Collections Demo

Demonstrate how to use collections

# Objectives

* 1. Demonstrate how to use Lists.

# create a project

* 1. Create a new C# application called CollectionsDemo.

# Lists

* 1. In Program.cs create a list, add some elements and display it using a for loop:

class Program

{

static void Main(string[] args)

{

List<string> Names = new List<string>();

Names.Add("Linus Torvalds");

Names.Add("Donald Knuth");

Names.Add("Grace Hopper");

for(int i=0;i < Names.Count;++i)

{

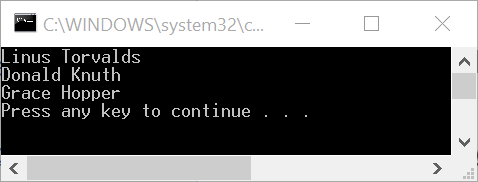
Console.WriteLine(Names[i]);

}

}

}

* 1. Try it with Ctrl-F5:

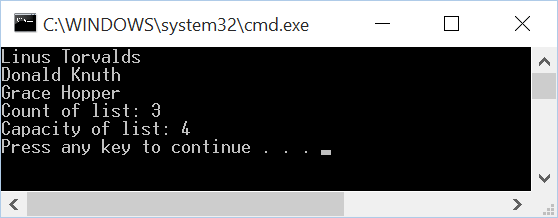


* 1. Add code to display the count and capacity of our list:

Console.WriteLine("Count of list: " + Names.Count.ToString());

Console.WriteLine("Capacity of list: " + Names.Capacity.ToString());

* 1. Try it with Ctrl-F5:

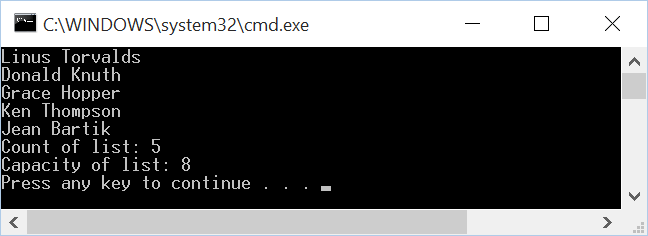


* 1. Add two more names:

Names.Add("Donald Knuth");

Names.Add("Jean Bartik");

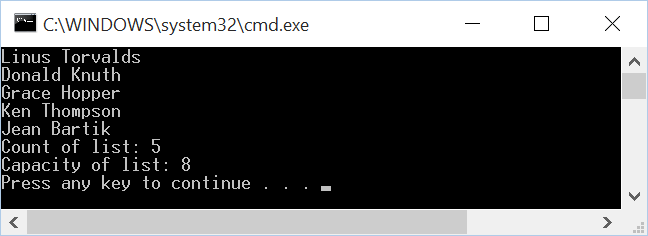
* 1. Run it again:



* 1. Notice that the capacity jumped to 8!
  2. Trim the list:

Names.TrimExcess();

* 1. Run it again:



* 1. Notice the capacity is now the same size as the count.
  2. Remove item at index 3:

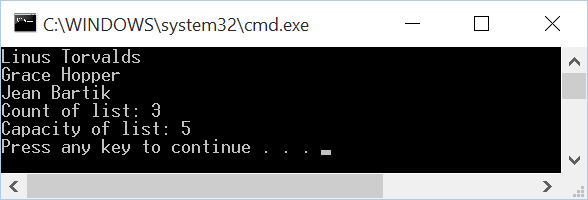
Names.RemoveAt(3);

* 1. Run it again:



* 1. Remove another item using the Remove method:

Names.Remove("Donald Knuth");



* 1. Extract the code to display the list into a function:

private static void DisplayNames(List<string> Names)

{

for (int i = 0; i < Names.Count; ++i)

{

Console.WriteLine(i+1+"; "+Names[i]);

}

Console.WriteLine();

}

* 1. Change remaining code so that we have a do while loop and provide user ability to work with list:

string doAnother;

do

{

DisplayNames(Names);

Console.Write("(A)dd, (R)emove, (S)earch: ");

string operation = Console.ReadLine();

switch (operation)

{

case "A":

Console.Write("Name: ");

Names.Add(Console.ReadLine());

break;

case "R":

Console.Write("Index or name: ");

string nameOrIndex = Console.ReadLine();

int index;

if(int.TryParse(nameOrIndex,out index))

{

Names.RemoveAt(index-1);

}

else

{

Names.Remove(nameOrIndex);

}

break;

case "S":

Console.Write("Name: ");

string name = Console.ReadLine();

Console.WriteLine("Index is " + Names.IndexOf(name)+1);

break;

}

Console.Write("Do another(y/n)? ");

doAnother = Console.ReadLine();

} while (doAnother == "y");

* 1. Run with Ctrl-F5 and experiment with adding, removing and searching for items.