## 1.Create lists

var names=new List<string>{"Ram","Raheem","Robert"};

foreach(var name in names)

{

Console.WriteLine($"Hello {name.ToUpper()}!");

}

O/P: Hello RAM!

Hello RAHEEM!

Hello ROBERT!

2. //Modify list contents

var names=new List<string>{"Ram","Raheem","Robert"};

//Console.WriteLine();

names.Add("Maria");

names.Add("Bill");

names.Add("al");

names.Remove("al");

foreach(var name in names)

{

Console.WriteLine($"Hello {name.ToUpper()}!");

}

O/P: Hello RAM!

Hello RAHEEM!

Hello ROBERT!

Hello MARIA!

Hello BILL!

3. //Search and sort lists

var names=new List<string>{"Ram","Raheem","Robert"};

var index=names.IndexOf("Robert");

if(index!=-1)

Console.WriteLine($"The name {names[index]} is at index {index}");

var notFound = names.IndexOf("Not Found");

Console.WriteLine($"When an item is not found, IndexOf returns {notFound}");

//Sorting

Console.WriteLine();

Console.WriteLine("Sorting:");

names.Sort();

foreach (var name in names)

{

Console.WriteLine($"Hello {name.ToUpper()}!");

}

O/P: The name Robert is at index 2

When an item is not found, IndexOf returns -1

Sorting:

Hello RAHEEM!

Hello RAM!

Hello ROBERT!

4. //Lists of other types

var fibonacciNumbers = new List<int> {1, 1};

var previous = fibonacciNumbers[fibonacciNumbers.Count - 1];

var previous2 = fibonacciNumbers[fibonacciNumbers.Count - 2];

fibonacciNumbers.Add(previous + previous2);

foreach(var item in fibonacciNumbers)

Console.WriteLine(item);

O/P: 1

1

2

5. //Challenge

var fibonacciNumbers = new List<int> {1, 1};

while (fibonacciNumbers.Count < 20)

{

var previous = fibonacciNumbers[fibonacciNumbers.Count - 1];

var previous2 = fibonacciNumbers[fibonacciNumbers.Count - 2];

fibonacciNumbers.Add(previous + previous2);

}

foreach(var item in fibonacciNumbers)

Console.WriteLine(item);

O/P: 1

1

2

3

5

8

13

21

34

55

89

144

233

377

610

987

1597

2584

4181

6765