

Hugo Mazariego

Prof. Zhao

CS 457:1001

20 October 2021

Assignment 2: Now With Tuples

For the second assignment, I was tasked with giving Hugo's Database the ability to insert, delete, modify, and select tuples with basic conditions. After a bit of grueling effort and a small amount of hair-tearing, the database can now filter tuples using the WHERE key, insert tuples into tables, delete tuples from tables, and modify tuples in tables. As before, to use this program you must type "python3 ./HugosDatabase" This will bring up the UI to begin the program. It will then ask if you want to type your commands or if you would like to read in from a file. Please do not use "python3 ./HugosDatabase<PA2_test.sql>" as that will break the program. Start the program as specified above, press "f" to enter the file reader, and type either the local or absolute path of the file. If tests do not work as expected, try deleting the database storage directory and trying again. This will cause the program to make an empty database storage directory. Additionally, the SQL script provided on Canvas has a few spelling errors, trying to query the table "products" when the table is named Products. This will cause the database to raise an alarm. If you do not want to change the sql file by hand, a corrected copy is included in the zip file. The only changes included are the correct name of the table Products in some queries and some spacing between the values in the insert for my own visual comfort. The removal of these spaces will not affect the outcome.

The way the program stores tuples is rather messy, though that is due to my inexperience with Python. The table class has a list of lists that contains all the values of the tuples. Since tuples don't have their own names, I thought early on it'd be easier to store them as such. This was a bad design because I frequently got confused as to if my index was for a tuple or a value in the tuple. Lesson learned from that, just because Python can store lists within lists, does not make it convenient. Time will be dedicated to refactoring my old code as it feels messy and unintelligible. In order to permanently store tuples on a device, each tuple is given its own file to store its data in. This makes insertion, modification, and deletion easier as instead of going through a table file and reading line by line and hoping not to mess something up, it's easier to simply overwrite the information in a single file that only contains information of one tuple. Likewise, it's easier to simply delete a file with tuple information than to comb through a file and carefully remove the right one. As for the naming convention for the tuple files, that was a little difficult at first as tuples had no names, so it would need to be something unique. At first, I attempted to name them based on their index in the list but realized that deletion could cause errors. In order to save time and energy, I named each tuple all its values concatenated together.