**Program.cs**

using System;

using System.Collections;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ExceptionAssignment

{

class Program

{

static void Main(string[] args)

{

try

{

Stack stack = CheckStack();

foreach (var stackItem in stack)

{

Console.WriteLine(stackItem);

}

}

catch (StackException ex)

{

Console.WriteLine(ex.Message);

}

Console.ReadKey();

}

static Stack CheckStack()

{

Console.WriteLine("The size of the stack is 5");

Stack mystack = new Stack(5);

mystack.Push(1);

mystack.Push(2);

mystack.Push(3);

mystack.Push(4);

mystack.Push(5);

mystack.Push(6);

Console.WriteLine("No of elements in stack is " + mystack.Count);

if (mystack.Count > 5)

{

throw new StackException("Stack Overflow");

}

else if (mystack.Count == 0)

{

throw new StackException("Stack Underflow");

}

return mystack;

}

}

}

**StackException.cs**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ExceptionAssignment

{

internal class StackException : Exception

{

public StackException(string msg) : base(msg) { }

}

}