系统互连，是一种数据库访问协议，提供了访问数据库的API接口。基于ODBC的应用程序，对数据库操作不依赖于具体的DBMS，不直接与DBMS打交道，所有数据库操作由对应DBMS的ODBC驱动程序完成，即：系统中不需要安装DBMS系统，如SQL SERVER 2005，但必须有SQL SERVER 2005的ODBC驱动程序，然后在ODBC管理器中注册数据源后，就可以在应用程序中通过ODBC API访问该数据库。ODBC数据库访问技术只适用于windows系统，因为需要在ODBC驱动程序管理器中进行数据源注册，而只有windows才集成了ODBC驱动程序管理器(“控制面板/管理工具/数据源”)。

ADO具有跨系统平台特性，它直接对DBMS数据库进行操作，即系统中必须有DBMS，但不需要驱动程序，不需要注册数据源，所以具有很好的可移植性。

所以，在Linux平台如何连接和使用MSQL数据库呢？我们可以同样可以使用ADO、unixODBC和MySQL API。这里不再赘述前两者的用法，读者可自行研究实践，下文将详细讲解MySQL创建数据库和C++利用MSQL API连接和操作数据库。

1. **MSQL数据库的设计和建立**

**2.1数据表的设计**

MySQL数据库管理系统（DBMS）中，包含的MySQL中定义数据字段的类型对你数据库的优化是非常重要的。MySQL支持多种类型，大致可以分为三类：数值、日期/时间和字符串(字符)类型。

下面以大学熟悉的学生选课管理系统中用到的数据库为例，来设计相应的数据表。主要有三张表：学生表，课程表和选课表。

**学生表设计:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段(Field) | 类型(Type) | 可空(Null) | 键(Key) | 默认值(Default) | 其他(Extra) |
| 学号（studentNo） | VARCHAR(12) | N | PRI | NULL |  |
| 姓名（name） | VARCHAR(12) | N | N | NULL |  |
| 学院（school） | VARCHAR(12) | N | N | NULL |  |
| 年级（grade） | VARCHAR(12) | N | N | NULL |  |
| 专业（major） | VARCHAR(12) | N | N | NULL |  |
| 性别（gender） | Boolean | N | N | NULL |  |

**课程表设计:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段(Field) | 类型(Type) | 可空(Null) | 键(Key) | 默认值(Default) | 其他(Extra) |
| 课程号（courseNo） | VARCHAR(10) | N | PRI | NULL |  |
| 课程名（courseName） | VARCHAR(10) | N | N | NULL |  |
| 课时（hour） | TINYINT Unsigned | N | N | NULL |  |
| 课程学分（credit） | TINYINT Unsigned | N | N | 2 |  |

**选课表设计:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段(Field) | 类型(Type) | 可空(Null) | 键(Key) | 默认值(Default) | 其他(Extra) |
| 编号（id） | INT Unsigned | N | PRI | NULL | auto\_increment |
| 学号（studentNo） | VARCHAR(12) | N | MUL | NULL |  |
| 课程号（courseNo） | VARCHAR(10) | N | MUL | NULL |  |
| 选课时间（time） | TIMESTAMP | N | N | CURRENT\_TIMESTAMP |  |

对于上面三张数据表的设计，你会发现：

1. MySQL在为数据定义字符串(字符)类型时，需要在类型名称的后面的小括号内指明最长字符数，即TypeName(M)，这里的M指的是字符数，而不是数据占用的字节数。例如varchar(12)，则表述存储的字符数不能超过12 ，超过12则截断，小于12个字符则以实际占用的存储空间来存储，这样就节省了存储空间。
2. 在对选课表的设计时，我们添加外键约束，可以使得两张表关联，保证数据的一致性和实现一些级联操作。
3. MySQL中有四种Key: Primary Key（主键）, Unique Key（唯一键）, Index（索引）和Foreign Key（外键）。四者的区别如下：

**定义：**

主键：唯一标识一条记录，不能有重复，不允许为空。

外键：表的外键是另一表的主键, 外键可以有重复的, 可以是空值。

索引：该字段没有重复值记录，但可以有一个空值。

唯一键：唯一标识一条记录，不能有重复，可以为空。

**作用：**

主键：用来保证数据完整性。

外键：用来和其他表建立联系，以保证数据的一致性和级联操作。

索引：是提高查询排序的速度。

唯一键：用来用防止数据插入的时候重复。

**个数：**

主键：主键只能有一个。

外键：一个表可以有多个外键。

索引：一个表可以有多个唯一索引。

唯一键：一个表可以有多个。

从中可以看出，索引和唯一键很相似，二者的区别在于作用不同，当然，在MySQL具体实现上应该也是不同的。

关于四者的理解，上未参考到权威的资料，请读者保持怀疑的态度接受。

**2.2数据库的建立**

在安装完mysql之后，我们要建立自己的数据库。下面将详细地一步一步演示如何创建上面设计好的数据库。在MySQL模式下，使用status命令可以查看MySQL版本，本人使用的MySQL版本：5.6.30 MySQL Community Server (GPL)。

关于mysql大小写问题，mysql命令是不区分大小写的。数据表的表名在windows下不分，linux下分。数据表的字段名在windows和linux下都不分。

1. **登录mysql DBMS**

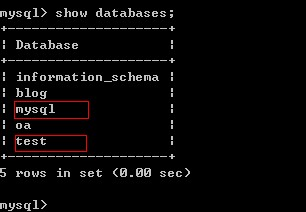
Linux命令行中输入如下命令：

|  |
| --- |
| mysql -hlocalhost -uroot -p123456 |

说明：依次指明登录mysql的主机地址，用户和用户密码。

1. **使用show查看当前mysql服务器上存在什么数据库**

|  |
| --- |
| show databases; |



说明：在进入mysql模式下，使用mysql命令时，每条语句要以分号结束。但是use [DatabaseName]却不用，可能是mysql的一个小bug。类似于quit和exit退出mysql的命令，不需要一个分号。（如果你喜欢，你可以用一个分号终止这样的语句）。

1. **创建数据库**

|  |
| --- |
| mysql> CREATE DATABASE StudentCourse; |

1. **使用use语句访问数据库**

|  |
| --- |
| mysql>use StudentCourse; |

1. **创建数据表student**

|  |
| --- |
| mysql> create table student(  studentNo varchar(12) not null,  name varchar(12) not null,  school varchar(12) not null,  grade varchar(12) not null,  major varchar(12) not null,  gender boolean not null,  primary key(studentNo)  )engine=MISAM default charset=utf8; |

观察上面的建表语句，需注意以下几点：

（a）其中，布尔类型boolean在MySQL是以类型tinyint(1)来实现，这里的1指代的是数据显示时最短长度。

实际上tinyint(1)是可以插入-128到127之间的其它数值。因为mysql数据库中以 ：数据类型(m)来定义数据类型，其中 数字m在不同的数据类型中表示含义是不同的。 整型数系统已经限制了取值范围，tinyint占一个字节、int占4个字节。所以整型数后面的m不是表示的数据长度，而是表示数据在显示时显示的最小长度（长度为字符数）。

tinyint(1) 这里的1表示的是最短显示一个字符。tinyint(2) 这里的2表示的是最短显示两个字符，但这里光设置m是没有意义的，你还需要指定当数据少于长度m时用什么来填充，比如zerofill（表示有0填充）。设置tinyint(2) zerofill你插入1时他会显示01。设置tinyint(4) zerofill你插1时他会显示0001。

（b）还要注意一点是，使用engine可指明引擎，如果省略了engine语句，则使用默认的引擎(MYISAM)。MYSQL支持三个引擎：ISAM、MYISAM和HEAP。另外两种类型INNODB和BERKLEY（BDB），也常常可以使用。

（c）设置default charset指明mysql数据表的编码方式，不显示指定编码方式的话，数据表的默认编码方式一般是latin1。也可以通过如下命令查看数据表的编码方式：

show create table t\_user\_friend;

如果需要转换数据表的编码格式，使用如下命令：

alter table t\_user\_friend convert to character set utf8;

如果想查看当前数据库的编码格式：

mysql>status;或者show variables like 'character%' ;

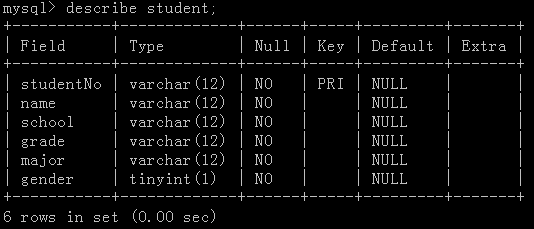
为了验证所建立的数据表是否是按照期望的方式创建的，使用如下命令：

#查看表中的列

SHOW COLUMNS FROM student;

#或者直接使用describe

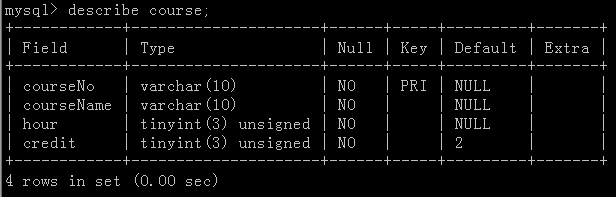
describe student;



1. **创建数据表course**

|  |
| --- |
| mysql> create table course(  studentNo varchar(12) not null primary key,  courseNo varchar(10) not null,  hour tinyint unsigned not null,  credit tinyint unsigned not null default 2  )engine=MYISAM default charset=utf8; |

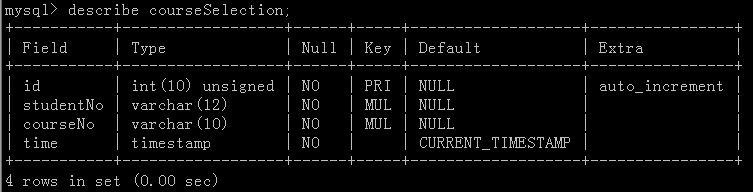
同样可以使用describe查看表信息：



1. **创建数据表courseSelection**

|  |
| --- |
| mysql> create table courseSelection(  id int unsigned not null auto\_increment primary key,  studentNo varchar(12) not null,  courseNo varchar(10) not null,  time timestamp not null default CURRENT\_TIMESTAMP,  FOREIGN KEY(studentNo) REFERENCES student(ISBN) ON UPDATE CASCADE ON DELETE CASCADE,  FOREIGN KEY(courseNo) REFERENCES course(courseNo) ON UPDATE CASCADE ON DELETE CASCADE  )engine=MYISAM default charset=utf8 AUTO\_INCREMENT=0; |

查看表信息：



**（8）其它关于数据库和数据表的操作命令**

删除数据库：

mysql> DROP DATABASE 库名;

删除数据表：

mysql> DROP TABLE 表名；

将表中记录清空：

mysql> DELETE FROM 表名;

1. **MSQL数据库的连接和操作**

下面将讲解利用MySQL API来编写我们自己的用于访问MySQL的中间件，也是我们自己的组件。我们的组件在应用程序和MySQL数据库的结构如下图所示：

Application

Own Component

MySQL

MySQL API

下面就来设计和实现我们自己的C++访问MySQL数据库的组件。

3.1头文件的设计

3.2具体实现

3.3使用demo

**参考文献：**

[1]http://www.runoob.com/mysql/mysql-data-types.html

[2]http://www.cnblogs.com/BeginMan/p/3249472.html：Mysql 查看、创建、更改 数据库和表

[3]http://zhidao.baidu.com/link?url=6f\_spGQnJfdZX-MT0S2YsZekRafC7Ku2hwPUiEVGY5wKzP4RfUINk\_ITwZi1-cfv3fbgWT1ZSn7wADQ7qG0upn-focpl9rz3lVuZOKv3Rxa：php里tinyint(1)为什么还能插入99这个值？搜索

[4]http://www.cnblogs.com/sopc-mc/archive/2011/11/01/2232212.html:MySQL引擎

[5]http://blog.csdn.net/aoogoo/article/details/4473466：MySQL中的四种Key、

[6]http://www.cnblogs.com/ywb-lv/archive/2012/03/12/2391860.html:SQL的主键和外键约束

[7]