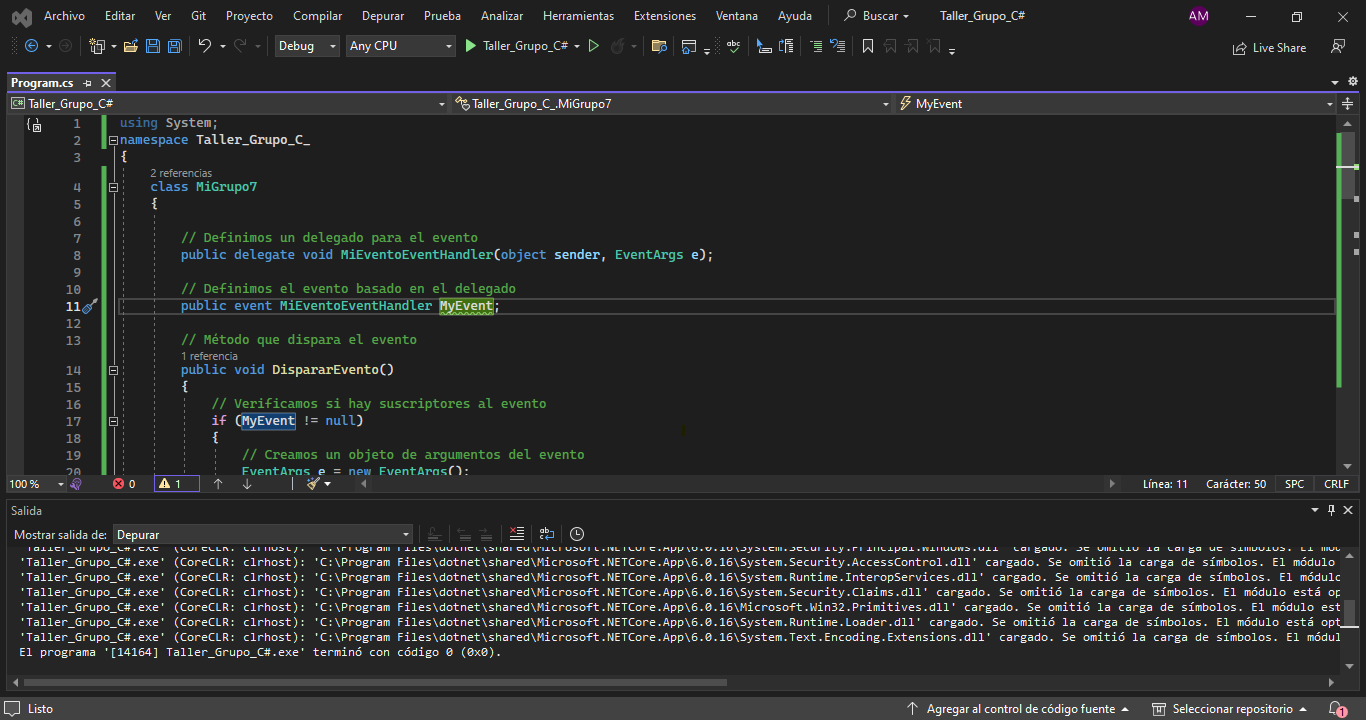
12-6-2023

Taller en Grupo: resolver un programa trabajando en C#

Ing. Mariuxi Tejada

ANTHONY ESTUARDO CORDOVA MORENO, KAISER CRESPIN CHAVEZ, JULIANA RAMIREZ SACARELO, PAULA CARDENAS SARES, YOSELYN VELEZ CAMPUSANO

universidad de guayaquil

Realizar en grupo un programa en C# que contenga eventos y excepciones. Texto

Descripción generada automáticamente

# Código

// ANTHONY ESTUARDO CORDOVA MORENO,

// KAISER CRESPIN CHAVEZ,

// JULIANA RAMIREZ SACARELO,

// PAULA CARDENAS SARES,

// YOSELYN VELEZ CAMPUZANO

// GRUPO 7

using System;

namespace Taller\_Grupo\_C\_

{

class MiGrupo7

{

// Definimos un delegado para el evento

public delegate void MiEventoEventHandler(object sender, EventArgs e);

// Definimos el evento basado en el delegado

public event MiEventoEventHandler MyEvent;

// Método que dispara el evento

public void DispararEvento()

{

// Verificamos si hay suscriptores al evento

if (MyEvent != null)

{

// Creamos un objeto de argumentos del evento

EventArgs e = new EventArgs();

// Disparamos el evento

MyEvent(this, e);

}

}

}

class Program

{

static void Main(string[] args)

{

try

{

// Creamos una instancia de la clase

MiGrupo7 miObjeto = new MiGrupo7();

// Suscribimos un método al evento

miObjeto.MyEvent += MiMetodo;

// Disparamos el evento

miObjeto.DispararEvento();

}

catch (Exception ex)

{

Console.WriteLine("Se ha producido una excepción: " + ex.Message);

}

}

// Método que se ejecuta cuando se dispara el evento

static void MiMetodo(object sender, EventArgs e)

{

Console.WriteLine("El evento ha sido disparado.");

}

public override bool Equals(object? obj)

{

return base.Equals(obj);

}

public override int GetHashCode()

{

return base.GetHashCode();

}

public override string? ToString()

{

return base.ToString();

}

}

}