CMPE 321 - Introduction to Database Systems

Spring/2018

Project 2 - Implementing Storage Manager System

Ahmet Yasin Alp
2015400213

1. Introduction

My system consists of 10Mb database file that stores pages, SystemCatalog.txt and DataFiles for each type that stores id's of pages which store records of that type.

Init function fill database with zeros.

For page reading and writing there are readPage and writePage functions.

For creating new type, code updates SystemCatalog and create new data file.

For creating new record, if emptyPage in SystemCatalog points a page then go that page and find empty slot for record. If that page becomes full, search a not full page in data file, if it fails and data file is not full then reserve a new page for that type and change data file and SystemCatalog. If there is no empty page remains then set emptyPage in SystemCatalog to invalid(negative) value.

For deleting a type, delete data file, delete line in SystemCatalog, fill all pages that is used by deleted type with zero.

For deleting a record, search page by page in datafile, if record is found then change record header. If that page is full change this in page header.

If a page is allocated a type, it can not be reallocated until the the type is deleted.

List of commands:

INIT : initialize database full zeros

CREATE TYPE <TypeName> <Number of fields> <FiledNames>...: Creates new type

CREATE RECORD <Type> <Fields> : Create new record

SEARCH <Type> <PrimaryKey>: Search records by primary key

LIST TYPE : list all types

LIST RECORD <type> : list all records of that type

DELETE RECORD <Type> <PrimaryKey> : delete record

DELETE TYPE <Type> : delete type

EXIT: quits program

2. Changes

- There is no file header. It moved to System Catalog
- In page header there is field for isUsed to store if that page is used for store a type. Because one page consists of only one type of records
- In record header isDeleted changes to isExists. Easy to initialize
- Page header is 6 bytes.(1byte isUsed , 1byte isFull, 4bytes numberOfRecords)
- Record header is 1 byte

3. Sample Usage

```
INITY
Wrong Syntax. Please try again
INIT
Database is created
CREATE TYPE TYPE1 3 FIRSTFIELD SECONDONE THIRDONE
New Record Type was created with name: TYPE1
CREATE RECORD TYPE1 34 1 9
Record was created
LIST TYPE
 All types in Database:
--> TYPE1
LIST RECORD TYPE1
34 1 9
CREATE TYPE TYPE2 1 ANAOTHERFIELD
New Record Type was created with name: TYPE2
LIST TYPE
 All types in Database:
--> TYPE1
--> TYPE2
CREATE RECORD TYPE2 45
Record was created
CREATE RECORD TYPE2 13
Record was created
LIST TYPE
 All types in Database:
--> TYPE1
--> TYPE2
LIST RECORD TYPE1
34 1 9
LIST RECORD TYPE2
45
13
```

```
SEARCH TYPE1 34
Record found: TYPE1 34 1 9
LIST RECORD TYPE1
34 1 9
LIST RECORD TYPE2
45
13
DELETE RECORD TYPE2 45
Record found and deleted
LIST RECORD TYPE2
13
DELETE TYPE TYPE1
Deleted type: TYPE1
LIST TYPE
 All types in Database:
--> TYPE2
CREATE TYPE TYPE3 2 Q W
New Record Type was created with name: TYPE3
LIST TYPE
 All types in Database:
--> TYPE2
--> TYPE3
CREATE RECORD TYPE3 34 56
Record was created
LIST RECORD TYPE3
34 56
```

SystemCatalog.txt:

TYPE2 1 ANAOTHERFIELD 1 1

TYPE3 2 Q W 0 1

TYPE2_DataFile.txt:

1

TYPE3_DataFile.txt:

0

4. Conclusion

There is no error handling. In most cases if user try to exceed limits , program does nothing.

It is not space efficient because there is few page reallocation.

To sum up, like every design it has some ups and downs. I tried to keep the design simple so it is not well optimized but easy to maintain and implement