Database API Specs

Database()

* Constructor for Database
* Parameters: none (creates empty database)
* Todo: implement
* Exceptions: Error if there are arguments.

addTable(string name, Table t1)

* Adds a table (t1) to the database with name (name)
* Parameters:
  + name – name of table
  + t1 – table object to add
* todo: implement
* Exceptions: Error exact parameters not used.

dropTable(string name)

* Removes a table from the database
* Parameters:
  + name – name of table to drop
* todo: implement
* Exceptions: Error if no arguments or name is not a string

vector<string> listTables()

* Returns a vector (of strings) that stores all the names of the tables
* Parameters: none
* Todo: implement
* Exceptions: Error if arguments are provided

vector<Table> getTables()

* Returns a vector (of tables) of all tables in the database
* Parameters: none
* Todo: implement
* Exceptions: Error if arguments are provided

Table Query(string SELECT, string FROM, string WHERE)

* Returns a table that matches the search query of SELECT, FROM, and WHERE
* Parameters: string SELECT, string FROM, string WHERE
* Todo implement
* Exceptions: Error if arguments don’t match

Table API Specs

Table(int rows, int columns)

* Constructor for Table. It allows for an empty Table if rows and columns are 0.
* Parameters:
  + Rows – the number of rows to be created
  + Columns – the number of columns to be created
* Todo: implement
* Exceptions: must be exactly 2 integer arguments.

void addAttribute(string attributeName)

* Adds a column to the table with an attribute name
* Parameters:
  + attributeName – name of column to be added
* Todo: implement

void deleteAttribute(string attributeName)

* Deletes a column of the table with an attribute name
* Parameters:
  + attributeName – name of column to be deleted
* Todo: implement

void insert(Record newRow)

* Adds a new record a table
* Parameters:
  + newRow –name of record to be added to the table
* Todo: implement

vector<string> getAttributes()

* Returns a vector of names of the attributes in a table
* Parameter: none
* Todo: implement

int size()

* Returns the number of records found in the table
* Parameter: none
* Todo: implement

void specifyKey(string attributeKey)

* Returns individual records using an attribute key. Iterates through records to find a match.
* Parameters:
  + attributeKey – key of attribute for iterator to match with
* Todo: implement

Table crossJoin(Table t1, Table t2)

* Cross joins two tables and returns a resulting table
* Parameter:
  + t1, t2 – tables to be cross joined
* todo: implement

Table naturalJoin(Table t1, Table t2)

* Natural joins two tables and returns a resulting table
* Parameter:
  + t1, t2 – tables to be natural joined
* todo: implement

int count(string attributeName)

* counts how many non-null entries in a column
* Parameter:
  + attributeName – column to count non-null entries
* todo: implement

int min (string attributeName)

* returns the minimum integer value in a column
* Parameters:
  + attributeName – column to be checked for the minimum
* todo: implement

int max (string attributeName)

* returns the maximum integer value in a column
* Parameters:
  + attributeName – column to be checked for the maximum
* todo: implement

Record API Specs

Record(int size = 0);

* Constructor for record. Initializes each of the elements with an empty string.
* Parameter:
  + int size, dictates how many elements the record has.
* Todo: implement

int size();

* Returns the number of elements in a record.
* Parameters: none
* Todo: implement

string at(int index);

* Returns the string containing the data at a given index
* Parameter:
  + int index, the location of the data being accessed
* Todo: implement

Void set(int index, string data);

* Writes data to a given index
* Parameters:
  + int index, the location of the data being written to
  + string data, the data that is being written
* Todo: implement