

数据库大作业说明文档

一、项目简介

1. 项目目标

本项目为数据库原理课程的project，目标旨在实现一个关系型数据管理系统。

2. 项目成员

2017013574 那森 2017013581 杨松霖 2017013591 邱圆辉

二、功能实现

1. 存储模块

1. 实现功能

- 利用序列化与反序列化实现记录的持久化
- 支持对记录进行插入、删除、修改、查询操作
- 支持Int, Long, Float, Double, String五种数据类型
- 实现了文件的页式存储

2. 实现方式

存储模块的实现主要与Table类与Page类紧密相关。关键的类方法与其功能说明列举如下：

- Table

```
1 // 将磁盘文件中恢复数据库信息
2 private void recover();
3
4 // 向表中插入新的行，插入不合法时抛出ValueException
5 public void insert(String[] values);
6
7 // 删除满足条件predicate的行
8 public int delete(Predicate<Row> predicate);
9
10 // 删除此表中的所有行
11 public void deleteAll();
12
13 // 尝试在表中查找并更新对应的行，当指定的行不存在时抛出ValueException
14 public int update(String[] columnNames, String[] values,
15 Predicate<Row> predicates);
16
17 // 当数据库操作完成时提交事务，并调用serialize方法进行数据的持久化存储
18 public void commit();
19
20 // 序列化表中数据并存储在指定磁盘页面中
21 private void serialize(Page page);
22
23 // 反序列化磁盘数据并以行列表的形式返回表中数据
24 private ArrayList<Row> deserilize(File file);
```

- Page

```

1 // 向当前页面插入一个大小为size的词条entry，并更新此页面已存储数据的大小
2 public void insert(Entry entry, int size);
3
4 // 从当前页面删除一个大小为size的词条entry，并更新此页面已存储数据的大小
5 public void delete(Entry entry, int size);

```

2. 元数据管理模块

1. 实现功能

- 实现了表的创建、删除与修改
- 实现了数据库的创建、删除与切换
- 实现了表和数据库的元数据的持久化
- 支持重启数据库时从持久化数据中恢复上一次系统信息

2. 元数据管理模块的实现主要与Database类与Manager类紧密相关。关键的类方法与其功能说明列举如下：

- Database

```

1 // 将当前数据库的所有数据进行持久化存储
2 private void persist();
3
4 // 以columns为schema，创建一个名为name的新表
5 public void create(String name, Column[] columns);
6
7 // 删除名为name的表
8 public void drop(String name);
9
10 // 待实现
11 public String select(QueryTable[] queryTables);
12
13 // 从数据库元数据文件中恢复当前数据库的所有表信息
14 private void recover();

```

- Manager

```

1 // 从数据库元数据文件中读取所有数据库的信息
2 private void recoverDatabase();
3
4 //
5 private void createDatabaseIfNotExists()
6
7 // 删除名为name的数据库
8 private void deleteDatabase(String name);
9
10 // 将当前连接的数据库更改为名为name的数据库
11 public void switchDatabase(String name, Context context);
12
13 // 创建一个名为name的新数据库
14 private void createDatabase(String name);

```

3. 查询模块

待实现。

4. 事务与恢复模块

待实现。

5. 通信模块

待实现。

6. 异常处理模块

待实现。

四、使用说明

待实现。