

Lists CS109

The List object behaves like a dynamic array. We can add and remove from it at will, and also delete items at specific locations. This can be useful when you don't know how many items will be processed in your array and you can then use a List instead. The textbook doesn't discuss Lists, but they're pretty straightforward to use if you understand arrays. Lists are just a simpler version of a Collection (The collection allows you to index values with different keys, while the list forces you to index values with a key identical to the value).

The following code sample illustrates the usage:

```
Dim stringList As New List(Of String)
Dim i As Integer

stringList.Add("foo")
stringList.Add("bar")
stringList.Add("zot")
stringList.Add("bah")

Console.WriteLine("Size of List: " & stringList.Count)
Console.WriteLine("Contents:")
For i = 0 To stringList.Count - 1
    Console.WriteLine("At index " & i & " Value=" & stringList(i))
Next
stringList.Remove("bar")
Console.WriteLine("Contents after remove bar:")
For i = 0 To stringList.Count - 1
    Console.WriteLine("At index " & i & " Value=" & stringList(i))
Next
stringList.RemoveAt(0)
Console.WriteLine("Contents after remove:")
For i = 0 To stringList.Count - 1
    Console.WriteLine("At index " & i & " Value=" & stringList(i))
Next
```

Output:

```
Size of List: 4
Contents:
At index 0 Value=foo
At index 1 Value=bar
At index 2 Value=zot
At index 3 Value=bah
Contents after remove bar:
At index 0 Value=foo
At index 1 Value=zot
At index 2 Value=bah
Contents after remove:
At index 0 Value=zot
At index 1 Value=bah
```

As you can see, the List is convenient because we don't need to declare the size up front like an array, and we are able to delete (or insert) items at specific places in the List. Both of these operations are not allowed with normal arrays.

One last operation that you may find useful is the IndexOf function. It tells you the position of a matching item, and -1 if there is no match. For example:

```
stringList.Add("foo")
stringList.Add("bar")
stringList.Add("zot")
stringList.Add("bah")

Console.WriteLine(stringList.IndexOf("foo"))
Console.WriteLine(stringList.IndexOf("zot"))
Console.WriteLine(stringList.IndexOf("moo"))
```

This code outputs:

```
0
2
-1
```

The string "foo" is found at position 0, and "zot" at position 2. , and "moo" gives us -1 because it does not exist in the List.

Exercise: Convert the Trivia program to use Lists instead of arrays.