PROBLEM

Find Min and Max in Array

Write a program to find largest and smallest number in the unsorted array. Assume that the maximum number of elements in an array is 50.

The link to download the template code is given below
Code Template

Input format :

Input consists of n+1 integers. The first integer corresponds to n, the number of elements in an array.

The next 'n' integers correspond to the elements in an array.

Output format :

Output displays largest and smallest element in the array.

Refer sample input and output for formatting specifications.

[All text in bold corresponds to the input and the rest corresponds to output.]

Sample Input and Output 1 :

Enter the number of elements in an array
5
Enter the elements in an array
3
6
2
4
1
Largest element is 6
Smallest element is 1

Program.cs/

```
using System;
class Program
    {
        static void Main(string[] args)
        {
            int[] a = new int[50];
            int i, n, large, small;
            Console.WriteLine("Enter the number of elements in an array");
            n = Convert.ToInt32(Console.ReadLine());
            Console.WriteLine("Enter the elements in an array");
            for(i=0; i<n; i++)
            a[i] = Convert.ToInt32(Console.ReadLine());
            large = small = a[0];
            for(i=1;i<n;i++)
            {
                if (a[i] > large)
                    large = a[i];
                if (a[i] < small)
                    small = a[i];
            }
            Console.WriteLine("Largest element is " + large);
            Console.WriteLine("Smallest element is " + small);
        }
    }

```

PROBLEM

Reverse an array

Write a program to reverse and display the array.

Input Format:

Input consists of n+1 integers. The first integer corresponds to n, the number of elements in an array.

The next 'n' integers correspond to the elements in an array.

Output format :

Display the reverse order of the array.

Refer sample input and output for formatting specifications.

Sample Input and Output1 :

```
Enter the number of elements in an array
6
Enter the elements in an array
46
58
96
22
55
10
Array after reversing
10
55
22
96
58
46
```

Program.cs/

```
using System;
class Program
    {
        static void Main(string[] args)
        {
            int n;
            Console.WriteLine("Enter the number of elements in an array");
            n = Convert.ToInt32(Console.ReadLine());
            int [] arr = new int[n];
            int i;
            Console.WriteLine("Enter the elements in an array");
            for(i=0;i<n;i++)

            arr[i] = Convert.ToInt32(Console.ReadLine());

            Console.WriteLine("Array after reversing");
            for(i=n-1;i>=0;i--)
            {
                Console.WriteLine(""+ arr[i]);
            }
            Console.ReadLine();

        }
    }

```

PROBLEM

Display Stall Array

Bdaah Contemporary Fashion and Lifestyle Exhibition is hosted by Bdaah Online. Focusing on sustainability, the event will have a section dedicated to handloom products such as clothing and textiles and there will also be an Ayurvedic section. Leather goods and Homeware products will also be showcased at the event. The Organizing committee plans to make a list of stalls in the Exhibition.
Help them to collect the stall details and print the list of stalls in detailed view.

Write a C# program with following classes and methods to get list of stall details from user and print them in detailed view.

[Note : Strictly adhere to the object oriented specifications given as a part of the problem statement.
Follow the naming conventions as mentioned. Create separate classes in separate files.]

Consider a class named **Stall** with the following member variables/ attributes:

Data Type	Attributes
string	_name
string	_owner
string	_stallType

Include appropriate **getters and setters**.
Include **default** and **parameterized** constructor for the class.
Prototype for the Parameterized Constructor **Stall(string _name, string _owner, string _stallType)**

Consider another class **Program** and write a Main method to test the above class.
In the main method read the stall details from the user and display them.

The link to download the template code is given below
Code Template

Input and Output Format:
Display the stall array in the insertion order.
Refer sample input and output for formatting specifications.
Use the specifier **Console.WriteLine(String.Format("{0,-15}{1,-15}{2,-20}", "Name", "Owner name", "Stall type"))** to display the stall details in the neat format

Sample Input and Output 1:

```
Enter the number of stalls
3
Enter stall details - 1
Enter the name
Wok and roll
Enter the owner name
Jonas
Enter the stall type
Promotional
Enter stall details - 2
Enter the name
Chez creations
Enter the owner name
Elizabeth
Enter the stall type
Promotional
Enter stall details - 3
Enter the name
Food plaza
Enter the owner name
Dustin
Enter the stall type
Maxima
Name        Owner name    Stall type
Wok and roll   Jonas        Promotional
Chez creations Elizabeth    Promotional
Food plaza    Dustin        Maxima
```

Stall.cs/ Program.cs/

```
using System;
class Stall
{
    private string _name;
    private string _owner;
    private string _stallType;

    public string Name
    {
        get => _name;
        set => _name = value;
    }
    public string Owner
    {
        get => _owner;
        set => _owner = value;
    }
    public string StallType
    {
        get => _stallType;
        set => _stallType = value;
    }
    public Stall()
    {

    }
    public Stall(string _name, string _owner, string _stallType)
    {
        this._name = _name;
        this._owner = _owner;
        this._stallType = _stallType;
    }
    public void display()
    {
        Console.WriteLine((String.Format("{0,-15}{1,-15}{2,-20}", Name, Owner, StallType)));
    }
}
```

Stall.cs/ Program.cs/

```
using System;
class Program
{
    public static void Main(string[] args)
    {
        int n;
        Console.WriteLine("Enter the number of stalls");
        n = Convert.ToInt32(Console.ReadLine());
        Stall[] stalls = new Stall[n];
        for (int i = 0; i < n; i++)
        {
            Console.WriteLine("Enter stall details - "+(i+1));
            Console.WriteLine("Enter the name");
            String _name = Console.ReadLine();
            Console.WriteLine("Enter the owner name");
            String _owner = Console.ReadLine();
            Console.WriteLine("Enter the stall type");
            String _stallType = Console.ReadLine();
            stalls[i] = new Stall(_name, _owner, _stallType);
        }
        Console.WriteLine(String.Format("{0,-15}{1,-15}{2,-20}", "Name", "Owner name", "Stall type"));
        for(int i=0; i<n; i++)
        {
            stalls[i].display();
        }
    }
}
```

PROBLEM

String Functions

Write a program to use string inbuilt functions and properties.

The program should provide a menu to process the following
1. Find length
2. Compare two string
3. Split a string

Consider a class **Program** with **Main** method, to display the menu and process it using the string inbuilt functions and property.

The link to download the template code is given below
Code Template

Input and Output Format:

Refer sample input and output for formatting specifications.

Sample Input and Output 1:

```
Menu
1.Find length of a string
2.Compare two string
3.Split a string
4.Exit
Enter your choice
1
Enter the sentence
Apple is good for health
Length of the string 'Apple is good for health' is 24
Menu
1.Find length of a string
2.Compare two string
3.Split a string
4.Exit
Enter your choice
2
Enter the first sentence
apple
Enter the second sentence
apple
'apple' and 'apple' are equal
Menu
1.Find length of a string
2.Compare two string
3.Split a string
4.Exit
Enter your choice
3
Enter the delimiter
|
Enter the sentence to split
Apple|Orange|Pineapple
Separated words are
1. Apple
2. Orange
3. Pineapple
Menu
1.Find length of a string
2.Compare two string
3.Split a string
4.Exit
Enter your choice
4
```

Sample Input and Output 2:

```
Menu
1.Find length of a string
2.Compare two string
3.Split a string
4.Exit
Enter your choice
2
Enter the first sentence
Apple
Enter the second sentence
apple
'Apple' and 'apple' are not equal
Menu
1.Find length of a string
2.Compare two string
3.Split a string
4.Exit
Enter your choice
4
```

Additional Sample TestCases

Sample Input and Output 1 :

```
Menu
1.Find length of a string
2.Compare two string
3.Split a string
4.Exit
Enter your choice
2
Enter the first sentence
Apple
Enter the second sentence
apple
'Apple' and 'apple' are not equal
Menu
1.Find length of a string
2.Compare two string
3.Split a string
4.Exit
Enter your choice
4
```

Program.cs/

```
using System;
class Program
{
    public static void Main(string[] args)
    {
        int choice;
        // string str;
        // string str1;
        do
        {
            Console.WriteLine("Menu\n1.Find length of a string\n2.Compare two string\n3.Split a string\n4.Exit");
            Console.WriteLine("Enter your choice");
            choice = Convert.ToInt32(Console.ReadLine());

            switch (choice)
            {
                case 1:
                    Console.WriteLine("Enter the sentence");
                    string str = Console.ReadLine();
                    int length = str.Length;
                    Console.WriteLine("Length of the string '{0}' is {1}" ,str,length);
                    break;
                case 2:
                    Console.WriteLine("Enter the first sentence");
                    str = Console.ReadLine();
                    Console.WriteLine("Enter the second sentence");
                    string str1 = Console.ReadLine();
                    if (String.Compare(str, str1)==0)
                        Console.WriteLine("'{0}' and '{1}' are equal",str,str1);
                    else
                        Console.WriteLine("'{0}' and '{1}' are not equal",str,str1);
                    break;
                case 3:
                    Console.WriteLine("Enter the delimiter");
                    char del = Convert.ToChar(Console.ReadLine());
                    Console.WriteLine("Enter the sentence to split");
                    str = Console.ReadLine();
                    Console.WriteLine("Separated words are");
                    String[] split_str = str.Split(del);
                    int j = 1;
                    foreach (var word in split_str)
                    {
                        Console.WriteLine("{0}. {1}",j,word);
                        j++;
                    }
                    break;
                case 4:
                    System.Environment.Exit(0);
                    break;
                default:
                    break;
            }
        } while (true);
    }
}
```

PROBLEM

Advanced Password Strength

The placement coordinator creates a unique student profile for all the students. He gives them a username and password and asks them to reset their password. Students can use their profile for login during placement drives. So the coordinator wanted to ensure the password strength.

Find the strength of the password based on the rules given below.
1. If the length of the password is less than 8, then the password strength is 'weak' and display '**Password strength is weak**'
2. If the password length is greater than eight but does not contain any special characters or numbers, the strength is considered as 'ok' and display '**Password strength is ok**'
3. If the password length is greater than 8 and contains special characters/numbers, the strength is considered as 'good' and display '**Password strength is good**'

Write a program to find the strength of the password.

Consider a class **Program** and write a Main method to test the above password strength.

The link to download the template code is given below
Code Template

Input and Output Format:
Refer sample input and output for formatting specifications.

[All text in bold corresponds to input and the rest corresponds to output]

Sample Input and Output 1:

Enter name of student
Devin
Enter the password
devin
Student name: Devin, password: devin
Password strength is weak

Sample Input and Output 2:

Enter name of student
Steffan
Enter the password
Steffan7@#
Student name: Steffan, password: Steffan7@#
Password strength is good

Sample Input and Output 3:

Enter name of student 3
Zahra
Enter the password
Zahrachiyo
Student name: Zahra, password: Zahrachiyo
Password strength is ok

Program.cs/

```
1  using System;
2  class Program
3      {
4          static void Main(string[] args)
5          {
6              string name;
7              string password;
8
9              Console.WriteLine("Enter name of student");
10             name = Console.ReadLine();
11             Console.WriteLine("Enter the password");
12             password = Console.ReadLine();
13             Console.Write("Student name: " + name+ ", password: " +password);
14
15
16             if ((password.Length > 8) && (((password.Contains("!")|| (password.
17
18             {
19                 Console.WriteLine("Password strength is good");
20             }
21             else if (password.Length > 8)
22             {
23                 Console.WriteLine("Password strength is ok");
24             }
25             else
26             {
27                 Console.WriteLine("Password strength is weak");
28             }
29             Console.ReadLine();
30         }
31     }
```

IF line

```
if ((password.Length > 8) && (((password.Contains("!")||
(password.Contains("@")||(password.Contains("#") || (password.Contains("$") ||
(password.Contains("%") || (password.Contains("^") || (password.Contains("&") ||
(password.Contains("*") || (password.Contains("(") || (password.Contains(")") ||
(password.Contains("!") || (password.Contains("1") || (password.Contains("1") ||
(password.Contains("2") || (password.Contains("3") || (password.Contains("4") ||
(password.Contains("5") || (password.Contains("6") || (password.Contains("7") ||
(password.Contains("8") || (password.Contains("9") || (password.Contains("10"))))))))))))))))))))))))))
```