Date : 22-08-2024

1.

Code :

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

namespace LINQ_Assignment
{

    class Student
    {
        public int StudentId { get; set; }
        public string Name { get; set; }
    }

    class Course
    {
        public int CourseId { get; set; }
        public string Title { get; set; }
    }

    class Enrollment
    {
        public int StudentId { get; set; }
        public int CourseId { get; set; }
    }

    class Program
    {
        static void Main(string[] args)
        {
```

```
var students = new List<Student>
{
 new Student { StudentId = 1, Name = "Alice" },
 new Student { StudentId = 2, Name = "Bob" },
 new Student { StudentId = 3, Name = "Charlie" },
 new Student { StudentId = 4, Name = "David" }
};
var courses = new List<Course>
{
 new Course { CourseId = 1, Title = "Math" },
 new Course { CourseId = 2, Title = "Science" },
 new Course { CourseId = 3, Title = "History" }
};
var enrollments = new List<Enrollment>
{
 new Enrollment { StudentId = 1, CourseId = 1 },
 new Enrollment { StudentId = 1, CourseId = 2 },
 new Enrollment { StudentId = 2, CourseId = 2 },
 new Enrollment { StudentId = 2, CourseId = 3 },
 new Enrollment { StudentId = 3, CourseId = 1 },
 new Enrollment { StudentId = 4, CourseId = 2 }
};

var res1 = students
 .Where(s => enrollments.Count(e => e.StudentId == s.StudentId) >= 2)
 .ToList();

Console.WriteLine("List of students enrolled in at least two courses:");
foreach (var student in res1)
{
    Console.WriteLine(student.Name);
}

var res2 = enrollments
.GroupBy(e => e.StudentId)
.Select(g => new { StudentId = g.Key, CourseCount = g.Count() })
```

```
        .Join(students, g => g.StudentId, s => s.StudentId, (g, s) => new { s.Name,
g.CourseCount })
        .GroupBy(s => s.CourseCount)
        .ToList();

        Console.WriteLine("\nStudents grouped by the number of courses they are
enrolled in:");
        foreach (var group in res2)
        {
            Console.WriteLine($"{group.Key} Course: {string.Join(", ", group.Select(s =>
s.Name))}");
        }
        var res3 = enrollments
        .GroupBy(e => e.CourseId)
        .Where(g => g.Count() > 1)
        .Select(g => new { CourseId = g.Key, Students = g.Select(e => e.StudentId).ToList()
})
        .Join(courses, g => g.CourseId, c => c.CourseId, (g, c) => new { c.Title, StudentIds =
g.Students })
        .ToList();

        Console.WriteLine("\nCourses with students enrolled in more than one course:");
        foreach (var course in res3)
        {
            var studentNames = students.Where(s =>
course.StudentIds.Contains(s.StudentId)).Select(s => s.Name);
            Console.WriteLine($"Course: {course.Title}, Students: {string.Join(", ",
studentNames)}");
        }

        var res4 =
            (from c in courses
             join e in enrollments on c.CourseId equals e.CourseId into g
             let StudentCount = g.Count()
             orderby StudentCount descending
             select new
```

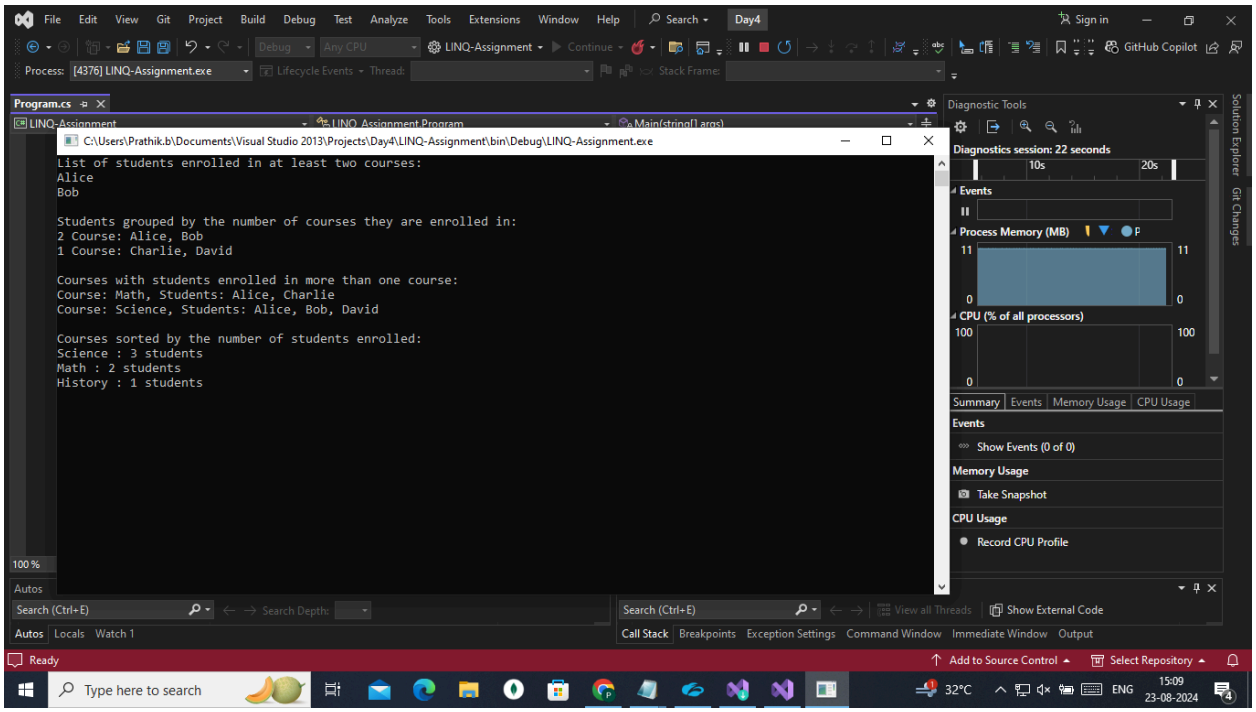
```
            {
                CourseTitle = c.Title,
                StudentCount = StudentCount
            }).ToList();

        Console.WriteLine("\nCourses sorted by the number of students enrolled:");
        foreach (var course in res4)
        {
            Console.Write(course.CourseTitle);
            Console.Write(" : ");
            Console.Write(course.StudentCount);
            Console.WriteLine(" students");
        }

        Console.ReadKey();

    }
  }
}
```

Output :


C:\Users\Prathik.b\Documents\Visual Studio 2013\Projects\Day4\LINQ-Assignment\bin\Debug\LINQ-Assignment.exe
List of students enrolled in at least two courses:
Alice
Bob
Students grouped by the number of courses they are enrolled in:
2 Course: Alice, Bob
1 Course: Charlie, David
Courses with students enrolled in more than one course:
Course: Math, Students: Alice, Charlie
Course: Science, Students: Alice, Bob, David
Courses sorted by the number of students enrolled:
Science : 3 students
Math : 2 students
History : 1 students