Davisbase - Team Yellow Project Document

The project structure contains the 'src' and the 'data' folders. The data folders contains the data files ('catalog')that are populated when data is entered into the database and read for further operations. The src folder contains the operational source code. The code is composed of the following five classes:

WelcomePage.java

It takes care of the user display prompt and version.

Utility.java

It contains all the functions accessed across different classes in the project. It includes the functions for parsing input string and converting it to a desired format, access paths for the files stored in the project, converting input data into a data structure used by the program for representing records and identifying input data for appropriate parsing.

QueryHandler.java

This class deals with display of table logic, 'show table' query, 'drop table' query, 'update' query and display of the help menu.

ParseQuery.java

This class parses the input command from the command prompt, and calls the appropriate function(query) depending on the keywords recognized in the command. It also extracts data from the input string that will be required to pass as arguments to the corresponding function. Individual functions have been defined for different queries. The queries dealt with from this class are Parse Create command, parse create Index command, parse Insert command, parse 'show' keyword, parse update command, parse delete command.

MyDatabase.java

This is the main class for our project. It also takes care of building internal representations of the database and the columns data (This project uses a hashmap for the internal representation of records in the table. This class initializes the hashmaps.)

Btree.java

This class deals with all operations associated with the 'b' and the 'b+' trees. It has functions for the actual working of every query defined in the system at the file level. It creates and deletes internal nodes of the tree, creates and deletes leaf nodes of the tree , initializes the tree, checks for keys on columns of the table, searches the tree to find/ read individual records and their values,and maintains the page headers on every operation,