CSCI 5408
DATA MANAGEMENT AND WAREHOUSING

Project TinyDb
Module-1

Group members:
Vraj Shah
Shivani Uppe
Kenee Patel

1. Selection of Linear Data Structures

For query and data processing, we have selected two data structures which are ArrayList and HashMap.

a. **ArrayList (Linear):** It allows dynamic resizing, which is useful when the number of rows in a table can vary. It provides fast indexed access, making it efficient to retrieve rows based on their position. Iteration over elements is straightforward, which is beneficial for operations like filtering rows based on conditions.
b. **HashMap (Non-linear):** It allows storing data in key-value pairs, which aligns well with storing column values by their names. Provides constant-time performance for put and get operations, making it efficient for accessing column values in a row. It can store different types of data as values, supporting various column types and constraints.

2. Custom File Format for Persistent Storage

To persist data in a custom file format, we will design a simple yet structured plain text format that ensures data integrity and readability.

Custom File Format Design:

- The first line will be the header, listing all columns with their data types and constraints, separated by a delimiter (|)

  Example:
  id: int [PRIMARY KEY] | name: String [] |

- Each subsequent line represents a row, with column values separated by a delimiter (|).

  Example:
  id: 1 | name: Data |
  id: 2 | name: Base |

Plain text format is chosen for simplicity and readability. This format ensures data can be easily inspected and edited if necessary. The structure with a header line followed by data lines ensures that the schema is clearly defined, and rows are stored in a consistent manner. This format supports the addition of new rows and easy retrieval of data.