# Practice GenericQueue

Create a GenericQueue generic class that uses and wraps a private field of **System.Collections.Queue** type (NOT the built in generic Queue type!).

Add GenericQueue Enqueue , Dequeue , Peek , Contains , GetItemsTypeName methods, Count Property that operates on the items stored in the private Queue collection , the operations will be of the Type defined when the GenericQueue was initialized (in the following example : int)

Use this testing code :

GenericQueue<int> numbersQ= new GenericQueue<int>();

System.Console.WriteLine($"GenericQueue API tester for {numbersQ.GetItemsTypeName()}:");

System.Console.WriteLine($"numbersQ.Count={numbersQ.Count}");

numbersQ.Enqueue(1);

numbersQ.Enqueue(2);

numbersQ.Enqueue(3);

numbersQ.Enqueue(4);

System.Console.WriteLine($"numbersQ.Count={numbersQ.Count}");

System.Console.WriteLine($"numbersQ.Contains(4)={numbersQ.Contains(4)}");

System.Console.WriteLine($"numbersQ.Contains(5)={numbersQ.Contains(5)}");

System.Console.WriteLine($"numbersQ.Peek()={numbersQ.Peek()}");

System.Console.WriteLine($"numbersQ.Dequeue()={numbersQ.Dequeue()}");

System.Console.WriteLine($"numbersQ.Dequeue()={numbersQ.Dequeue()}");

System.Console.WriteLine($"numbersQ.Dequeue()={numbersQ.Dequeue()}");

System.Console.WriteLine($"numbersQ.Dequeue()={numbersQ.Dequeue()}");

System.Console.WriteLine($"numbersQ.Count={numbersQ.Count}");

To produce this output:

A screenshot of a computer

Description automatically generated