

BAB VI

Collection

❖ Collection

Collection adalah Kelas pada C# yang disediakan untuk menyimpan data. Kelebihan Collection untuk menyimpan data adalah karena Collection lebih dinamis dalam penyimpanan data. Beberapa Kelas Collection Pada C# adalah ArrayList, SortedList, Stack, Queue, dan HashTable.

Stack

Stack(*tumpukan*) adalah salah satu tipe Collection untuk menyimpan elemen data dengan model penyimpanan **LIFO(Last In First Out)**. Stack bisa menyimpan nilai null ataupun duplikat nilai.

Untuk Inisialisasinya adalah sebagai berikut

```
Stack myStack = new Stack();
```

Berikut beberapa fungsi yang penting dan sering digunakan pada Stack

Method	Deskripsi
Push	Fungsi untuk memasukan item/element data kedalam Stack
Peek	Untuk mengambil item paling atas dari Stack
Pop	Menghapus dan mengambil item paling atas dari Stack
Contains	Untuk mengecek apakah item tertentu sudah ada pada Stack
Clear	Menghapus semua item pada Stack

Queue

Queue(*antrian*) adalah kebalikan dari Stack, Queue adalah salah satu tipe Collection untuk menyimpan elemen data dengan model penyimpanan **FIFO(First In First Out)**. Queue bisa menyimpan nilai null ataupun duplikat nilai.

```
Queue queue = new Queue();
```

Berikut beberapa fungsi yang penting dan sering digunakan pada Queue.

Method	Usage
Enqueue	Untuk memasukan item/element data kedalam Queue
Dequeue	Menghapus dan mengambil item paling awal dari Queue
Peek	Untuk mengambil item paling atas dari Queue
Contains	Untuk mengecek apakah item tertentu sudah ada pada Queue
Clear	Menghapus semua item pada Queue

PERCOBAAN

1. Penambahan data pada Stack

```
using System;
using System.Collections;

public class Program
{
    public static void Main()
    {
        Stack myStack = new Stack();
        myStack.Push("Selamat Datang");
        myStack.Push(null);
        myStack.Push(1);
        myStack.Push(2);
        myStack.Push(3);

        foreach (var item in myStack)
        {
            Console.WriteLine(item);
        }
        Console.ReadKey();
    }
}
```

2. Cara mengakses Elemen Stack

```
using System;
using System.Collections;

public class Program
{
    public static void Main()
    {
        Stack myStack = new Stack();
        myStack.Push(1);
        myStack.Push(2);
        myStack.Push(3);
        myStack.Push(4);
        myStack.Push(5);

        Console.WriteLine(myStack.Peek());
        Console.WriteLine(myStack.Peek());
        Console.WriteLine(myStack.Peek());
        Console.ReadKey();
    }
}
```

3. Cara mengambil Elemen dari Stack

```
using System;
using System.Collections;

public class Program
{
    public static void Main()
    {
        Stack myStack = new Stack();
        myStack.Push(1);
        myStack.Push(2);
        myStack.Push(3);
        myStack.Push(4);
        myStack.Push(5);

        Console.WriteLine("Jumlah elemen pada Stack : {0}", myStack.Count);

        while (myStack.Count > 0)
        {
            Console.WriteLine(myStack.Pop());
        }
        Console.WriteLine("Jumlah elemen pada Stack : {0}", myStack.Count);
        Console.ReadKey();
    }
}
```

4. Mengecek keberadaan elemen pada Stack

```
using System;
using System.Collections;

public class Program
{
    public static void Main()
    {
        Stack myStack = new Stack();
        myStack.Push(1);
        myStack.Push(2);
        myStack.Push(3);
        myStack.Push(4);
        myStack.Push(5);

        Console.WriteLine(myStack.Contains(2));
        Console.WriteLine(myStack.Contains(10));
        Console.ReadKey();
    }
}
```

5. Menghapus Elemen pada Stack

```
using System;
using System.Collections;

public class Program
{
    public static void Main()
    {
        Stack myStack = new Stack();
        myStack.Push(1);
        myStack.Push(2);
        myStack.Push(3);
        myStack.Push(4);
        myStack.Push(5);

        myStack.Clear();

        Console.WriteLine("Jumlah elemen pada Stack : {0}", myStack.Count);
        Console.ReadKey();
    }
}
```

6. Memasukan dan Menampilkan data pada Queue

```
using System;
using System.Collections;

public class Program
{
    public static void Main()
    {
        Queue queue = new Queue();
        queue.Enqueue(3);
        queue.Enqueue(2);
        queue.Enqueue(1);
        queue.Enqueue("Empat");

        Console.WriteLine("Jumlah elemet pada Queue : {0}", queue.Count);

        while (queue.Count > 0)
        {
            Console.WriteLine(queue.Dequeue());
        }
        Console.WriteLine("Jumlah elemet pada Queue : {0}", queue.Count);
        Console.ReadKey();
    }
}
```

7. Mengakses elemen pada Queue

```
using System;
using System.Collections;

public class Program
{
    public static void Main()
    {
        Queue queue = new Queue();
        queue.Enqueue(3);
        queue.Enqueue(2);
        queue.Enqueue(1);
        queue.Enqueue("Empat");

        Console.WriteLine("Jumlah elemen pada Queue: {0}", queue.Count);

        Console.WriteLine(queue.Peek());
        Console.WriteLine(queue.Peek());
        Console.WriteLine(queue.Peek());

        Console.WriteLine("Jumlah elemen pada Queue: {0}", queue.Count);
        Console.ReadKey();
    }
}
```

8. Mengecek keberadaan elemen pada Queue

```
using System;
using System.Collections;

public class Program
{
    public static void Main()
    {
        Queue queue = new Queue();
        queue.Enqueue(3);
        queue.Enqueue(2);
        queue.Enqueue(1);
        queue.Enqueue("Empat");

        Console.WriteLine(queue.Contains(2));
        Console.WriteLine(queue.Contains(100));
        Console.ReadKey();
    }
}
```

9. Menghapus elemen pada Queue

```
using System;
using System.Collections;

public class Program
{
    public static void Main()
    {
        Queue queue = new Queue();
        queue.Enqueue(3);
        queue.Enqueue(2);
        queue.Enqueue(1);
        queue.Enqueue("Empat");

        Console.WriteLine("Jumlah elemen pada Queue: {0}", queue.Count);

        queue.Clear();

        Console.WriteLine("Jumlah elemen pada Queue: {0}", queue.Count);
        Console.ReadKey();
    }
}
```

PRAKTIKUM WORKSHOP

1. Cobalah program di atas.
2. Berikan analisa dari setiap Program yang dijalankan
3. Buatlah pada laporan dan di Upload sebagai tugas Workshop.