using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Text.RegularExpressions;

using System.Threading.Tasks;

namespace CSharpAssignment1

{

class Program

{

delegate List<int> delegate\_obj1(List<int> abc);

static void Main(string[] args)

{

Console.WriteLine("====================\*\*\* 2D array with Size 3\*3 \*\*\*============");

\_TwoDArray p1 = new \_TwoDArray();

p1.run();

Console.WriteLine("====================\*\*\* Multiple Inheritance \*\*\*============");

CS1\_MulitpleInheritance p2 = new CS1\_MulitpleInheritance();

p2.message();

Console.WriteLine("====================\*\*\* Abstract Virtual \*\*\*============");

childClass ch = new childClass();

ch.sum();

Console.WriteLine("====================\*\*\* Delegate Instances Example \*\*\*============");

Math m = new Math();

delegate\_obj1 \_delegateobj = m.findAllDivisibleBy3;

List<int> numberlist = new List<int> { 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 18, 21, 30 };

Console.Write("Delegate Object calling divisible by 3 list from numlist : " );

\_delegateobj(numberlist);

Console.WriteLine();

Console.WriteLine("====================\*\*\* Lambda Expression \*\*\*============");

Console.Write("Lambda Expression calling divisible by 3 list from numlist ");

foreach (var item in numberlist.FindAll(j=> j%3==0))

{

Console.Write(item+", ");

}

Console.WriteLine();

Console.WriteLine("====================\*\*\* Extend of System.String with function IsEmail() \*\*\*============");

Console.WriteLine("valid mailid Check for : senthil@test.com is {0}", commonUtilities.isEMail("senthil@test.com"));

Console.WriteLine("valid mailid Check for : senthiltest.com is {0}", commonUtilities.isEMail("senthiltest.com"));

Console.ReadLine();

}

}

#region 2DArray 3\*3 Size

class \_TwoDArray

{

//static void Main(string[] args)

public void run()

{

string[,] \_matrix = new string[,]

{

{"AElement1","AElement2","AElement3" },

{"BElement1","BElement2","BElement3" },

{"CElement1", "CElement2","CElement3"}

};

for (int i = 0; i <= \_matrix.GetUpperBound(0); i++)

{

Console.WriteLine("{0} {1} {2}", \_matrix[i, 0], \_matrix[i, 1], \_matrix[i, 2]);

}

}

}

#endregion

#region MulitpleInheritance

class CS1\_MulitpleInheritance : Ibase1 , Ibase2

{

public void message()

{

Console.WriteLine("Example Message from : Multiple Inheritance!..");

}

}

public interface Ibase1

{

void message();

}

public interface Ibase2

{

void message();

}

#endregion

#region AbstractVirtualOverride

abstract class \_abstractBaseClass

{

public int Number = 100;

public abstract void sum();

}

class childClass : \_abstractBaseClass

{

public override void sum()

{

Console.WriteLine("Overrided Sum value :{0}", (Number + Number));

}

}

#endregion

#region Delegate Instances

class Math

{

public List<int> findAllDivisibleBy3(List<int> numlist)

{

List<int> result = numlist.FindAll((int i) => { return i % 3 == 0; });

foreach (var item in result)

{

Console.Write(item + ",");

}

return result;

}

}

#endregion

#region Extend String by IsEmail RegEx

public static class commonUtilities

{

public static bool isEMail(string input)

{

var result = Regex.Match(input, @"\w+([-+.']\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*", RegexOptions.IgnoreCase);

return result.Success;

}

}

#endregion

}

ScreenShot:

