**ASSIGNMENT 5**

1)

using System;

using System.Collections.Generic;

using System.Text;

namespace Collections

{

class Arrays

{

public void integer()

{

Console.WriteLine("Enter Size of an Array:");

int size = int.Parse(Console.ReadLine());

int[] arr = new int[size];

int[] arr2 = new int[size];

Console.WriteLine("Enter Elements to the array:");

for (int i = 0; i < size; i++)

{

arr[i] = int.Parse(Console.ReadLine());

}

Array.Copy(arr, arr2, size);

Console.WriteLine("Copy Elements:");

foreach (int array in arr)

{

Console.WriteLine(array);

}

Array.Sort(arr);

Console.WriteLine("Sort Elements");

foreach (int a in arr)

{

Console.WriteLine(a);

}

Array.Clear(arr, 2, 2);

Console.WriteLine("Clear Elements:");

foreach (int n in arr)

{

Console.WriteLine(n);

}

Array.Reverse(arr);

Console.WriteLine("Reverse Elements:");

foreach (int b in arr)

{

Console.WriteLine(b);

}

}

public void String()

{

Console.WriteLine("Enter Size of an Array:");

int size = int.Parse(Console.ReadLine());

string[] str = new string[size];

string[] str2 = new string[size];

Console.WriteLine("Enter Strings to the array:");

for (int i = 0; i < size; i++)

{

str[i] = Console.ReadLine();

}

Array.Copy(str, str2, size);

Console.WriteLine("Copy Strings:");

foreach (string array in str)

{

Console.WriteLine(array);

}

Array.Sort(str);

Console.WriteLine("Sort Strings:");

foreach (string a in str)

{

Console.WriteLine(a);

}

Array.Clear(str, 1, 2);

Console.WriteLine("Clear Strings:");

foreach (string n in str)

{

Console.WriteLine(n);

}

Array.Reverse(str);

Console.WriteLine("Reverse Strings:");

foreach (string b in str)

{

Console.WriteLine(b);

}

}

public static void Main(string[] args)

{

Arrays arr = new Arrays();

arr.integer();

Console.WriteLine("---------------");

arr.String();

}

}

}

2)

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Collections.Framework

{

public class Employee<A>

{

A[] Obj = new A[10];

int count = 0;

public void add(A item)

{

if (count + 1 < 11)

{

Obj[count] = item;

}

count++;

Console.WriteLine("no.of employees" + count);

}

public A this[int index]

{

get { return Obj[index]; }

set { Obj[index] = value; }

}

}

class Program

{

static void Main(string[] args)

{

Program a = new Program();

Employee<string> emp = new Employee<string>();

emp.add("Rick");

emp.add("Judith");

emp.add("Michonne");

emp.add("Mary");

emp.add("Lydia");

emp.add("Alden");

emp.add("Nora");

emp.add("Scarlett");

emp.add("Enid");

emp.add("Carol");

for (int i = 0; i < 10; i++)

{

Console.WriteLine("Employee details:");

Console.WriteLine(emp[i]);

}

}

}

}

3)

using System;

namespace Collections

{

class MyStack<t>

{

int size;

t[] stack;

int top;

public MyStack()

{

size = 10;

stack = new t[size];

}

public int Push(t element)

{

if (top == size - 1)

{

Console.WriteLine("stack overflows");

}

else

{

top = top + 1;

stack[top] = element;

}

return 0;

}

public t Pop()

{

t Removedele;

t temp = default(t);

if (top < 0)

{

Removedele = stack[top];

top = top + 1;

return Removedele;

}

return temp;

}

public t[] GetStackEle()

{

t[] elements = new t[top + 1];

Array.Copy(stack, 0, elements, 0, top + 1);

return elements;

}

}

class Program

{

static void Main(string[] args)

{

MyStack<string> S = new MyStack<string>();

while (true)

{

Console.WriteLine("1.Push");

Console.WriteLine("2.Pop");

Console.WriteLine("3.Exit");

Console.WriteLine("Enter choice");

int choice = int.Parse(Console.ReadLine());

switch (choice)

{

case 1:

{

Console.WriteLine("enter string to push");

string temp = Console.ReadLine();

int result = S.Push(temp);

if (result != -1)

{

Console.WriteLine("element pushed into stack");

}

else

{

Console.WriteLine("stack overflows");

}

break;

}

case 2:

{

string Result = S.Pop();

if (Result != null)

{

Console.WriteLine("delete element; " + Result);

}

else

{

Console.WriteLine("stack underflow");

}

break;

}

case 3:

{

System.Diagnostics.Process.GetCurrentProcess().Kill();

break;

}

default:

{

Console.WriteLine("you have entered wrong choice");

break;

}

}

}

}

}

}

5)

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Collection.framework

{

class Program

{

static void Main(string[] args)

{

//search employee name in the list

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

emp.add("Rick");

emp.add("Judith");

emp.add("Michonne");

emp.add("Mary");

emp.add("Lydia");

emp.add("Alden");

emp.add("Nora");

emp.add("Scarlett");

emp.add("Enid");

emp.add("Carol");

Console.WriteLine(Employeelist.Contains("Mary"));

}

}

}

6)

using System;

using System.IO;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Iterator

{

class Team

{

private string PlayerName;

private int RunsScored;

public string playername

{

get { return PlayerName; }

set { PlayerName = value; }

}

public int runsscored

{

get { return RunsScored; }

set { RunsScored = value; }

}

public void Display()

{

Console.WriteLine("player name :" + PlayerName);

Console.WriteLine("Runs Scored : " + RunsScored);

}

}

class Player

{

static void Main()

{

Console.WriteLine("enter size");

int n = int.Parse(Console.ReadLine());

string[] arr = new string[n];

for (int i = 0; i < n; i++)

{

Console.WriteLine("Player Name : ");

arr[i] = Console.ReadLine();

if (i < n)

{

Console.WriteLine("Runs Scored : ");

int RunsScored = int.Parse(Console.ReadLine());

}

}

Team T = new Team();

T.playername = "Mahi";

T.runsscored = 100;

T.Display();

}

}

}