Project Description

Part 1: Implement a paged file system and operations of records

- Primary folder: rbf, which includes pfm.h, pfm.cc, rbfm.h, and rbfm.cc
- Involved classes in this part: PagedFileManager (manage file I/O; pfm.h), FileHandle (represent a data file; pfm.h), RecordBasedFileManager (manage records; rbfm.h)
- Key functions implemented by me: insertRecord (insert a record in a data file; rbfm.cc)
- Project Report: Please refer to "Project1 Report.pdf", which elaborates on our DBMS's design of Internal Record Format and Data Page Format.
- Please refer to the following link for more details: https://grape.ics.uci.edu/wiki/public/wiki/cs222p-2017-fall-project1-description

Part 2: Implement operations of tables and tuples

- Primary folder: rm, which includes rm.h and rm.cc
- Involved classes in this part: RelationManager (manage catalogue, tables, and tuples; rm.h), RM_ScanIterator (search tuples in a table; rm.h), RBFM_ScanIterator (search records in a data file; rbfm.h)
- Key functions implemented by me: updateRecord (update a record in a data file; rbfm.cc), readAttribute (read a specific attribute of a record; rbfm.cc)
- Project Report: Please refer to "Project2 Report.pdf", which elaborates on our DBMS's design of Catalogue and Data File Format.
- Please refer to the following link for more details: https://grape.ics.uci.edu/wiki/public/wiki/cs222p-2017-fall-project2-description

Part 3: Implement operations of B+ tree indexes

- Primary folder: ix, which includes ix.h and ix.cc
- Involved classes in this part: IndexManager (manage B+ tree indexes; ix.h), IX_ScanIterator (search data entries in an index file; ix.h), IXFileHandle (represent an index file; ix.h)
- Key functions implemented by me: insertEntry (insert a data entry in an index file; ix.cc), printBtree (print a B+ tree in a JSON record format; ix.cc), testBtreeValid (verify that a resulting B+ tree is valid; ix.cc)
- Project Report: Please refer to "Project3 Report.pdf", which elaborates on our DBMS's design of Index Entry Format, Index Page Format, and Index File Format.
- Please refer to the following link for more details: https://grape.ics.uci.edu/wiki/public/wiki/cs222p-2017-fall-project3

Part 4: Implement relational operators

- Primary folder: qe, which includes qe.h and qe.cc
- Involved classes in this part: RM_IndexScanIterator (rm.h), Iterator (qe.h), Filter (qe.h), Project (qe.h), BNLJoin (qe.h), INLJoin (qe.h), Aggregate (qe.h)
- Key functions implemented by me: BNLJoin (join two tables using Block Nested-Loop Join; qe.cc), Aggregate (calculate MIN, MAX, SUM, AVG, and COUNT of a field; qe.cc)
- Project Report: Please refer to "Project4 Report.pdf", which elaborates on the implementation details of each relational operator.
- Please refer to the following link for more details: https://grape.ics.uci.edu/wiki/public/wiki/cs222p-2017-fall-project4