Student: Noah Yamanaka

Professor: Taha Bouhsine

Class / Date: CS01104 / 11/13/2024

Title: Student Database with Visualization of Data

Implementation Description:

This project code acts as a database system that can store, retrieve, and visualize the entry (student) data given to it. There are three major files; the StudentDatabase.m and Student.m files contain the class definitions for the StudentDatabase and Student classes, both of which are the major two classes used in the project's code. The other major file, main.m, acts as the main file where the methods and functions contained in all of the other files are called. The other files are where the visualization functions are stored to be called in the main.m file.

When beginning work on the project, the first file created was the "main.m" file so that the StudentDatabase class could be initialized with "db = StudentDatabase();" After the StudentDatabase.m file was created, work on the StudentDatabase class began with its definition. The class's property, an empty array of student objects, was declared along with the methods of the class, which were not completed until after the Student.m file and Student class were created, which was immediately after the empty array of student objects in the StudentDatabase class properties was declared. In the Student class, the student object's properties: ID, Name, Age, GPA, and Major were declared. Along with the properties, the "Student" function to convert the property variables into objects, the "displayInfo" function, and the setter functions allowing for student object data to be changed after being placed into the database were also created. With the Student class fully defined, the StudentDatabase methods "addStudent," "findStudentByID," "getStudentsByMajor," "saveToFile," and "loadFromFile" were then implemented, allowing the project to add new entries, search for students based on given parameters, and save and load the database to and from a .mat file. Now, transitioning to the "main.m" file, the methods implemented in the StudentDatabase and Student classes were tested with five initialized student objects being used. When it comes to the visualizations of the data within the database, three functions were created, with each one created to add a different graph to the same figure. The three different functions and their purposes are explained below. Following the creation of the three different graphing functions, they were then implemented in the "main.m" file using this code:

"subplot(3, 1, 1);
GPADistribution(loadedDB.Students);
subplot(3, 1, 2);
averageGPAByMajor(loadedDB.Students);
subplot(3, 1, 3);
ageDistribution(loadedDB.Students);"
Immediately after the implementation of the three graphs/charts above, the line "saveas(gcf, 'Visualizations.png');" was added, saving the three visualizations above as a .png file titled "Visualizations.png."

Summary:

1. Student Class:

Defines the structure of a student, including properties such as ID, Name, Age, GPA, and Major.

Contains methods such as “displayInfo” and setters, which, when called, can display a student’s data or modify the data of a student, ranging from the student’s ID to their GPA.

2. StudentDatabase Class:

Acts as a container for storing student objects while also containing methods for adding or searching for students via their ID or major.

3. Main Script (main.m):

Initializes the student database and adds five example students.

Demonstrates how the use of “findStudentByID” and “findStudentByMajor” methods, which are methods used to filter and search data who are stored in the StudentDatabase.m file, can be called in the main.m file.

Saves the visualizations as a .png file.

4. Visualization Functions:

GPADistribution: Creates a histogram of student GPAs, showing the distribution of all the students’ GPAs.

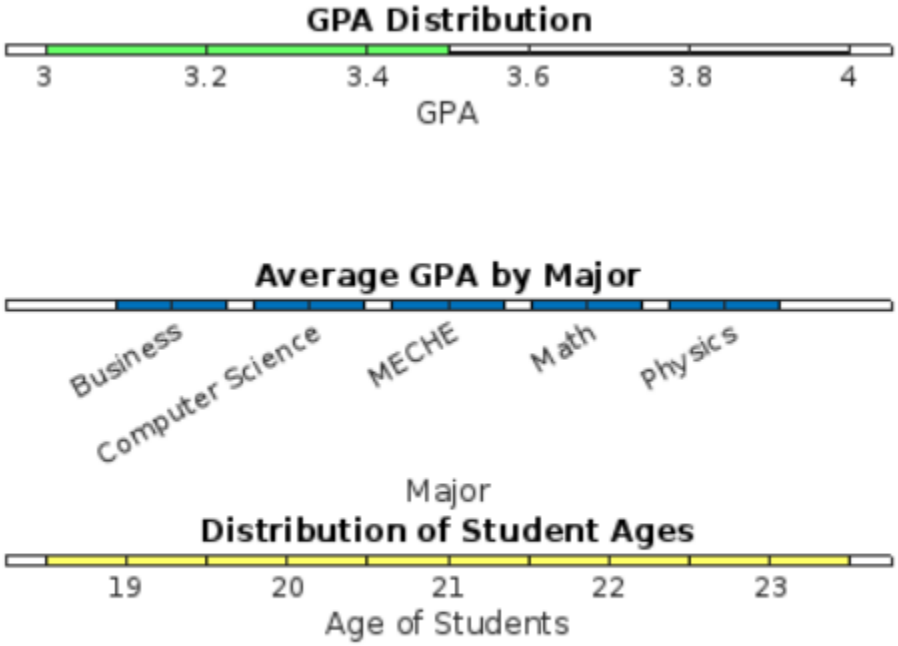
averageGPAByMajor: Plots the average GPA per major as a bar chart.

ageDistribution: Shows the age distribution among the student population.

Instructions For Running:

1. Install and set up MATLAB.
2. Go to the directory containing the project files and run the main.m file.
3. The database will initialize, add the example students, and demonstrate implemented features, such as the search and filter operations.
4. The visualizations will then be displayed and saved as “Visualizations.png” in the same directory.

Visualizations:



The bolded titles indicate which graph is which, with the axis’ being labeled by the grayed text under the graphs. They are also color coded, with green being "GPADistribution," blue being "averageGPAByMajor," and yellow being "ageDistribution." It should also be noted that they appear as one figure in the output, but the zoom and visual ranges for all three can be altered separately from the other.

Implemented Features:

1. Student and Database Classes
    a. Student Class: Stores individual student data with attributes of ID, name, age, GPA, and major of the student.
    b. StudentDatabase Class: Manages the collection of student objects with methods to filter, search, save, and load student data.
2. File Saving

    a. The student data is saved as a .mat file with the “saveToFile” method and can be loaded with the “loadFromFile” method.
3. Search/Filter
    a. “findStudentByID” method is used to identify and retrieve a student’s data by the unique ID number, which each student object is assigned upon initialization.
    b. “getStudentsByMajor” method is used to filter out students who have the same major as the one specified when the method was called.
4. Data Visualizations
    a. GPADistribution: Plots a histogram to show the distribution of the students’ GPAs
    b. averageGPAByMajor: Depicts every major’s average GPA as a bar chart.
    c. ageDistribution: Displays the age distribution among the student population.
    d. Saving Feature
        i. All of the generated plots are saved as a .png file titled “Visualizations.png”.