**Source code**

using System;

using System.Collections.Generic;

using System.Text;

using System.IO;

namespace ConsoleApp2

{

class teacher\_records

{

public int id { get; set; }

public string name { get; set; }

public int cls { get; set; }

public char sec { get; set; }

}

class assignment\_1\_teacher\_records

{

static public void add\_teachers(teacher\_records new\_record, List<teacher\_records> teacher\_list)

{

Console.WriteLine("\nEnter the number of records you want to enter..");

int record\_count = int.Parse(Console.ReadLine());

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

{

Console.WriteLine("\nEnter the teacher's ID :");

int ID = int.Parse(Console.ReadLine());

Console.WriteLine("Enter the teacher's name :");

string NM = Console.ReadLine();

Console.WriteLine("Enter Class :");

int clss = int.Parse(Console.ReadLine()); // reading class from user input

Console.WriteLine("Enter Section :");

char secn = Console.ReadLine()[0]; // reading signle char from user

new\_record = new teacher\_records();

new\_record.id = ID;

new\_record.name = NM;

new\_record.cls = clss;

new\_record.sec = secn;

teacher\_list.Add(new\_record);

}

}

static public void display\_teachers(teacher\_records new\_record, List<teacher\_records> teacher\_list)

{

for (int i = 0; i < teacher\_list.Count; i++)

{

Console.WriteLine("\n\nID = " + teacher\_list[i].id);

Console.WriteLine("Teacher Name = " + teacher\_list[i].name);

Console.WriteLine("Class = " + teacher\_list[i].cls);

Console.WriteLine("Section = " + teacher\_list[i].sec + "\n\n");

}

if (teacher\_list.Count < 1)

Console.WriteLine("\n Alert.., No teacher's data found \n");

}

static public void dump\_data\_to\_text\_file(List<teacher\_records> teacher\_list)

{

string path\_to\_file = @"E:\Assignment 1\teacher\_records.txt";

string rec = "";

for (int i = 0; i < teacher\_list.Count; i++)

{

rec += teacher\_list[i].id + " " + teacher\_list[i].name + " " + teacher\_list[i].cls + " " + teacher\_list[i].sec + "\n";

}

File.WriteAllText(path\_to\_file, rec);

}

static public void convert\_text\_data\_to\_list(teacher\_records new\_record, List<teacher\_records> teacher\_list)

{

string path\_to\_file = @"E:\Assignment 1\teacher\_records.txt";

if (File.Exists(path\_to\_file))

{

string[] lines = File.ReadAllLines(path\_to\_file);

String[] text\_file\_data;

for (int i = 0; i < lines.Length; i++)

{

text\_file\_data = lines[i].Split(' ');

new\_record = new teacher\_records();

new\_record.id = int.Parse(text\_file\_data[0]);

new\_record.name = text\_file\_data[1];

new\_record.cls = int.Parse(text\_file\_data[2]);

new\_record.sec = text\_file\_data[3][0];

teacher\_list.Add(new\_record);

}

}

else

Console.WriteLine("\nFile not found at location, A new file will be created\n");

}

static public void update\_teachers\_data(List<teacher\_records> teacher\_list)

{

Console.WriteLine("Enter the teacher's id to update data ..");

int update\_id = int.Parse(Console.ReadLine());

int index = teacher\_list.FindIndex(x => x.id == update\_id);

if (index > -1)

{

Console.WriteLine("Enter choice \n1. Name \n2. Class\n3. Section");

int choice = int.Parse(Console.ReadLine());

if (choice == 1)

{

Console.WriteLine("Enter new name ..");

string nn = Console.ReadLine();

teacher\_list[index].name = nn;

}

if (choice == 2)

{

Console.WriteLine("Enter new class allocated ..");

int nc = int.Parse(Console.ReadLine());

teacher\_list[index].cls = nc;

}

if (choice == 3)

{

Console.WriteLine("Enter new section allocated ..");

char ns = Console.ReadLine()[0];

teacher\_list[index].sec = ns;

}

}

else

Console.WriteLine("\nIndex not found in records");

}

static public void delete\_records(List<teacher\_records> teacher\_list)

{

Console.WriteLine("\nEnter the teacher's id to delete record ..");

int delete\_id = int.Parse(Console.ReadLine());

int index = teacher\_list.FindIndex(x => x.id == delete\_id);

if (index > -1)

{

Console.WriteLine("\nTeacher's Name :" + teacher\_list[index].name + "\nAllocated Class :" + teacher\_list[index].cls + "\nAllocated Section :" + teacher\_list[index].sec);

Console.WriteLine("Will be deleted from records ..\n");

teacher\_list.RemoveAt(index);

}

else

Console.WriteLine("ID not found in records \n");

}

static void Main(string[] args)

{

/\* We will create a list of teachers type each index of this list will hold an object of class teacher,

each instances will hold the records of teacher (id,name,class,section)

\*/

// [obj1] [obj2] [obj3]

List<teacher\_records> teacher\_list = new List<teacher\_records>();

// instantiating class teacher\_record

teacher\_records new\_record = null;

convert\_text\_data\_to\_list(new\_record, teacher\_list);

while (true)

{

Console.WriteLine("Enter Choice \n 1. Add a record \n 2. Display record \n 3. Save data to text file and Exit \n 4. Update data\n 5. Delete record\n 6. Exit without saving");

// try

//{

int choice = int.Parse(Console.ReadLine());

switch (choice)

{

case 1:

add\_teachers(new\_record, teacher\_list);

break;

case 2:

display\_teachers(new\_record, teacher\_list);

break;

case 3:

dump\_data\_to\_text\_file(teacher\_list);

break;

case 4:

update\_teachers\_data(teacher\_list);

break;

case 5:

delete\_records(teacher\_list);

Console.WriteLine("Under Dev");

break;

case 6:

break;

}

if (choice == 3 || choice == 6) // breaking out of the while loop

break;

// }

// catch

//{

// Console.WriteLine("Please enter a numerical value .. ");

// }

}

}

}

}