12214994_SaurabhRana

Title

Student Name Collection Using Non-Generic Collections

Question No : 1 / 1

Problem Statement: Student Name Collection

You need to create a program that **collects student names** and ensures **no duplicate names** are entered.
The program should use **non-generic collections in C#**.

The program should be written within the **Program class** in the **Program.cs** file.

Constraints

- **All classes must be public**
- **All methods must be private and static**
- Use **ArrayList** to store student names
- Do **not** use generic collections like `List<T>`
- Treat names as **case-insensitive** (e.g., `John` and `john` are considered the same)

Requirements

1. Input Names

- Continuously prompt the user to enter student names
- Stop taking input when the user enters the keyword:
- `stop`

---

2. Check for Duplicates

- Ensure duplicate names are **not added** to the collection

---

3. Display All Names

- After the user stops entering names, display **all unique student names** entered

---

Input Format

- Names entered by the user, **one at a time**
- The word `stop` to terminate the input process

Output Format

For each name entered

- If added:
- `<name> added to the collection.`
- If already exists:
- `<name> is already in the collection.`

Final Output

```
Unique student names entered:
<name1>
<name2>
...
```

Classes and Methods

Program Class

Main Method (`Main(string[] args)`)

- Collects student names
- Ensures no duplicates are added
- Stores messages for later display

- Calls `DisplayStudentNames` to print all unique student names

IsValidName Method

```
private static bool IsValidName(string name)
```

- Validates that the input name is **not null or empty**

IsNameInCollection Method

```
private static bool IsNameInCollection(ArrayList studentNames, string name)
```

- Checks if the name (case-insensitive) already exists in the collection

---

DisplayStudentNames Method

```
private static void DisplayStudentNames(ArrayList studentNames)
```

- Displays all unique student names entered by the user

Test Cases

Test Case ID	Input	Expected Output
TC01	`Test1`, `Test2`, `test1`, `stop`	Test1 added Test2 added test1 already exists
TC02	`Word`, `word`, `stop`	Word added word already exists
TC03	`A`, `B`, `C`, `stop`	A, B, C listed

Sample Input 1

```
Test1
Test2
test1
Test2
```

```
stop
```

Sample Output 1

```
Test1 added to the collection.
Test2 added to the collection.
test1 is already in the collection.
Test2 is already in the collection.
Unique student names entered:
Test1
Test2
```

Sample Input 2

```
Word
word
2
word3
word4
stop
```

Sample Output 2

```
Word added to the collection.
word is already in the collection.
2 added to the collection.
word3 added to the collection.
word4 added to the collection.
Unique student names entered:
Word
2
word3
word4
```

```
using System.Collections;


class Program
{
    private static bool IsValidName(string name)
    {
        return !string.IsNullOrWhiteSpace(name);
```

```
}
private static bool IsNameInCollection(ArrayList studentNames, string name)
{
   return studentNames.Contains(name);

}
private static void DisplayStudentNames(ArrayList studentNames)
{
   Console.WriteLine("Unique student names entered:");
   foreach(var names in studentNames)
   {
      Console.WriteLine(names);
   }
}
static void Main()
{
   ArrayList studentNames = new ArrayList();

   while (true)
   {
      string input = Console.ReadLine()??"";
      if (input.ToLower() == "stop")
      {
         break;
      }
```

```
            if (!IsValidName(input))
            {
               Console.WriteLine("Enter valid name.");
               continue;
            }

            if (IsNameInCollection(studentNames,input))
            {
               Console.WriteLine("{0} aleady in collection.",input);
               continue;
            }
            studentNames.Add(input);
            Console.WriteLine("{0} added to the collection.",input);
         }
         DisplayStudentNames(studentNames);
      }

}
```