

## 实战项目 – RDBMS

### 一、项目名称

RDBMS - 关系数据库管理系统

### 二、项目目的

1. 检验 Module03 – C++语言 中的 面向对象编程-泛型编程-C++标准库 部分的掌握程度；
2. 重点锻炼：STL 容器、STL 算法、字符串处理、I/O 流、泛型编程；
3. 加强数据持久化的概念；
4. 锻炼项目的架构能力；
5. 加深对关系数据库的了解；
6. 作为后续版本的前驱（后续版本将加入多线程、分化客户端和服务端）。

### 三、知识覆盖范围

1. C++编程语言、包括标准库的各个方面；
2. 关系数据库相关的知识。

### 四、开发方式与时间

个人，时间：72 hours+。

### 五、建议的开发环境

Linux、Eclipse、vim、make、g++、gdb、CVS。

### 六、简单需求描述

目前有很多优秀的关系数据库产品，如：Oracle、DB2、Sybase、MS SQLServer、MySQL、ProgesQL 等，这些数据库管理系统都在一定程度上支持 SQL 语言，特别是 SQL92 标准，普遍支持程度较高。此次项目要求实现一个简易的（功能受限的）关系数据库管理系统，通过 insert、delete、update、select 4 种 SQL 语句对数据库中的数据执行：增、删、改、查 操作。

本系统的主要模块：数据定义模块、数据操纵模块、数据持久化模块、SQL 语句解析模块、UI 模块。

## 1，数据定义模块：

数据定义模块职责：定义数据库、表、表字段、表字段的属性、表与表之间的关系：

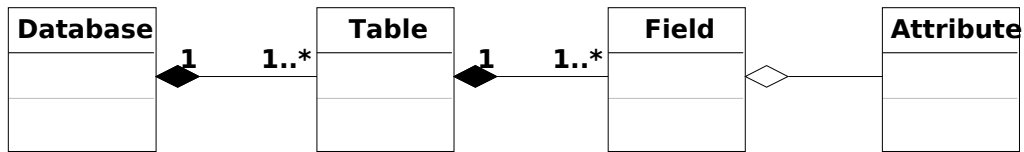


图 1-1 数据对象之间的关系

关于表、表字段和字段属性：

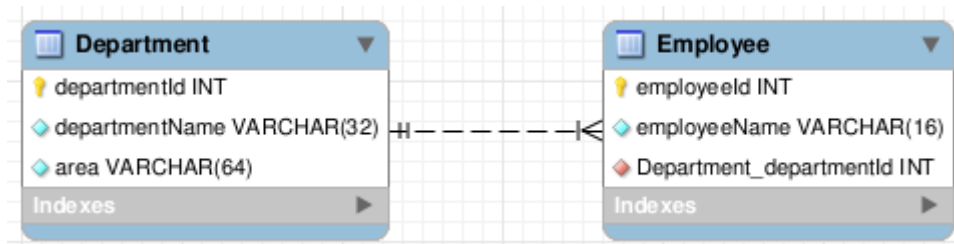


图 1-2 表、字段、字段属性、表与表的关系

字段属性说明，下面以字段 Department.departmentId 为例：

- A、名称：departmentId
- B、类型：INT
- C、是否为主键：YES
- D、可否为 NULL：NO
- E、能否重复(UNIQUE)：NO
- F、默认值：(无)

表与表之间的关系说明，以图 1-2 为例：

- A、一个 Department 中有多个 Employee，也就是每个 Employee 必属于某一个 Department
- B、表 Employee 中持有表 Department 的主键(PK)作为外键(FK)

## 2，数据操纵模块

数据操纵模块的职责：支持下列 SQL 语句，对数据库中的数据进行增、删、改、查等操作。

1. 实现两种数据定义语句：create database (创建数据库)、create table (创建表)

如下面的数据定义语句创建一个名为 Company 的数据库和 Department 的表：

```

-- 创建数据库
CREATE DATABASE Company;

-- 创建表
CREATE TABLE Department (
  departmentId INT NOT NULL ,

```

```
departmentName VARCHAR(32) NOT NULL ,
area VARCHAR(64) NOT NULL ,
PRIMARY KEY (departmentId)
);
```

## 2. insert 语句：插入数据

如下面语句往表 Department 中插入一条记录：

```
INSERT INTO Department VALUES( 1, 'RND', 'Shanghai' );
```

## 3. update 语句：修改数据

如下面语句修改表 Department 中的一条或多条记录：

```
-- 修改单条记录
UPDATE Department SET area = 'Beijing' WHERE departmentId = 1;

-- 部分修改
UPDATE Department SET area = 'Beijing' WHERE departmentId < 3;

-- 修改全部记录
UPDATE Department SET area = 'Beijing';
```

## 4. delete 语句：删除记录

如下面的语句删除表 Department 中的一条或多条记录：

```
-- 删除单条记录
DELETE FROM Department WHERE departmentId = 1;

-- 部分删除
DELETE FROM Department WHERE departmentId > 1;

-- 删除全部记录
DELETE FROM Department;
```

## 5. select 语句：查询语句

如下面的语句查询表 Department 中一条或多条记录的一个或多个字段：

```
-- 查询表 Department 中所有记录
SELECT * FROM Department;

-- 查询表 Department 中 departmentId = 1 记录
SELECT * FROM Department WHERE departmentId = 1;

-- 查询表 Department 中所有记录中的 departmentName 字段
SELECT departmentName FROM Department;

-- 查询表 Department 中 departmentId = 1 的记录中的 departmentName 字段
```

```
SELECT departmentName FROM Department WHERE departmentId = 1;
```

### 3，数据持久化模块

数据持久化模块的职责：高效、完整的将内存中的数据保存到磁盘（文件），且能在应用程序重新启动之后能正确读取磁盘中的数据。

### 4，SQL 语句解析模块（SQL 语法解析器）

具体的 SQL 语法请参考 SQL92 标准相关的资料。

### 5，UI 模块

Ui 模块负责与本系统的交互：接受用户的输入、将操作的结果呈现给用户。

#### 1. 输入界面

```
mydb> SELECT Name FROM t1
-> WHERE Age > 12;
```

语句的结束符：一条 SQL 语句以特定的结束符结束，本次我们将结束符定为分号(;)。

第一提示符：如上面的第一行中，**mydb>** 是第一提示符；

第二提示符：如上面的第二行中，**->** 是第二提示符，一个 SQL 语句没有结束的情况下换行，将出现第二提示符。

#### 2. 输出界面

输出的结果 1，普通命令输出：

```
mydb> insert into t1 values('t',22,'kwarph@gmail.com');
Query OK, 1 row affected (0.00 sec) -- 输出
```

输出结果 2，查询命令的输出：

```
mydb> select * from t1;
+-----+-----+-----+
| Name | Age | Email |
+-----+-----+-----+
| t    | 22 | kh@gmail.com |
| ken  | 25 | xy@gmail.com |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

## 七、交付

1. 类设计和类图；
2. 完整的项目代码，包括完整的目录结构、C++源代码、Makefile、项目可能使用的各类 Script 等等。