```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ConAppStoreUpdateInTextFile
{
    public class Teacher
    {
        public int ID { get; set; }
        public string Name { get; set; }
        public string ClassSection { get; set; }
        public Teacher(int id, string name, string classSection)
        {
            ID = id;
            Name = name;
            ClassSection = classSection;
        }
    }
}
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.IO;

namespace ConAppStoreUpdateInTextFile
{
    public class TeacherManager
    {
        private string filePath;
        public TeacherManager(string path)
        {
            filePath = path;
        }
        public void AddTeacher(Teacher teacher)
        {
            using(StreamWriter writer = File.AppendText(filePath))
            {
                writer.WriteLine($"{teacher.ID}, {teacher.Name},
{teacher.ClassSection}");
            }
        }
        public void UpdateTeacher(int id, Teacher updatedTeacher)
        {
            string[] lines = File.ReadAllLines(filePath);
            using(StreamWriter writer = new StreamWriter(filePath))
            {
                foreach(string line in lines)
                {
                    string[] parts = line.Split(',');
                    if (int.Parse(parts[0]) == id)
                    {
```

```
                        writer.WriteLine($"{updatedTeacher.ID},
{updatedTeacher.Name}, {updatedTeacher.ClassSection}");
                    }
                    else
                    {
                        writer.WriteLine(line);
                    }
                }
            }
        }
        public void DisplayTeachers()
        {
            Console.WriteLine("Teacher Records: ");
            string[] lines = File.ReadAllLines(filePath);
            foreach(string line in lines)
            {
                string[] parts = line.Split(',');
                Console.WriteLine($"ID: {parts[0]}, Name: {parts[1]}, Class Section:
{parts[2]}");
            }
        }
    }
}


using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.IO;

namespace ConAppStoreUpdateInTextFile
{
    internal class Program
    {
        static void Main(string[] args)
        {
            string filePath = @"D:\Mphasis\Live Session\SimpliLearn Projects\Phase 1
Final Project\Storing and Updating Teacher
Records\ConAppStoreUpdateInTextFile\TeacherRecord.txt";
            TeacherManager teacherManager = new TeacherManager(filePath);
            bool isRunning = true;
            while (isRunning)
            {
                Console.WriteLine("Select an option");
                Console.WriteLine("1. Add Teacher");
                Console.WriteLine("2. Update Teacher");
                Console.WriteLine("3. Display Teachers");
                Console.WriteLine("4. Exit");
                Console.Write("Enter your choice: ");
                int choice = int.Parse(Console.ReadLine());
                switch (choice)
                {
                    case 1:
                        Console.Write("Enter Teacher ID: ");
                        int id = int.Parse(Console.ReadLine());
                        Console.Write("Enter Teacher Name: ");
```

```
                        string name = Console.ReadLine();
                        Console.Write("Enter Class and Section: ");
                        string classSection = Console.ReadLine();

                        teacherManager.AddTeacher(new Teacher(id, name,
classSection));
                        Console.WriteLine("Teacher Added");
                        break;
                    case 2:
                        Console.Write("Enter the Teacher ID to be updated: ");
                        int updateId = int.Parse(Console.ReadLine());
                        Console.Write("Enter New Name: ");
                        string newName = Console.ReadLine();
                        Console.Write("Enter New Class and Section: ");
                        string newClassSection = Console.ReadLine();

                        teacherManager.UpdateTeacher(updateId, new Teacher(updateId,
newName, newClassSection));
                        Console.WriteLine("Teacher updated.");
                        break;
                    case 3:
                        teacherManager.DisplayTeachers();
                        break;
                    case 4:
                        isRunning = false;
                        Console.WriteLine("Exiting the program.");
                        break;

                    default:
                        Console.WriteLine("Invalid choice. Please select a valid
option.");
                        break;
                }
            }
        }
    }
}
```