using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System;

#region Example 1

//using Outer.Midlle.Inner;

//using Fruits;

//using CustomConsole;

//namespace CustomConsole

//{

// public class Console

// {

// public static void WriteLine(string text = "")

// {

// System.Console.WriteLine(text);

// }

// }

//}

////Namespaces

//namespace ConsoleApp2

//{

// public class Program

// {

// static void Main(string[] args)

// {

// System.Console.WriteLine("Salam {0}", "Salam");

// CustomConsole.Console.WriteLine("Salam ");

// }

// }

//}

//namespace Fruits

//{

// class Apple

// {

// public void ShowDetails()

// {

// }

// }

// class Mango

// {

// }

//}

//namespace MyProject

//{

// class Program

// {

// static void Main(string[] args)

// {

// var apple = new Fruits.Apple();

// }

// }

//}

#endregion

#region Example 2

//namespace Outer

//{

// namespace Midlle

// {

// class Samsung

// {

// public Samsung()

// {

// }

// }

// namespace Inner

// {

// class A { }

// class B

// {

// public B()

// {

// }

// }

// }

// }

// class Apple

// {

// }

//}

//namespace Network

//{

// namespace HTTP

// {

// namespace HTTPClient

// {

// class HttpClientResponse

// {

// }

// }

// }

//}

//namespace Project

//{

// using NHC = Network.HTTP.HTTPClient;

// public class Program

// {

// static void Main(string[] args)

// {

// NHC.HttpClientResponse httpResponse = new NHC.HttpClientResponse();

// // var obj=new B();

// }

// }

//}

#endregion

#region Overriding

//namespace MyNamespace

//{

// class Student

// {

// public string Name { get; set; }

// public string Surname { get; set; }

// public int Age { get; set; }

// public override string ToString()

// {

// return $"Name : {Name} Surname : {Surname} Age : {Age}";

// }

// public override bool Equals(object obj)

// {

// if (obj is Student other)

// {

// return this.Name == other.Name && this.Age == other.Age && this.Surname == other.Surname;

// }

// return false;

// }

// }

//class Program

//{

// static void Main(string[] args)

// {

// Student student1 = new Student

// {

// Name = "John",

// Surname = "Johnlu",

// Age = 23

// };

// Student student2 = new Student

// {

// Name = "John",

// Surname = "Johnlu",

// Age = 23

// };

// Console.WriteLine(student1.Equals(student2));

// Console.WriteLine(Object.ReferenceEquals(student1,student2));

// //Console.WriteLine(student);

// }

// }

//}

#endregion

#region Operator Overloading

namespace MyNamespace

{

class Point

{

public int X { get; set; }

public int Y { get; set; }

public override string ToString()

{

return $"X : {X} Y : {Y}";

}

public override bool Equals(object obj)

{

if(obj is Point other)

{

return other.X == this.X && other.Y == this.Y;

}

return false;

}

}

class Program

{

static void Main(string[] args)

{

Point p1 = new Point

{

X = 10,

Y = 20

};

Point p2 = new Point

{

X = 10,

Y = 21

};

if (p1.Equals(p2))

{

Console.WriteLine("They are same");

}

else

{

Console.WriteLine("They are different");

}

}

}

}

#endregion