using System;

using System.Collections;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace EmployeeDemo

{

class CollectionsDemo

{

static void Main()

{

ArrayList list = new ArrayList();

list.Add(1);

list.Add(2);

list.Add(3);

Console.WriteLine("Elements are ");

foreach(int temp in list)

Console.WriteLine(temp);

list.Insert(1, 100);

Console.WriteLine("Elements after insertion are ");

foreach (int temp in list)

Console.WriteLine(temp);

list.Remove(2);

Console.WriteLine("Elements after deletion are ");

foreach (int temp in list)

Console.WriteLine(temp);

list.RemoveAt(0);

Console.WriteLine("Elements after deletion are ");

foreach (int temp in list)

Console.WriteLine(temp);

Console.WriteLine("STACK");

Stack stack = new Stack();

stack.Push(1);

stack.Push(2);

stack.Push(3);

Console.WriteLine("Elements of stack are ");

foreach (int temp in stack)

Console.WriteLine(temp);

stack.Pop();

Console.WriteLine("Elements of stack after removing are ");

foreach (int temp in stack)

Console.WriteLine(temp);

Console.WriteLine("QUEUE");

Queue queue = new Queue();

queue.Enqueue(1);

queue.Enqueue(2);

queue.Enqueue(3);

Console.WriteLine("Elements of queue are ");

foreach (int temp in queue)

Console.WriteLine(temp);

queue.Dequeue();

Console.WriteLine("Elements of queue after removing are ");

foreach (int temp in queue)

Console.WriteLine(temp);

Hashtable ht = new Hashtable();

// HashTable stores values in key,value

ht[1] = 90;

ht[2] = 89;

ht[10] = 76;

Console.WriteLine("Marks of RollNo 2: " + ht[2]);

foreach(int key in ht.Keys)

Console.WriteLine($"Marks of {key} are {ht[key]}");

}

}

}

using System;

using System.Collections;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace EmployeeDemo

{

class GenericCollectionsDemo

{

static void Main()

{

List<int> list = new List<int>();

list.Add(1);

list.Add(2);

foreach(int x in list)

Console.WriteLine(x);

Stack<string> stack = new Stack<string>();

stack.Push("Ajay");

Queue<float> queue = new Queue<float>();

queue.Enqueue(1.4f);

Dictionary<int, int> result = new Dictionary<int, int>();

result[1] = 100;

result[2] = 90;

}

}

}

Jagged Array

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace EmployeeDemo

{

class JaggedArray

{static void Main()

{

// Declare Jagged Array

int[][] marks = new int[10][];

Console.WriteLine("Enter Mars for Student 1");

marks[0] = new int[2];

marks[0][0] = 10;

marks[0][1] = 90;

Console.WriteLine("Enter Mars for Student 2");

marks[1] = new int[5];

marks[1][0] = 10;

marks[1][1] = 90;

marks[1][2] = 10;

marks[1][3] = 90;

marks[1][4] = 10;

for(int i=0;i<marks.Length;i++)

{

Console.WriteLine($"Marks of Student {i+1}");

for(int j=0; j<marks[i].Length;j++)

{

Console.Write(marks[i][j] + " ");

}

}

}

}

}