Student Management System

**Code:**

#include <iostream>

#include <string>

using namespace std;

// Structure to represent a single student

class Student {

public:

int id;

string name;

int age;

string grade;

};

// Array and counter to store students

Student students[100];

int student\_count = 0;

// Function to add a new student

void addStudent() {

if (student\_count >= 100) {

cout << "Cannot add more students. Maximum limit reached.\n";

return;

}

cout << "Enter ID: ";

cin >> students[student\_count].id;

cin.ignore();

cout << "Enter Name: ";

getline(cin, students[student\_count].name);

cout << "Enter Age: ";

cin >> students[student\_count].age;

cin.ignore();

cout << "Enter Grade: ";

getline(cin, students[student\_count].grade);

student\_count++;

cout << "Student added successfully!\n";

}

// Function to update a student's details

void updateStudent() {

int id;

cout << "Enter the ID of the student to update: ";

cin >> id;

for (int i = 0; i < student\_count; i++) {

if (students[i].id == id) {

cin.ignore(); // Clear input buffer

cout << "Enter new Name: ";

getline(cin, students[i].name);

cout << "Enter new Age: ";

cin >> students[i].age;

cin.ignore();

cout << "Enter new Grade: ";

getline(cin, students[i].grade);

cout << "Student details updated successfully!\n";

return;

}

}

cout << "Student with ID " << id << " not found.\n";

}

// Function to display all students

void displayStudents() {

if (student\_count == 0) {

cout << "No students to display.\n";

return;

}

cout << "\n--- Student List ---\n";

for (int i = 0; i < student\_count; i++) {

cout << "ID: " << students[i].id<<endl;

cout << " Name: " << students[i].name<<endl;

cout << " Age: " << students[i].age<<endl;

cout << " Grade: " << students[i].grade << endl;

}

}

// Main function

int main() {

int choice;

do {

cout << "\n--- Student Management System ---\n";

cout << "1. Add Student\n";

cout << "2. Update Student\n";

cout << "3. Display All Students\n";

cout << "4. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

addStudent();

break;

case 2:

updateStudent();

break;

case 3:

displayStudents();

break;

case 4:

cout << "Exiting...\n";

break;

default:

cout << "Invalid choice. Please try again.\n";

}

} while (choice != 4);

return 0;

}