using System;

using System.Collections.Generic;

using System.IO;

internal class Teacher

{

public int ID { get; set; }

public string Name { get; set; }

public string Class { get; set; }

public string Section { get; set; }

public override string ToString()

{

return $"{ID},{Name},{Class},{Section}";

}

}

namespace TeacherRecordSystem

{

class Program

{

static void Main(string[] args)

{

bool running = true;

while (running)

{

Console.WriteLine("\nTeacher Record System");

Console.WriteLine("1. Add Teacher");

Console.WriteLine("2. Display Teachers");

Console.WriteLine("3. Update Teacher");

Console.WriteLine("4. Exit");

Console.Write("Select an option: ");

int choice = Convert.ToInt32(Console.ReadLine());

switch (choice)

{

case 1:

AddTeacher();

break;

case 2:

DisplayTeachers();

break;

case 3:

UpdateTeacher();

break;

case 4:

running = false;

break;

}

}

}

private static void AddTeacher()

{

Console.Write("Enter Teacher ID: ");

int id = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Teacher Name: ");

string name = Console.ReadLine();

Console.Write("Enter Class and Section: ");

string classAndSection = Console.ReadLine();

Teacher teacher = new Teacher(id, name, classAndSection);

FileOperations.AddTeacher(teacher);

}

private static void DisplayTeachers()

{

List<Teacher> teachers = FileOperations.ReadTeachers();

foreach (var teacher in teachers)

{

Console.WriteLine($"{teacher.ID}, {teacher.Name}, {teacher.ClassAndSection}");

}

}

private static void UpdateTeacher()

{

Console.Write("Enter the ID of the teacher to update: ");

int id = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Updated Teacher Name: ");

string name = Console.ReadLine();

Console.Write("Enter Updated Class and Section: ");

string classAndSection = Console.ReadLine();

Teacher updatedTeacher = new Teacher(id, name, classAndSection);

FileOperations.UpdateTeacher(id, updatedTeacher);

}

}

public class Teacher

{

public int ID { get; set; }

public string Name { get; set; }

public string ClassAndSection { get; set; }

public Teacher(int id, string name, string classAndSection)

{

ID = id;

Name = name;

ClassAndSection = classAndSection;

}

public override string ToString()

{

return $"{ID},{Name},{ClassAndSection}";

}

}

public static class FileOperations

{

private static string filePath = "teachers.txt";

public static void AddTeacher(Teacher teacher)

{

using (StreamWriter sw = File.AppendText(filePath))

{

sw.WriteLine(teacher.ToString());

}

}

public static List<Teacher> ReadTeachers()

{

List<Teacher> teachers = new List<Teacher>();

if (File.Exists(filePath))

{

using (StreamReader sr = new StreamReader(filePath))

{

string line;

while ((line = sr.ReadLine()) != null)

{

var data = line.Split(',');

teachers.Add(new Teacher(int.Parse(data[0]), data[1], data[2]));

}

}

}

return teachers;

}

public static void UpdateTeacher(int id, Teacher updatedTeacher)

{

List<Teacher> teachers = ReadTeachers();

using (StreamWriter sw = new StreamWriter(filePath))

{

foreach (var teacher in teachers)

{

if (teacher.ID == id)

sw.WriteLine(updatedTeacher.ToString());

else

sw.WriteLine(teacher.ToString());

}

}

}

}

}