

# **CS-457 Database Management Systems**

## **Project 1**

March 5th, 2023

### **Install:**

<https://github.com/Stehfyn/cs457>

### **Contact:**

Stephen Foster

[stephenfoster@nevada.unr.edu](mailto:stephenfoster@nevada.unr.edu)

## System Design

The database for this project is designed such that a file is analogous to a database, and that special syntax denotes tables within the database. The following is a visual representation:

```

/databases-+
    +-db1
    |
    +-db2
    |
    +-db3
    |
    +-db4
    |
    +-db5
    |
    db1:
    +-----+
    |[table1]
    |a4 = int
    |a5 = varchar(20)|
    |
    |[table2]
    |a6 = float
    |a7 = char(30)
    +-----+
  
```

As is expressed above, “databases” is a file directory, with db1-db5 being text files with no extension. On the right-hand side of the above visual, we can see tables denoted using angular brackets ([ ]), with attributes and constraints on the left and right side of the equals (=), respectively.

## System Implementation

The database for this project is written using python3.9+, and has a file structure that is the following:

```

/src-+
    +-assignment1.py
    |
    +-database.py
    |
    +-database_impl.py
    |
    +-database_parser.py
  
```

- **assignment1.py** - The database program entrypoint. Depending on how the program was called (interactively, non-interactively, certain command-line

arguments, etc.), this file will execute the appropriate functionality found in **database.py**.

- **database.py** - The database program manager. This file contains the DatabaseManager class and related functionality to keep track of which database is in use, as well as the implementations of the batch processor, interpreter, and graphical user interface. Additionally, this file implements the wrapper functions that “connect” the output of the database parser to the appropriate database function.
- **database\_impl.py** - The database function implementations. Functions for creating databases, dropping databases, creating tables, dropping tables, etc. exist here. These are the database functions that **database.py** connect with output from **database\_parser.py**.
- **database\_parser.py** - The database parser that either accepts or rejects raw input as valid database operation language. This parser both tokenizes arguments as well as ensures valid table syntax, then returns the output to **database.py** to connect to the proper database functionality in **database\_impl.py**.

## System Interface

### Install

For quick install and execution, please clone <https://github.com/Stehfyn/cs457> and follow the instructions in the readme. Detailed instructions are as follows:

1. Install python3.9 via:

```
sudo apt-get update && sudo apt-get install python3.9
```

2. Install external dependencies via:

```
python3.9 -m pip install -r requirements.txt
```

3. Navigate to /src containing assignment1.py and execute:

```
python3.9 assignment1.py
```

## Interpreter

The database project can be executed in interpreter mode, by invoking:

```
stehfyn@ubuntu:~/Desktop/cs457/src$ python3.9 assignment1.py  
#
```

## Batch Processor

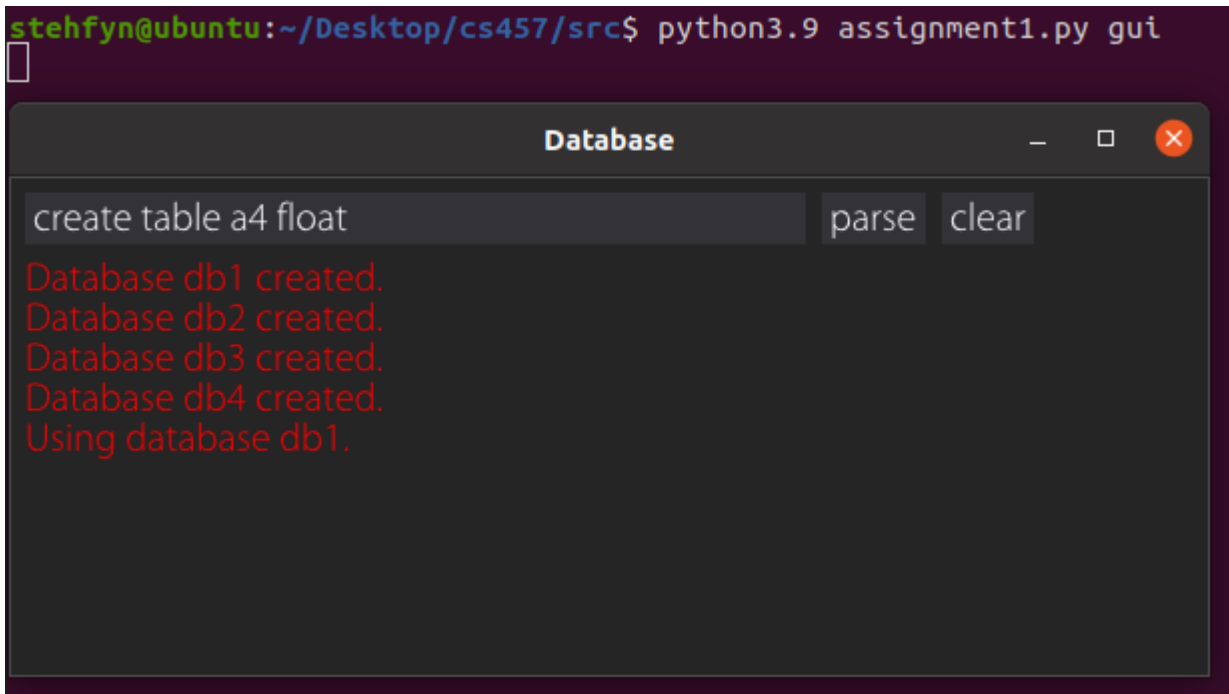
The database project batch processor at present only expects filepaths as command-line arguments, and **does not** read in information from stdin. This behavior is accessed by invoking:

```
stehfyn@ubuntu:~/Desktop/cs457/src$ python3.9 assignment1.py PA1_test.sql  
Database db_1 created.  
!Failed to create database db_1 because it already exists.  
Database db_2 created.  
Database db_2 deleted.  
!Failed to delete database db_2 because it does not exist.  
Database db_2 created.  
Using database db_1.  
Table tbl_1 created.  
!Failed to create table tbl_1 because it already exists.  
Table tbl_1 deleted.  
!Failed to delete tbl_1 because it does not exist.  
Table tbl_1 created.  
a1 int | a2 varchar(20)  
Table tbl_1 modified.  
a1 int | a2 varchar(20) | a3 float  
Table tbl_2 created.  
a3 float | a4 char(20)  
Using database db_2.  
Table tbl_1 created.  
a3 float | a4 char(20)  
stehfyn@ubuntu:~/Desktop/cs457/src$
```

## Graphical User Interface

The GUI can be accessed by invoking:

```
stehfyn@ubuntu:~/Desktop/cs457/src$ python3.9 assignment1.py gui
```



The screenshot shows a terminal window with the command `python3.9 assignment1.py gui` executed. Below the terminal, a graphical user interface window titled "Database" is open. The window has a dark theme and contains a text input field with the text "create table a4 float". To the right of the input field are two buttons labeled "parse" and "clear". Below the input field, there is a list of red text messages: "Database db1 created.", "Database db2 created.", "Database db3 created.", "Database db4 created.", and "Using database db1.".

### Notable Dependencies

The project relies solely on standard python3.9+ libraries to accomplish the required functionality, and two external libraries to accomplish the graphical user interface. The notable libraries are:

- **argparse** - The tokenizer for the database language, aids in implementing the grammar.
- **configparser** - The file reader/writer. This project leverages its simplicity in reading and writing “sections” of a file through a python dictionary-like programming interface, and is wrapped/extended by **database\_impl.py** to implement database reading and writing.
- **dearpygui (external)** - An external graphical user interface library for python, is used by **database.py** to implement a windowed interface to the database functionality.
- **psutil (external)** - An external library to determine the parent process of **assignment1.py**, thus aiding in determining the run mode of the database (interactive, non-interactive, etc.).