

**本科实验报告**

|  |  |
| --- | --- |
| 课程名称： | 数据库原理 |
| 姓 名： | 闫徐天任 |
| 学 院： | 计算机学院 |
| 系： | 计算机 |
| 专 业： | 软件工程 |
| 学 号： | 3160103827 |
| 指导教师： | 陈岭 |

2018年 6 月 28 日

# 总体设计报告

## 基本信息

* 编程语言：C++；
* 源代码规模：6000行左右；
* 实现的功能：MIniSQL大程大纲中的所有功能。

## 代码风格

* 左大括号换行；
* 缩进4个空格；
* 变量名：variable\_name;
* 函数名：camelCase;
* 类名：ThisIsClass;
* 错误处理方式：使用C++异常。

## 系统体系结构

我们的体系结构参考了设计指导中的体系结构，但是略有修改。

Interface & Interpreter

API

Record Manager

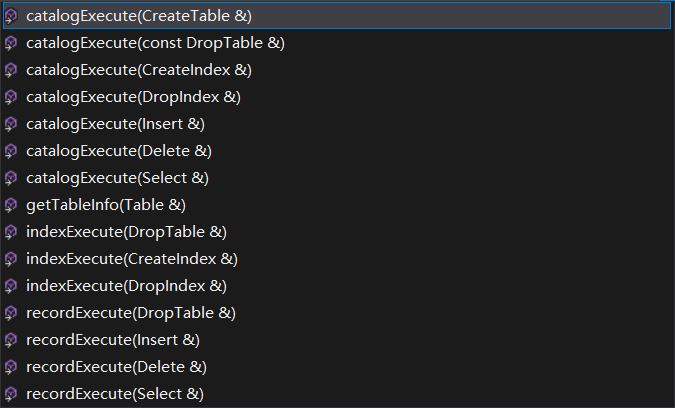
Index Manager

Catalog Manager

Buffer Manager

## 围绕API的消息传递方式

Interpreter根据用户输入，可能调用以下API：



如上图所示，record manager、index manager和catalog manager模块分别去实现自己的Execute函数。这些函数接收的参数均是一个类的对象的引用。这些类是我们传递信息的关键所在。这些类的继承关系如下图所示：

Table类

string table\_name;

int column\_num;

string column\_name[];

ColumnType column\_type[];

int string\_length[];

bool is\_unique[];

string primary\_key;

int exist\_index\_num;

string all\_index\_name[];

Select类

查询条件

Insert类

待插入的项

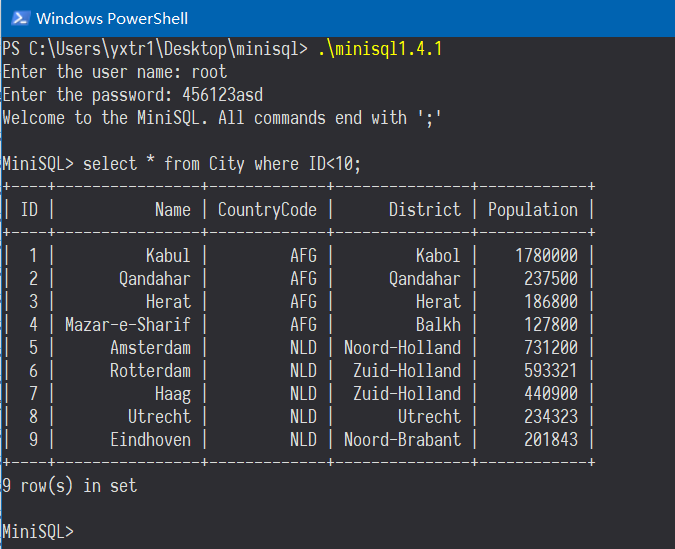
Delete类

删除条件

CreateTable类、DropTable类只是为方便进行函数重载，这两个继承类没有额外的成员变量。一般情况下，Interpret可以得到表名与操作名。于是Interpreter首先构造一个空的、与用户操作对应的类的对象，将表名和其他必要信息填入该对象、再调用API，通过Catalog Manger填入其他表的信息，再调用API让其他模块完成操作。

## 用户界面设计

我们的MiniSQL是命令行程序，以下是我们的MiniSQL的运行截图。



## 组内分工

闫徐天任（组长）：Record Manager、部分API、整合

曾辉：Buffer Manager

赵丰年：Index Manager、部分API

范佳臣：Interpreter、Interface

吕明远：Catalog Manager、部分API