实验报告

一、实验环境

Ubuntu 22.04
Jdk 1.8
Ant 1.10.12

二、实验流程

Exercise 1：Filter and Join

Predicate.java、JoinMedicate.java 和 Filter.java 的实现非常简单。运算符以 DbIterators 作为其基本迭代器来扫描元组，并返回满足特定条件或由特定算法产生的元组。Filter 中实现的是返回满足过滤条件的元组，Predicate 是其构造函数的一部分。Join 中实现的是两个元素的自然联结，JoinPredicate 是其构造函数的一部分。其中 Join 联结实现的方式是嵌套循环。对于这两个操作都有对应的谓词辅助类 Predicate、JoinPredicate。帮助去对比元组字段之间是否符合条件。

Exercise 2：Aggregates

代码很简单。要完成的就是 SQL 中的分组（GROUP BY）与聚合（aggregator）的操作。

Exercise 3: HeapFile Mutability

在 HeapPage、HeapFile、BufferPool 中实现 deleteTuple、insertTuple 的功能。最终修改的是 HeapPage 中 header，修改 slot 的占用情况。因为删除元组时，实际上不会删除任何元组，而是将其槽标记为未使用。元组插入扫描第一个未使用的插槽，并将新的元组覆盖到其中。要注意的点是：对于修改后的页，应及时记录变更为脏页。

Exercise 4: Insertion and deletion

类比实现 insertTuple（）和 deleteTuple（）。

运行结果：

ant test

```
BUILD SUCCESSFUL
Total time: 12 seconds
```

ant systemtest

```
BUILD SUCCESSFUL
Total time: 16 seconds
```

Query walkthrough：

```
ubuntu@ubuntu-virtual-machine:/mnt/hgfs/Database/Lab3-SimpleDB Operators/lab3$ java -jar dist/simpledb.jar parser catalog.txt
Added table : data with schema INT_TYPE(f1),INT_TYPE(f2)
Computing table stats.
Done.
SimpleDB>
SimpleDB>
SimpleDB> select d.f1, d.f2 from data d;
Started a new transaction tid = 0
Added scan of table d
Added select list field d.f1
Added select list field d.f2
d.f1    d.f2
-----------------
Field_Name:d.f1==>Value:1
Field_Name:d.f2==>Value:10

Field_Name:d.f1==>Value:2
Field_Name:d.f2==>Value:20

Field_Name:d.f1==>Value:3
Field_Name:d.f2==>Value:30

Field_Name:d.f1==>Value:4
Field_Name:d.f2==>Value:40

Field_Name:d.f1==>Value:5
Field_Name:d.f2==>Value:50


 5 rows.
Transaction 0 committed.
---------------
0.07 seconds
```

三、困难

这次完成的较快，只用了一天时间，可能因为有些是和前面类似的，写起来比较熟练。而且是对操作的实现，比较具体不会那么抽象，好理解。基本没怎么遇到困难。