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

namespace ConsoleProjektH1
{
    class Program
    {
        static void Main(string[] args)
        {
            MyFunctions mf = new MyFunctions();

            while (true)
            {
                Console.Write(":>");
                string input = Console.ReadLine().ToLower();

                switch (input)
                {
                    case "showall":
                        mf.ShowAll();
                        break;
                    default:
                        if (input.StartsWith("addperson"))
                        {
                            mf.AddPerson(input.Substring("addperson".Length + 1).Replace(" ", ""));
                        }
                        else if (input.StartsWith("deleteperson"))
                        {
                            mf.DeletePerson(input.Substring("deleteperson".Length + 1).Replace(" ", ""));
                        }
                        else if (input.StartsWith("changeperson"))
                        {
                            string[] stringArray = input.Split(' ');
                            mf.ChangePerson(stringArray[1], stringArray[2]);
                        }
                        break;
                }
            }
        }
    }

    class MyFunctions
    {
        // Show the entire list
        public void ShowAll()
        {
            List<string> stringList = AlterNameList();

            foreach (var name in stringList)
            {
                if (name != "")
                {
                    Console.WriteLine(name);
                }
            }
        }
    }
}
```

```csharp
            }
        }

        // Add a person at the end of the list
        public void AddPerson(string b)
        {
            File.AppendAllText(Environment.CurrentDirectory + "\\NameList.txt",
Capitalize(b) + ";");
        }

        // Remove a person with a specific name
        public void DeletePerson(string b)
        {
            List<string> stringList = AlterNameList();

            stringList.Remove(Capitalize(b));

            File.WriteAllText(Environment.CurrentDirectory + "\\NameList.txt",
"");

                foreach (var name in stringList)
                {
                    File.AppendAllText(Environment.CurrentDirectory +
"\\NameList.txt", Capitalize(name) + ";");
                }
        }

        // Change the person with a specific name, to another name
        public void ChangePerson(string b, string c)
        {
            List<string> stringList = AlterNameList();

            string CapitalizedB = Capitalize(b);
            string CapitalizedC = Capitalize(c);

            stringList[stringList.IndexOf(CapitalizedB)] = CapitalizedC;

            File.WriteAllText(Environment.CurrentDirectory + "\\NameList.txt",
"");

                foreach (var name in stringList)
                {
                    File.AppendAllText(Environment.CurrentDirectory +
"\\NameList.txt", Capitalize(name) + ";");
                }
        }

        // Capitalize the first letter in a string / char array
        public string Capitalize(string a)
        {
            char[] newCharArray = a.ToCharArray();
            if (a != "")
            {
                newCharArray[0] = char.ToUpper(a[0]);
            }
            return new string(newCharArray);
        }
```

```csharp
            // Fetch the list of names for alteration
            public List<string> AlterNameList ()
            {
                    string content = File.ReadAllText(Environment.CurrentDirectory +
"\\NameList.txt");
                    List<string> stringList = new List<string>();

                    foreach (var name in content.Split(';'))
                    {
                            if (name != "")
                            {
                                    stringList.Add(Capitalize(name));
                            }
                    }

                    return stringList;
            }
        }
}
```