```
using System;
using System.Collections.Generic;
using System.IO;

namespace Phaseproject
{
   internal class Program
   {
      static void Main(string[] args)
      {

         using (FileStream fs = new FileStream(@"C:\project\Teacher.txt", FileMode.Open))
         using (StreamReader sr = new StreamReader(fs))
         {
            string content = sr.ReadToEnd();
            string[] lines = content.Split(new string[] { Environment.NewLine },
StringSplitOptions.RemoveEmptyEntries);

            List<Teacher> listTeachers = new List<Teacher>();
            foreach (string line in lines)
            {
               string[] column = line.Split(',');

                  Teacher teacher = new Teacher();
                  teacher.Id = column[0];
                  teacher.FirstName = column[1];
                  teacher.LastName = column[2];
                  teacher.CClass = column[3];
                  teacher.Section = column[4];
```

```
            listTeachers.Add(teacher);
        }
        Console.WriteLine(content);
    }
    Console.WriteLine("1.create\n2.update\n3.delete\n4.search\n5.display");
    static void update()
    {
        List<Teacher> listTeachers = new List<Teacher>();
        string teacherfile = "C:\\project\\Teacher.txt";
        string[] arrteacher = System.IO.File.ReadAllLines(teacherfile);

        foreach (string line in arrteacher)
        {
            string[] l = line.Split(',');
            Teacher teacher = new Teacher();
            teacher.Id = l[0];
            teacher.FirstName = l[1];
            teacher.LastName = l[2];
            teacher.CClass = l[3];
            teacher.Section = l[4];
            listTeachers.Add(teacher);

        }
        string id;
        Console.WriteLine("Enter the id you want to update:");
        id = Console.ReadLine();
        foreach (Teacher t in listTeachers)
        {
            if (t.Id == id)
```

```
    {
        Console.WriteLine("enter first name:");
        string ufirstname=Console.ReadLine();
        Console.WriteLine("enter last name:");
        string ulastname=Console.ReadLine();
        Console.WriteLine("enter class:");
        string uclass=Console.ReadLine();
        Console.WriteLine("enter section:");
        string usection = Console.ReadLine();
        t.FirstName = ufirstname;
        t.LastName = ulastname;
        t.CClass = uclass;
        t.Section = usection;
        Console.WriteLine("updated one is:");
        Console.WriteLine($"{ t.Id},{ t.FirstName},{ t.LastName},{ t.CClass},{ t.Section}");


        break;

    }

}
int count = 0;
string[] arr = new string[listTeachers.Count];
foreach (Teacher t1 in listTeachers)
{
    string s = ($"{t1.Id},{t1.FirstName},{t1.LastName},{t1.CClass},{t1.Section}");
    arr[count] = s;
    count++;
```

```
    }
    File.WriteAllLines(@"C:\project\Teacher.txt", arr);



}
static void create()
{
    List<Teacher> listTeachers = new List<Teacher>();
    string teacherfile = "C:\\project\\Teacher.txt";
    string[] arrteacher = System.IO.File.ReadAllLines(teacherfile);

    foreach (string line in arrteacher)
    {
        string[] l = line.Split(',');
        Teacher teacher = new Teacher();
        teacher.Id = l[0];
        teacher.FirstName = l[1];
        teacher.LastName = l[2];
        teacher.CClass = l[3];
        teacher.Section = l[4];
        listTeachers.Add(teacher);

    }
    string UIId = "";
    string UIFirstName = "";
    string UILastName = "";
    string UIClass = "";
    string UIsection = "";
```

```
using (FileStream fs = new FileStream(@"C:\project\Teacher.txt", FileMode.Append))
using (StreamWriter sw = new StreamWriter(fs))
{
   Teacher teacher = new Teacher();
   teacher.Id = UIId;
   teacher.FirstName = UIFirstName;
   teacher.LastName = UILastName;
   teacher.CClass = UIClass;
   teacher.Section = UIsection;

   Console.WriteLine(" enter additional data to create?");



   Console.WriteLine("Please enter the teacher id: ");
   UIId = Console.ReadLine();
   Console.WriteLine("Please enter the teacher firstname: ");
   UIFirstName = Console.ReadLine();
   Console.WriteLine("Please enter the teacher Lastname: ");
   UILastName = Console.ReadLine();
   Console.WriteLine("Please enter the teacher class: ");
   UIClass = Console.ReadLine();
   Console.WriteLine("Please enter the section: ");
   UIsection = Console.ReadLine();
   string fullText = (UIId + "," + UIFirstName + "," + UILastName +"," + UIClass + "," + UIsection);
   sw.WriteLine(fullText);
   int count = 0;
   string[] arr = new string[listTeachers.Count];
  /* foreach (Teacher t1 in listTeachers)
   {
```

```
            string s = ($"{t1.Id},{t1.FirstName},{t1.LastName},{t1.CClass},{t1.Section}");
            arr[count] = s;
            count++;

        }
        File.WriteAllLines(@"C:\project\Teacher.txt", arr);*/
    }
}
static void delete()
{
   List<Teacher> listTeachers = new List<Teacher>();
    string teacherfile = "C:\\project\\Teacher.txt";
    string[] arrteacher = System.IO.File.ReadAllLines(teacherfile);

    foreach (string line in arrteacher)
    {
        string[] l = line.Split(',');
        Teacher teacher = new Teacher();
        teacher.Id = l[0];
        teacher.FirstName = l[1];
        teacher.LastName = l[2];
        teacher.CClass = l[3];
        teacher.Section = l[4];
        listTeachers.Add(teacher);

    }
    string id;
    Console.WriteLine("Enter the id to delete:");
    id=Console.ReadLine();
```

```
  foreach(Teacher t in listTeachers)
   {
      if(t.Id == id)
      {
         listTeachers.Remove(t);
         break;

      }
      else
      {
         Console.WriteLine("enterd id is not there");
      }
   }

   int count = 0;
   string[] arr = new string[listTeachers.Count];
   foreach (Teacher t1 in listTeachers)
   {
      string s = ($"{t1.Id},{t1.FirstName},{t1.LastName},{t1.CClass},{t1.Section}");
      arr[count] = s;
      count++;

   }
   File.WriteAllLines(@"C:\project\Teacher.txt", arr);

}

static void search()
{
```

```
List<Teacher> listTeachers = new List<Teacher>();
string teacherfile = "C:\\project\\Teacher.txt";
string[] arrteacher = System.IO.File.ReadAllLines(teacherfile);

foreach (string line in arrteacher)
{
   string[] l = line.Split(',');
   Teacher teacher = new Teacher();
   teacher.Id = l[0];
   teacher.FirstName = l[1];
   teacher.LastName = l[2];
   teacher.CClass = l[3];
   teacher.Section = l[4];
   listTeachers.Add(teacher);

}
Console.WriteLine("enter id:");
string id = Console.ReadLine();
foreach (Teacher t in listTeachers)
{
   if (t.Id == id)
   {
      Console.WriteLine("given {0} is present in the given file", id);
      Console.WriteLine($"{t.Id},{t.FirstName},{t.LastName},{t.CClass},{t.Section}");
      break;

   }
   else
   {
```

```
            Console.WriteLine("enterd id is not there");
        }
    }

}
static void display()
{

    static void firstname()
    {
        List<Teacher> listTeachers = new List<Teacher>();
         string teacherfile = "C:\\project\\Teacher.txt";
         string[] arrteacher = System.IO.File.ReadAllLines(teacherfile);
         foreach (string line in arrteacher)
         {
            string[] l = line.Split(',');
            Teacher teacher = new Teacher();
            teacher.Id = l[0];
            teacher.FirstName = l[1];
            teacher.LastName = l[2];
            teacher.CClass = l[3];
            teacher.Section = l[4];
            listTeachers.Add(teacher);

         }

        Console.WriteLine("After sorting by First Name:");
        listTeachers.Sort((a, b) => a.FirstName.CompareTo(b.FirstName));
```

```
    foreach (Teacher s in listTeachers)
    {
        Console.WriteLine($"{ s.Id},{ s.FirstName},{ s.LastName},{ s.CClass},{ s.Section}");



    }
}
static void id()
{
    Console.WriteLine("After sorting by Id:");
    List<Teacher> listTeachers = new List<Teacher>();
    string teacherfile = "C:\\project\\Teacher.txt";
    string[] arrteacher = System.IO.File.ReadAllLines(teacherfile);
foreach (string line in arrteacher)
{
    string[] l = line.Split(',');
    Teacher teacher = new Teacher();
    teacher.Id = l[0];
    teacher.FirstName = l[1];
    teacher.LastName = l[2];
    teacher.CClass = l[3];
    teacher.Section = l[4];
    listTeachers.Add(teacher);
}



    listTeachers.Sort((a, b) => a.Id.CompareTo(b.Id));
```

```
        foreach (Teacher s in listTeachers)
        {
            Console.WriteLine($"{ s.Id},{ s.FirstName},{ s.LastName},{ s.CClass},{ s.Section}");



        }

        int count = 0;
        string[] arr = new string[listTeachers.Count];
        foreach (Teacher t1 in listTeachers)
        {
            string s = ($"{t1.Id},{t1.FirstName},{t1.LastName},{t1.CClass},{t1.Section}");
            arr[count] = s;
            count++;


        }
        File.WriteAllLines(@"C:\project\Teacher.txt", arr);



    }
    firstname();
    id();

}
    while (true)
{
    int option;
    Console.WriteLine("Enter Option You want to perform: ");
    option=Convert.ToInt32(Console.ReadLine());
```

```
            switch (option)
               {
                  case 1:
                     create();
                     break;
                  case 2:
                     update();
                  break;
                  case 3:
                     delete();
                     break;
                  case 4:
                     search();
                     break;
                  case 5:
                     display();
                     break;
                  default:
                     Console.WriteLine("Invalid option");
                     break;
               }
            }
         }
      }
   }
```