**Programming Assignment Unit 3**

**University of the People**

**CS 1102-01 - AY2024-T5: Programming 1**

**Avernell Christopher Melikhov**

**July 11th, 2024**

/\*\* This program manages student information.

\* Users can add new students by entering their name, ID, age, and grade, which are stored in a list.

\* The program allows viewing the total number of students and displaying all stored student details if any are added.

\* It also supports updating student information by searching for a student ID and modifying their name, age, and grade if found.

\* Error messages handle cases where users input invalid choices or IDs that do not exist in the student list.

\* \*/

**package** application;

**import** textio.TextIO;

**import** java.util.ArrayList;

**public** **class** StudentManager {

**public** **static** **int** *totalNumberOfStudents* = 0; // static variable for the total number of students

**public** **static** ArrayList<StudentDetails> *studentList* = **new** ArrayList<>(); // initializing list variable to store student details

**public** **static** **void** main(String[] args) {

*displayMenu*();

}

// Display the main menu and handle user input

**public** **static** **void** displayMenu() {

System.***out***.println(" Welcome to the Student Manager");//Welcome message to user

System.***out***.println();

**try** {

**while** (**true**) {

System.***out***.println();

System.***out***.println("This is the main menu:");

System.***out***.println("1. Add a new student");

System.***out***.println("2. View total number of students");

System.***out***.println("3. View all student details");

System.***out***.println("4. Update student details");

System.***out***.println("5. Exit");

System.***out***.print("Choose a number to continue: ");

**int** displayMenu = TextIO.*getlnInt*(); // Read user input as integer

// The if statement handles errors if an incorrect number is added

**if** (displayMenu < 1 || displayMenu > 5) {

System.***out***.println("\nYour input is invalid. Please choose a number between 1-5!");

**continue**;

}

// Switch case conditional to handle user's choice

**switch** (displayMenu) {

**case** 1:

*addStudent*(); // lets the user add a student

**break**;

**case** 2:

System.***out***.println("Total number of students: " + *getTotalNumberOfStudents*() + "\n"); // tells the user the total number of students

**break**;

**case** 3:

*viewStudentDetails*(); // method to view the students

**break**;

**case** 4:

*updateStudent*(); // method to update the student

**break**;

**case** 5:

System.***out***.println("Have a great day! Exiting..."); // exit the switch case conditional

**return**;

**default**:

System.***out***.println("Invalid choice! Please choose a number between 1-5!"); // catching errors

}

}

} **catch** (Exception e) { // Handles other Errors

System.***out***.println("Warning! Error: " + e.getMessage()); // getMessage() prints the description of the exception.

}//end of try/catch block

}//end displayMenu

//\*\* FUNCTION 1 - TO ADD STUDENT DETAILS \*\*//

//Allows the user to input information to add students

**public** **static** **void** addStudent() {

System.***out***.println("You can input Student Details");

System.***out***.println();

System.***out***.println("Enter student name: ");

String name = TextIO.*getln*();

System.***out***.println("Enter student ID: ");

**int** id = TextIO.*getInt*();

System.***out***.println("Enter student age: ");

**int** age = TextIO.*getInt*();

System.***out***.println("Enter student grade: ");

**int** grade = TextIO.*getInt*();

StudentDetails newStudent = **new** StudentDetails(name, id, age, grade);

*studentList*.add(newStudent); //add students to the list

*totalNumberOfStudents*++;// incrementing the total number of students

System.***out***.println("Student added!");

}// end student details method

//\*\* FUNCTION 2 - To VIEW STUDENT DETAILS \*\*//

//Allows the user to view the students

**public** **static** **void** viewStudentDetails() {

**if** (*studentList*.isEmpty()) {

System.***out***.println("No students have been added!");// What is displayed if no students are in the list

} **else** {

**for** (**int** i = 0; i < *studentList*.size(); i++) { // if they are in the list a for loop iterates through the list and displays the students

StudentDetails student = *studentList*.get(i);

System.***out***.println("Name: " + student.getName());

System.***out***.println("ID: " + student.getId());

System.***out***.println("Age: " + student.getAge());

System.***out***.println("Grade: " + student.getGrade());

System.***out***.println("-----");

}

}

}// end viewStudentDetails

//\*\* FUNCTION 3 TO UPDATE STUDENT \*\*//

//Allows the user to update the students in a list

**public** **static** **void** updateStudent() {

System.***out***.println();

System.***out***.print("Enter the ID of the student to update: ");// finding the student with a corresponding id

**int** id = TextIO.*getInt*();

StudentDetails student = **null**; // initializing student variable and setting to null as per Java instructions

**for** (**int** i = 0; i < *studentList*.size(); i++) {

StudentDetails s = *studentList*.get(i);// assigning variable s to an element in studentList

**if** (s.getId() == id) { // if a corresponding id is found it s is assigned to the variable student

student = s;

**break**;

}

}

**if** (student == **null**) {

System.***out***.println("Student with ID " + id + " not found.");//if student variable does not have a value this message is displayed

} **else** {

// Updating student details with set method which update the state data of an object

System.***out***.println("Updating details for student with ID: " + id);

System.***out***.print("Enter new name (current: " + student.getName() + "): ");

String newName = TextIO.*getln*();

student.setName(newName);

System.***out***.print("Enter new age (current: " + student.getAge() + "): ");

**int** newAge = TextIO.*getInt*();

student.setAge(newAge);

System.***out***.print("Enter new grade (current: " + student.getGrade() + "): ");

**int** newGrade = TextIO.*getInt*();

student.setGrade(newGrade);

System.***out***.println("Student details updated!");

}

}//end of Function 3

//\*\* METHOD TOTAL NUMBER OF STUDENTS \*\*//

//Static method to get the total number of students

**public** **static** **int** getTotalNumberOfStudents() {

**return** *totalNumberOfStudents*;

}// end get total number of students

//\*\*ADD STUDENT DATA \*\*//

**static** **class** StudentDetails {

// Variables

**private** String name; // name of the student

**private** **int** id; // if of student

**private** **int** age; // age of student

**private** **int** grade;

// Method to add student data

**public** StudentDetails(String name, **int** id, **int** age, **int** grade) {

**this**.name = name;

**this**.id = id;

**this**.age = age;

**this**.grade = grade;

}//end Student details class

// Get student Details methods

**public** String getName() {

**return** name;

}

**public** **int** getId() {

**return** id;

}

**public** **int** getAge() {

**return** age;

}

**public** **int** getGrade() {

**return** grade;

}

// Set student details methods

**public** **void** setName(String name) {

**this**.name = name;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

**public** **void** setGrade(**int** grade) {

**this**.grade = grade;

}

}

}//end class StudentManager

**References**

Eck, D. J. (2022). *Introduction to programming using java version 9, JavaFX edition.* Licensed under CC 4.0.

<https://math.hws.edu/javanotes/>

GeeksforGeeks. (2021, June 21). *Iterate List in Java using Loops*. GeeksforGeeks. <https://www.geeksforgeeks.org/iterate-list-in-java-using-loops/>

OpenAI. (2024). Explanation of setter methods in Java code. Retrieved from [ChatGPT](https://chat.openai.com/).

*Updating student information (Runtime)*. (n.d.). Stack Overflow. <https://stackoverflow.com/questions/20837053/updating-student-information-runtime>