

C# Fundamentals

3. Basic Collections and String Manipulation

Objective: Requirements:

- Write a program to manage a list of strings (e.g., names or tasks).
- Use a List<string> to store items.
- Allow the user to add, remove, and display items.
- Utilize loops and basic string methods (like Trim(), ToUpper()) to process user input.

Program code:

```
using System;

using System.Collections.Generic;

class Program
{
    static void Main()
    {
        List<string> tasks = new List<string>(); // List to store tasks

        while (true)
        {
            Console.WriteLine("\nTask Manager");

            Console.WriteLine("1. Add Task");
            Console.WriteLine("2. Remove Task");
            Console.WriteLine("3. Display Tasks");
            Console.WriteLine("4. Exit");

            Console.Write("Choose an option: ");

            string choice = Console.ReadLine().Trim(); // Trim whitespace

            switch (choice)
            {
                case "1":
```

```

Console.Write("Enter a task: ");

string task = Console.ReadLine().Trim(); // Remove leading/trailing spaces

if (!string.IsNullOrEmpty(task))
{
    tasks.Add(task); // add task to the list of tasks created
    Console.WriteLine($"Task '{task.ToUpper()}' added.");
}
else
{
    Console.WriteLine("Task cannot be empty.");
}

break;

case "2":
    if (tasks.Count == 0)
    {
        Console.WriteLine("No tasks to remove.");
    }
    else
    {
        Console.Write("Enter task to remove: ");

        string removeTask = Console.ReadLine().Trim().ToLower(); // Convert to lowercase

        // COverts the tasks to lowercase to find and remove the selected task
        string taskToRemove = tasks.Find(t => t.ToLower() == removeTask);

        if (taskToRemove != null)
        {
            tasks.Remove(taskToRemove);

            Console.WriteLine($"Task '{taskToRemove.ToUpper()}' removed.");
        }
    }
}

```

```

        else
        {
            Console.WriteLine("Task not found.");
        }

    }

    break;

case "3":
    if (tasks.Count == 0)
    {
        Console.WriteLine("No tasks available.");
    }
    else
    {
        Console.WriteLine("\nYour Tasks:");
        foreach (var t in tasks)
        {
            Console.WriteLine($"{t.ToUpper()}"); // Display tasks in uppercase
        }
    }
    break;

case "4":
    Console.WriteLine("Exiting program...");
    return;

default:
    Console.WriteLine("Invalid choice. Please enter 1, 2, 3, or 4.");
    break;
}

```

```
}  
  
}  
  
}
```

Output:

```
C:\Rohith\Backup\Desktop\Pr x + v  
  
Task Manager  
1. Add Task  
2. Remove Task  
3. Display Tasks  
4. Exit  
Choose an option: 1  
Enter a task: Complete the module  
Task 'COMPLETE THE MODULE' added.  
  
Task Manager  
1. Add Task  
2. Remove Task  
3. Display Tasks  
4. Exit  
Choose an option: 1  
Enter a task: Play Games  
Task 'PLAY GAMES' added.  
  
Task Manager  
1. Add Task  
2. Remove Task  
3. Display Tasks  
4. Exit  
Choose an option: |
```

```
C:\Rohith\Backup\Desktop\Pr x + v  
  
4. Exit  
Choose an option: 2  
Enter task to remove: complete the module  
Task 'COMPLETE THE MODULE' removed.  
  
Task Manager  
1. Add Task  
2. Remove Task  
3. Display Tasks  
4. Exit  
Choose an option: 3  
  
Your Tasks:  
- PLAY GAMES  
  
Task Manager  
1. Add Task  
2. Remove Task  
3. Display Tasks  
4. Exit  
Choose an option: 2  
Enter task to remove: sleep and relax  
Task not found.  
  
Task Manager  
1. Add Task  
2. Remove Task  
3. Display Tasks  
4. Exit  
Choose an option: |
```

```
Microsoft Visual Studio Debu x + v - □ x
1. Add Task
2. Remove Task
3. Display Tasks
4. Exit
Choose an option: 6
Invalid choice. Please enter 1, 2, 3, or 4.

Task Manager
1. Add Task
2. Remove Task
3. Display Tasks
4. Exit
Choose an option: 3

Your Tasks:
- PLAY GAMES

Task Manager
1. Add Task
2. Remove Task
3. Display Tasks
4. Exit
Choose an option: 4
Exiting program...

C:\Rohith\Backup\Desktop\Presidio\Pre-Training\4. C# Fundamentals\Task3\StringManipulation\StringManipulation\bin\Debug\net8.0\StringManipulation.exe (process 17548) exited with code 0 (0x0).
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .|
```