

## DAY 4

**DATE:07-08-25**

**NAME: NANCY M**

### 1. Create two threads to read two separate text files

```
namespace Day4taskthread;
class Program
{
    static void Main()
    {
        Thread f1 = new Thread(() =>
ReadFile("C:\\Users\\nancy.m\\source\\repos\\file1.txt"));
        Thread f2 = new Thread(() =>
ReadFile("C:\\Users\\nancy.m\\source\\repos\\file2.txt"));
        f1.Start();
        f2.Start();

        f1.Join();
        f2.Join();

        Console.WriteLine("Finished reading files using Threads.\n");
    }

    static void ReadFile(string fileName)
    {
        string content = File.ReadAllText(fileName);
        string shortName = Path.GetFileName(fileName);
        Console.WriteLine($"Thread reading {shortName}");
        Console.WriteLine(content);
    }
}
```

```
Microsoft Visual Studio Debug Console
Thread reading file1.txt
good
Thread reading file2.txt
mrng
Finished reading files using Threads.

C:\Users\nancy.m\source\repos\Day4taskthread\Day4taskthread\bin\Debug\net8.0\Day4taskthread.exe (process 1
ith code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically c
le when debugging stops.
Press any key to close this window . . .
```

### 1.1 Same two files, try to read using the Task Async await.

```
namespace Day4taskasync;
class Program
{
    static async Task Main()
    {
        Console.WriteLine("started reading");

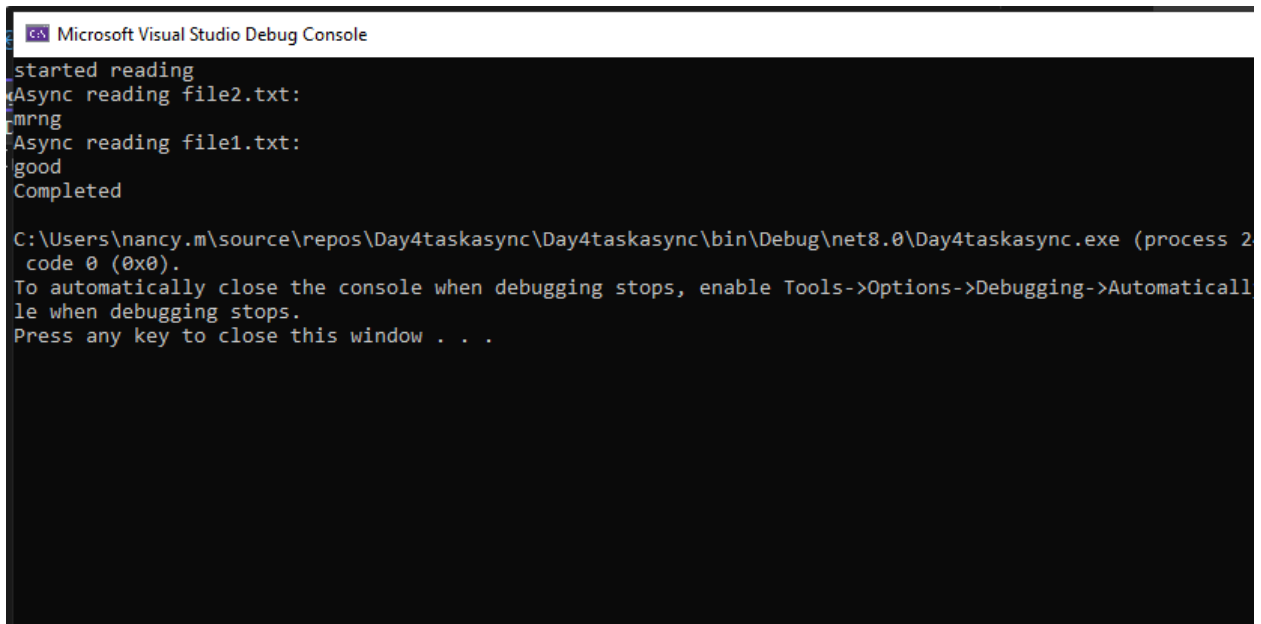
        Task f1 = ReadFileAsync("C:\\Users\\nancy.m\\source\\repos\\file1.txt");
        Task f2 = ReadFileAsync("C:\\Users\\nancy.m\\source\\repos\\file2.txt");

        await Task.WhenAll(f1, f2);

        Console.WriteLine("Completed");
    }

    static async Task ReadFileAsync(string fileName)
    {
        string content = await File.ReadAllTextAsync(fileName);
        string shortName = Path.GetFileName(fileName);
        Console.WriteLine($"Async reading {shortName}:");
        Console.WriteLine(content);
    }
}
```

```
}  
}
```



The screenshot shows the Microsoft Visual Studio Debug Console window. The output text is as follows:

```
started reading  
Async reading file2.txt:  
mrng  
Async reading file1.txt:  
good  
Completed  
  
C:\Users\nancy.m\source\repos\Day4taskasync\Day4taskasync\bin\Debug\net8.0\Day4taskasync.exe (process 2  
code 0 (0x0)).  
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automaticall  
le when debugging stops.  
Press any key to close this window . . .
```

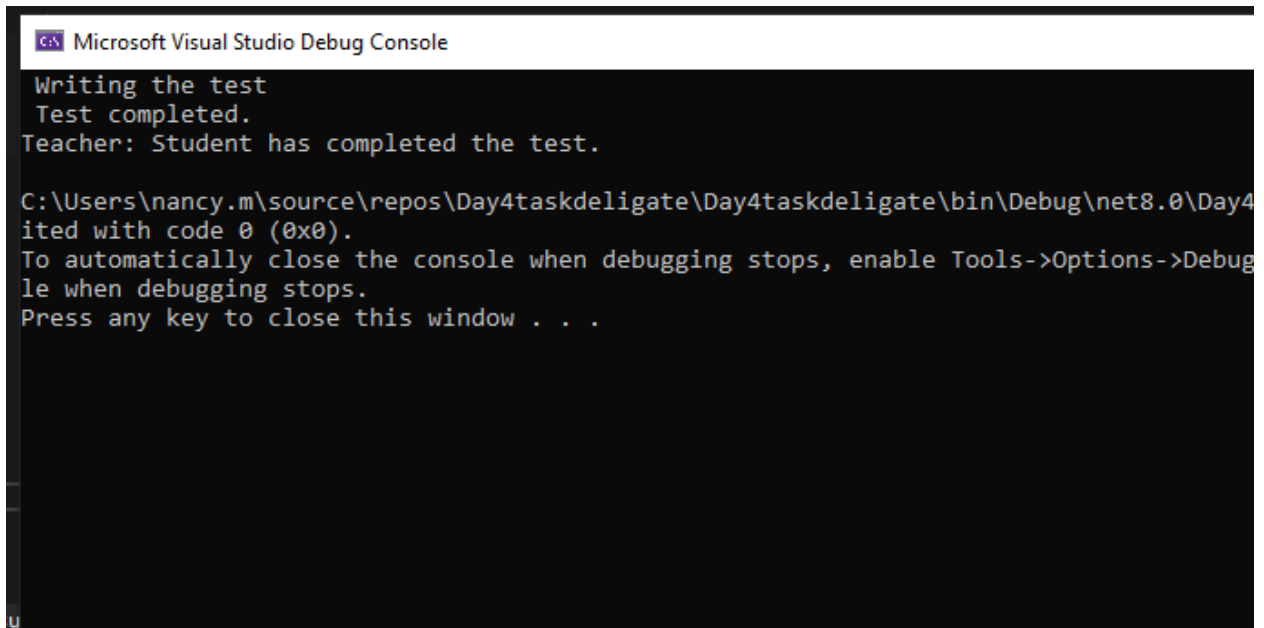
- 2. (Optional) Create delegate use case between teacher class and student class:**
- teacher method should have `test_completed()` method passed as delegate to student
  - student class should have a method `write_test()` which will inturn call the parent delegate.

```
using System;  
namespace Day4Delegate  
{  
    class Program  
    {  
        public delegate void TestDelegate();  
  
        static void Main(string[] args)  
        {  
            Teacher teacher = new Teacher();  
            Student student = new Student();  
            student.WriteTest(teacher.TestCompletednotify);  
        }  
    }  
}
```

```

    }
    class Teacher
    {
        public void TestCompletednotify()
        {
            Console.WriteLine("Teacher: Student has completed the test.");
        }
    }
    class Student
    {
        public void WriteTest(Program.TestDelegate noticeteacher)
        {
            Console.WriteLine(" Writing the test");
            Console.WriteLine(" Test completed.");
            noticeteacher();
        }
    }
}

```



The screenshot shows the Microsoft Visual Studio Debug Console with the following output:

```

Writing the test
Test completed.
Teacher: Student has completed the test.

C:\Users\nancy.m\source\repos\Day4taskdeligate\Day4taskdeligate\bin\Debug\net8.0\Day4
ited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debug
le when debugging stops.
Press any key to close this window . . .

```

The console window is titled "Microsoft Visual Studio Debug Console". The output matches the execution of the provided C# code, showing the messages from the Student class and the Teacher class.