

PROGRAMACIÓN ORIENTADA A OBJETOS



PROFESOR: VILLALOBOS
MARTÍNEZ RODOLFO



ALUMNO: ORTIS CASTRO JUAN
MANUEL



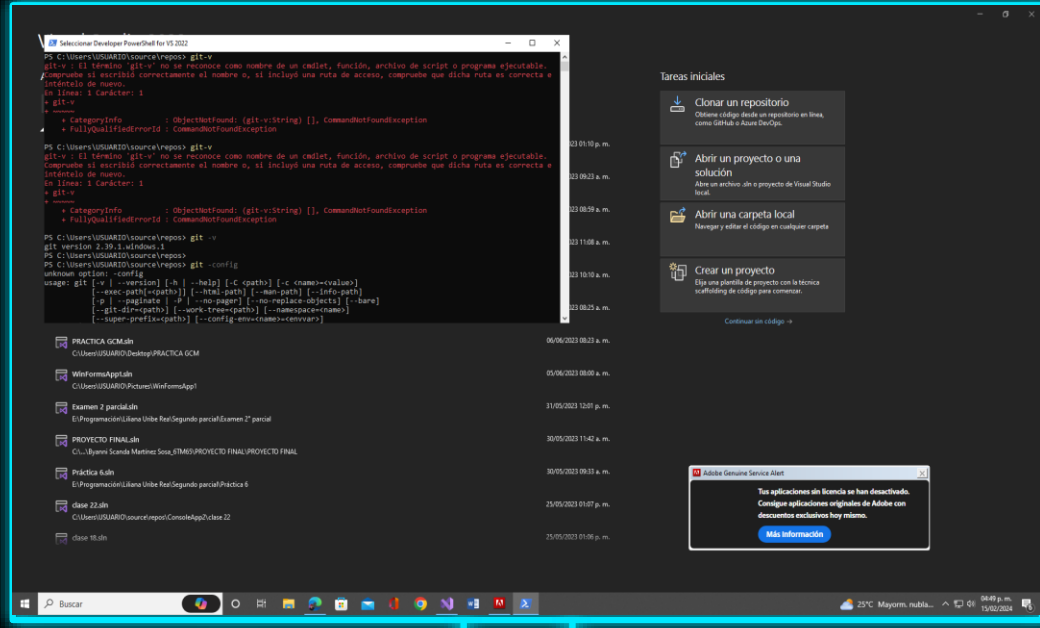
GRUPO: 6TV65

CLASE

01

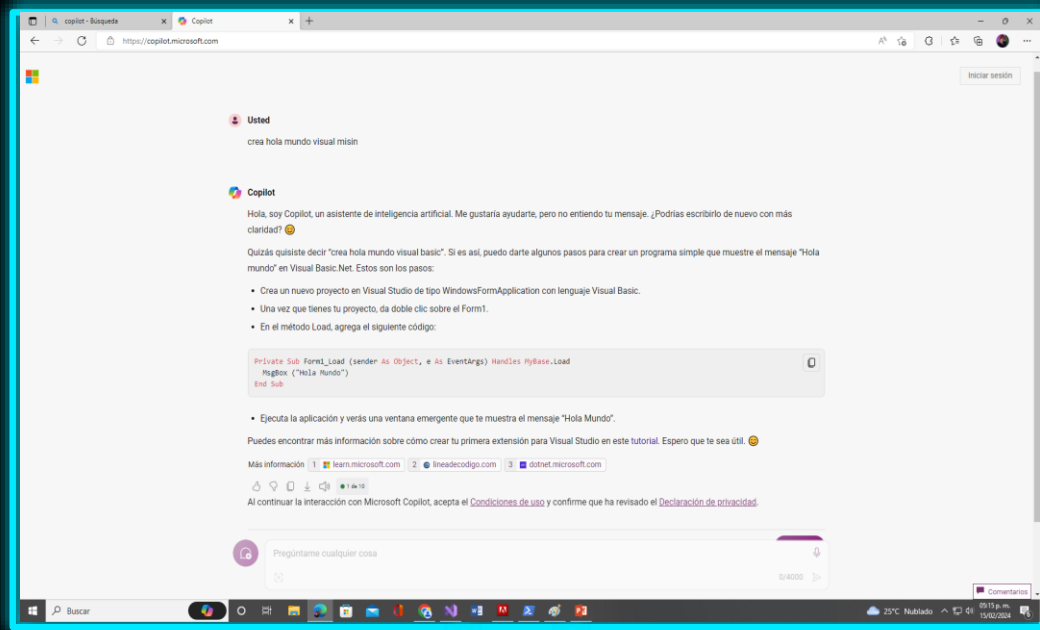
INTRODUCCIÓN

CLASE 1: REVISIÓN DE CONFIGURACIÓN

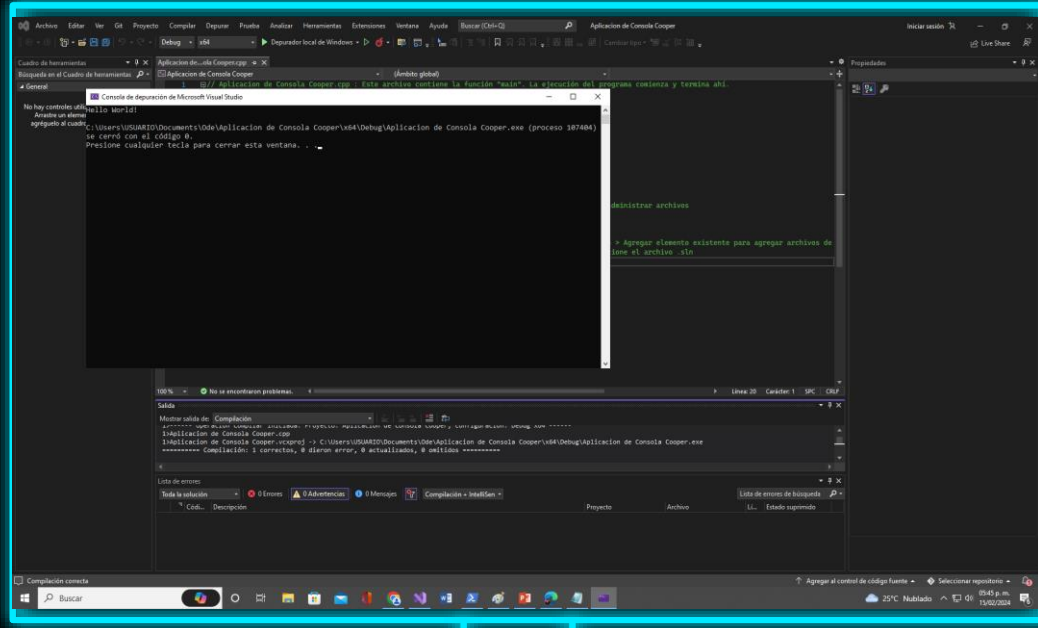


- Utilizando el comando `git -v` con dio la versión de git instalada
- Con `git config -L` revisamos el nombre de usuario y email

CLASE 1: REVISIÓN DE CONFIGURACIÓN



CLASE 1: REVISIÓN DE CONFIGURACIÓN



CLASE 1: REVISIÓN DE CONFIGURACIÓN

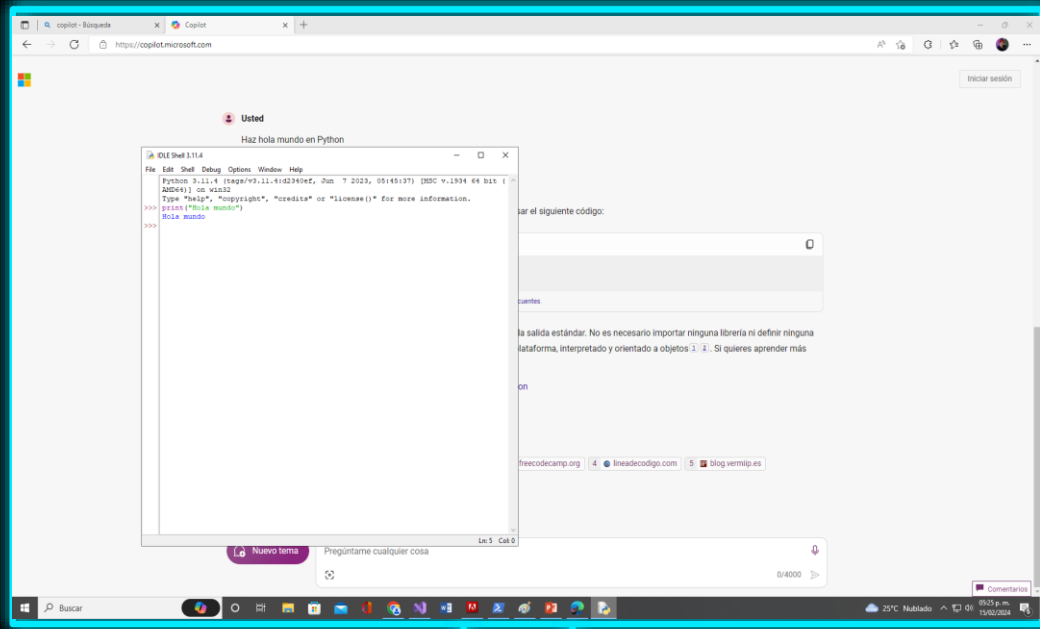


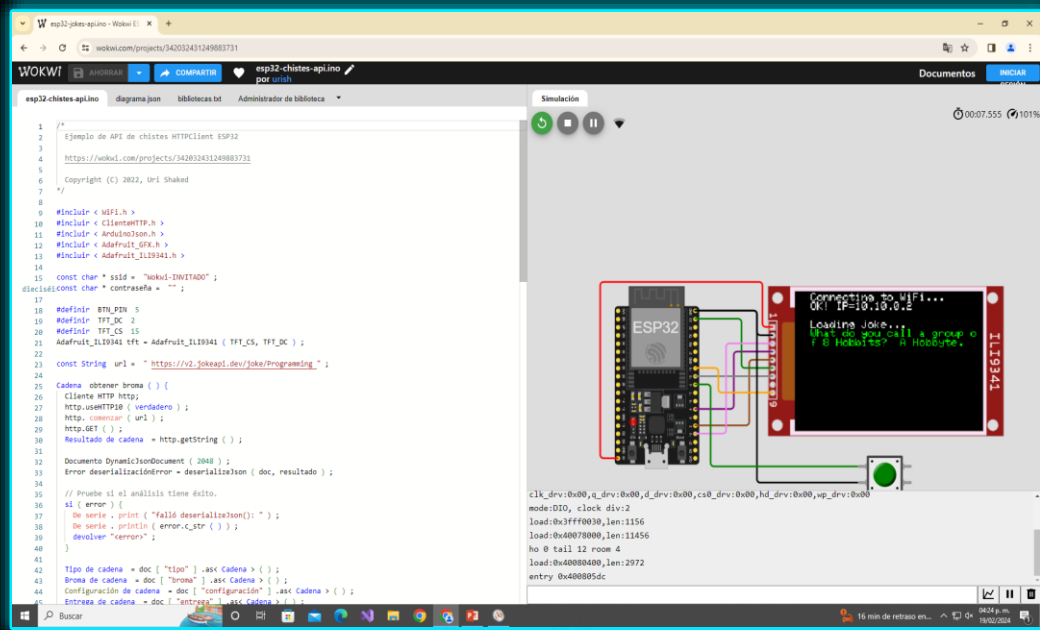
Figure 1



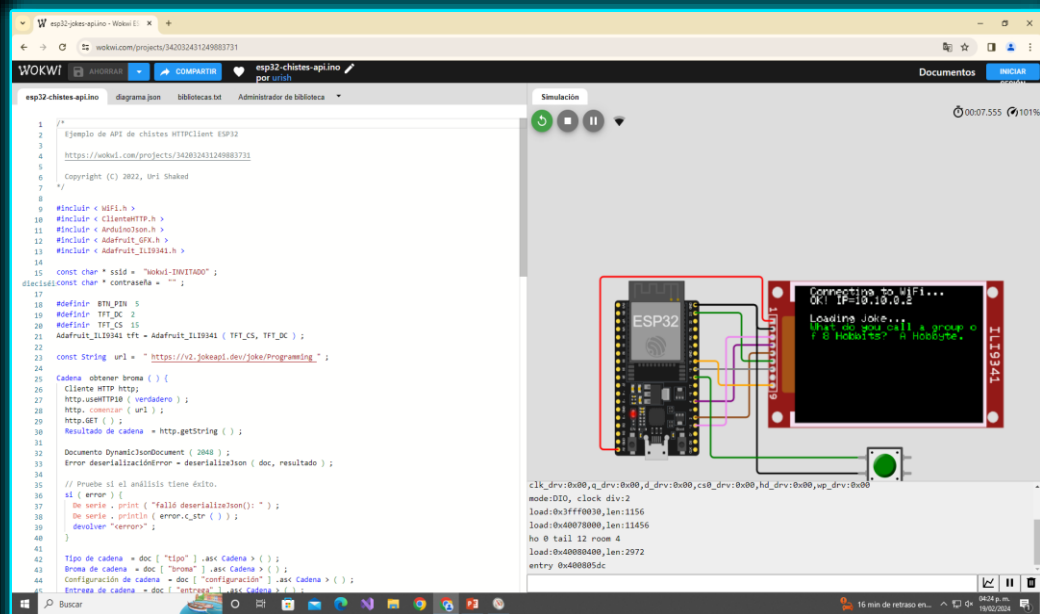
CLASE

02

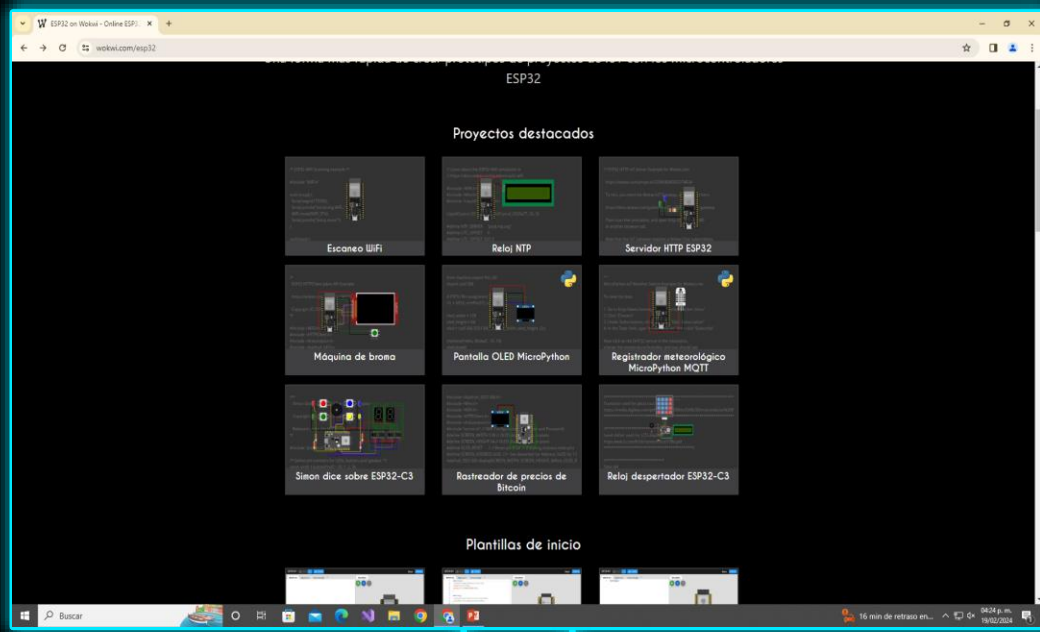
CLASE 2:



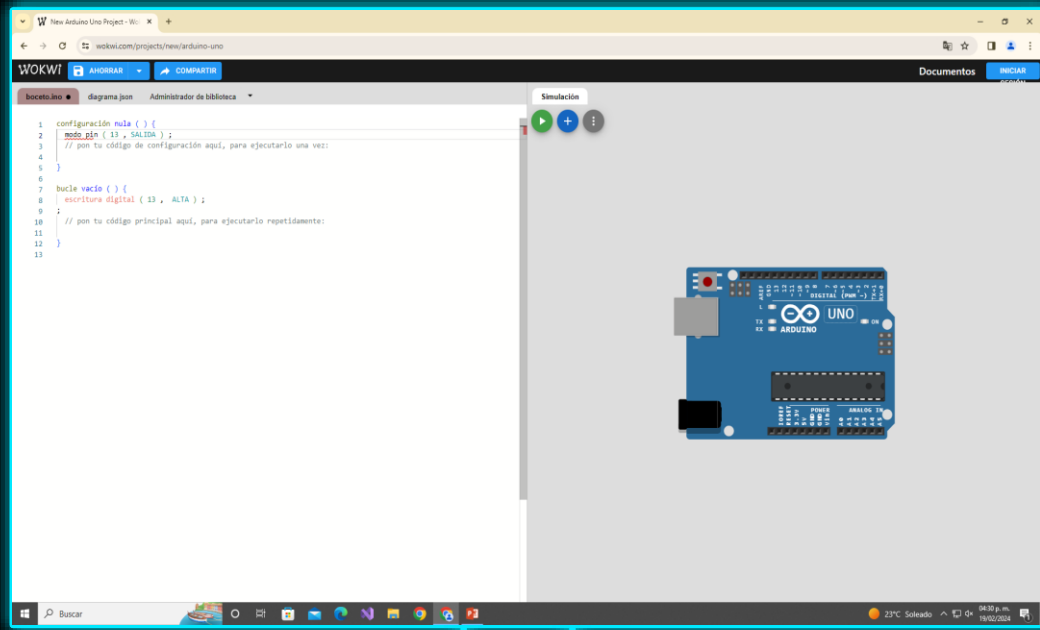
CLASE 2:



CLASE 2:



CLASE 2:





Nota: Crear proyecto (Carpeta)



Crear la carpeta



Crear archivo (nombre).CPP



Usar Mc comando cl



Cl (nombre).cpp

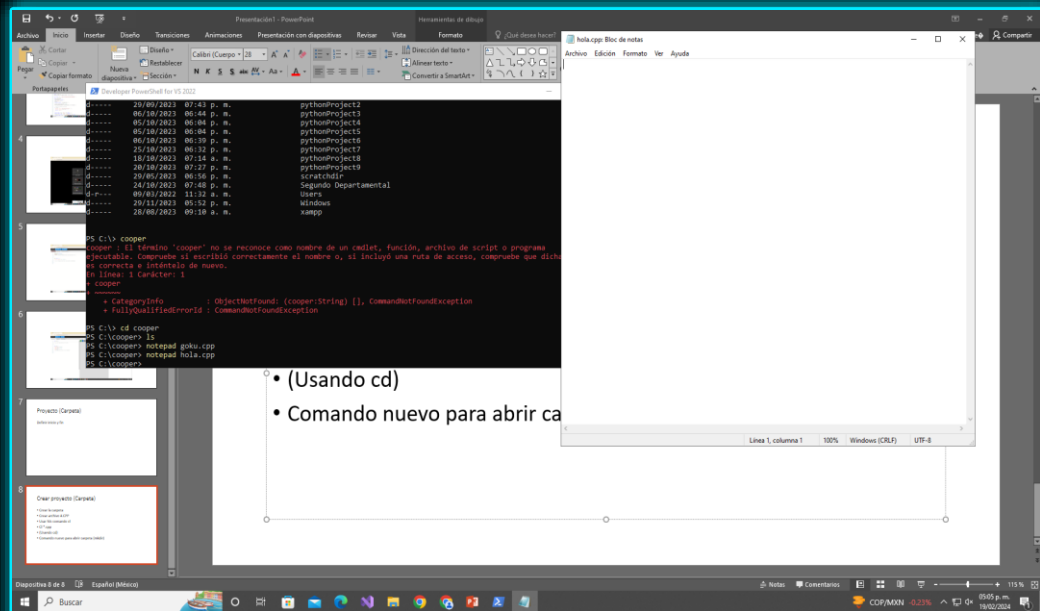


(Usando cd)

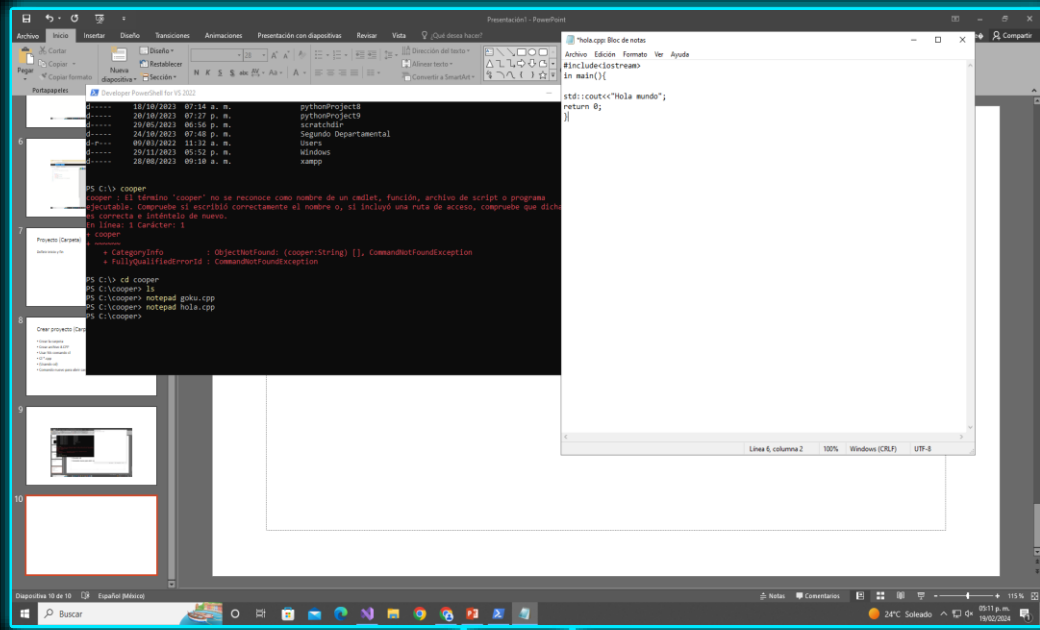


Comando nuevo para abrir carpeta (***mkdir***)

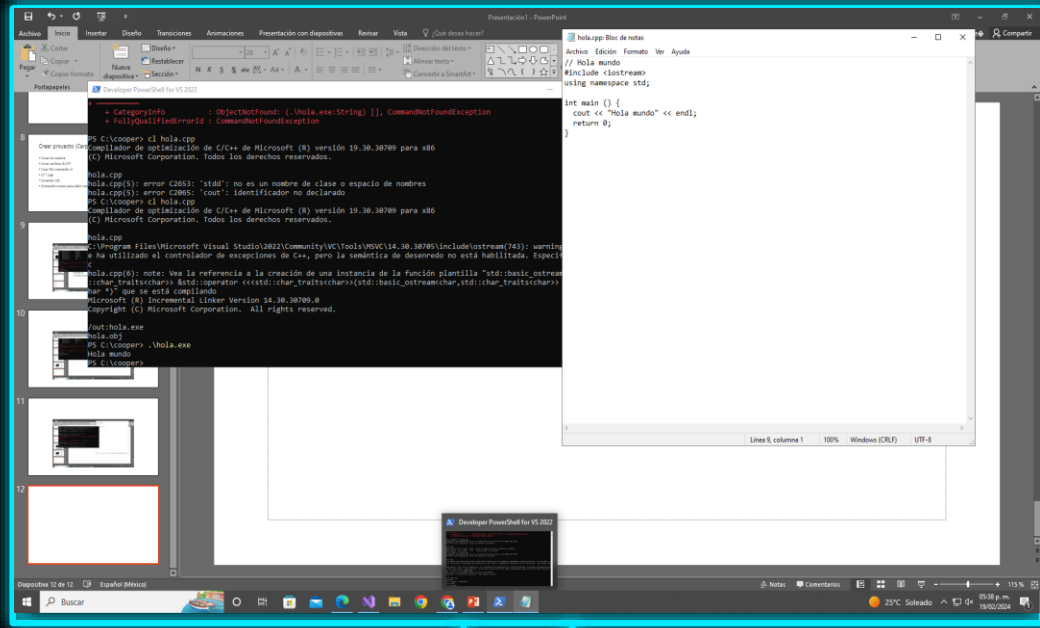
CLASE 2:



CLASE 2:



CLASE 2:



CLASE

03

CLASE 3

The screenshot shows a computer monitor with a web browser and a terminal window. The browser displays a page titled 'Aprende Git Branching' with a sidebar containing a list of tasks: 'level intro3', 'hint', 'Asegúrate de hacer commit en el orden especificado (bugfix antes de main)', 'delay 2000', and 'show goal'. The main content area shows a Git branching diagram with branches 'main' and 'bugfix' and a 'bugfix' label. The terminal window shows the same list of tasks. The browser also displays a 'Objetivo a cumplir' (Goal to achieve) section with a diagram showing the flow of branches: 'main' to 'bugfix' to 'main' to 'bugfix' to 'main'.

Browser tabs: JSORJS - the JavaScript origins, WhatsApp, Learn Git Branching, GitHub

Browser address bar: learninggitbranching.js.org/locales-es_ES

Browser title: Aprende Git Branching

Browser sidebar:

- Understand
- Level: Aprende merge en Git
- level intro3
- hint
- Asegúrate de hacer commit en el orden especificado (bugfix antes de main)
- delay 2000
- show goal

Browser main content:

Objetivo a cumplir

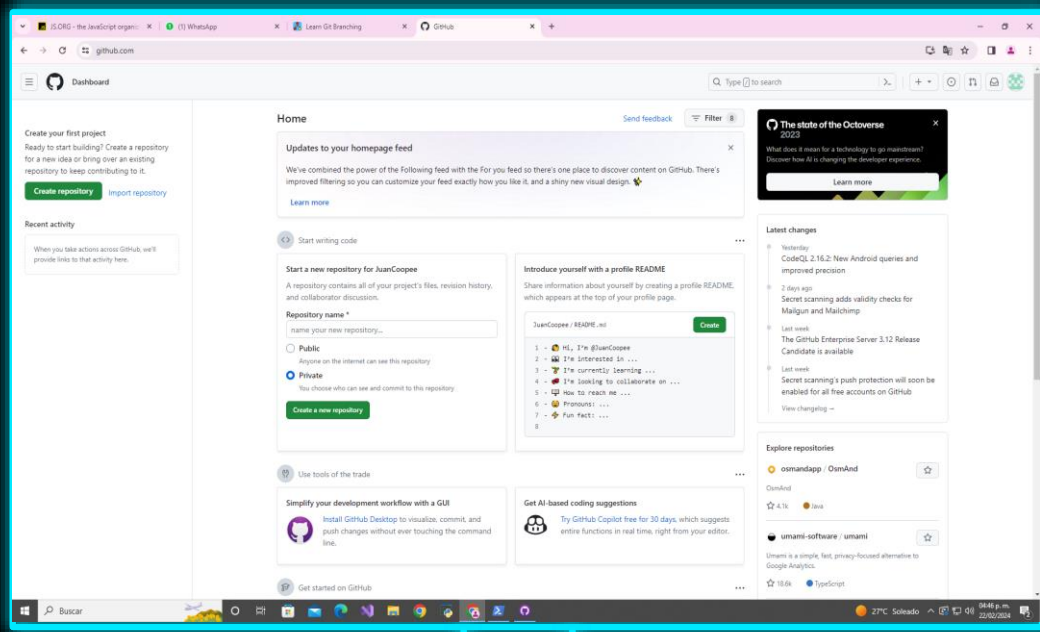
Podrás visualizar esta pantalla con "Node goal"

Diagram showing the flow of branches: main, bugfix, main, bugfix, main

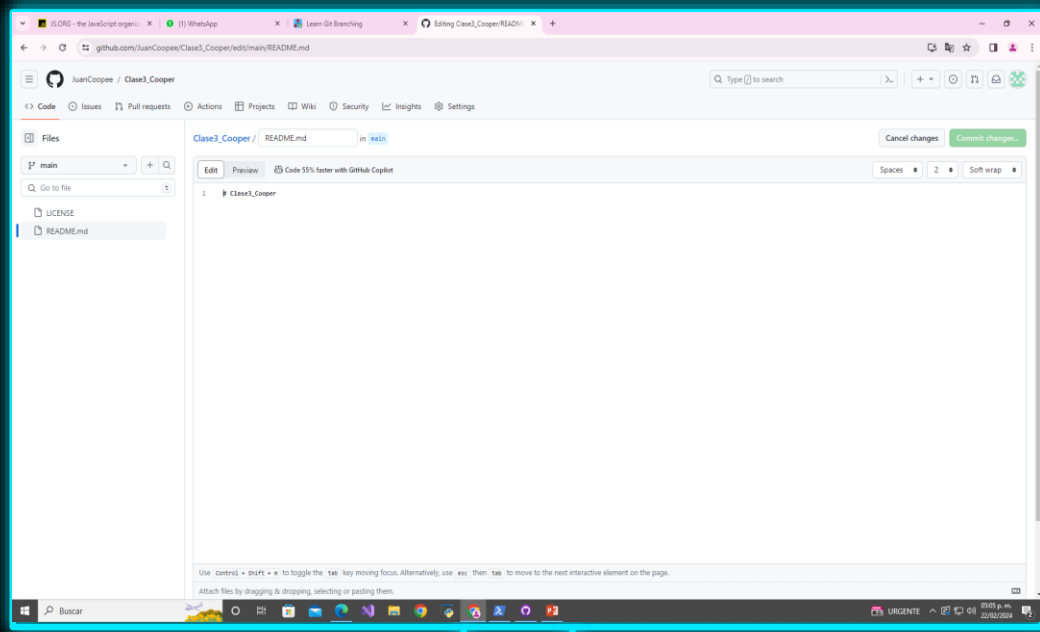
Terminal window:

```
$ level intro3
$ hint
Asegúrate de hacer commit en el orden especificado (bugfix antes de main)
$ delay 2000
$ show goal
```

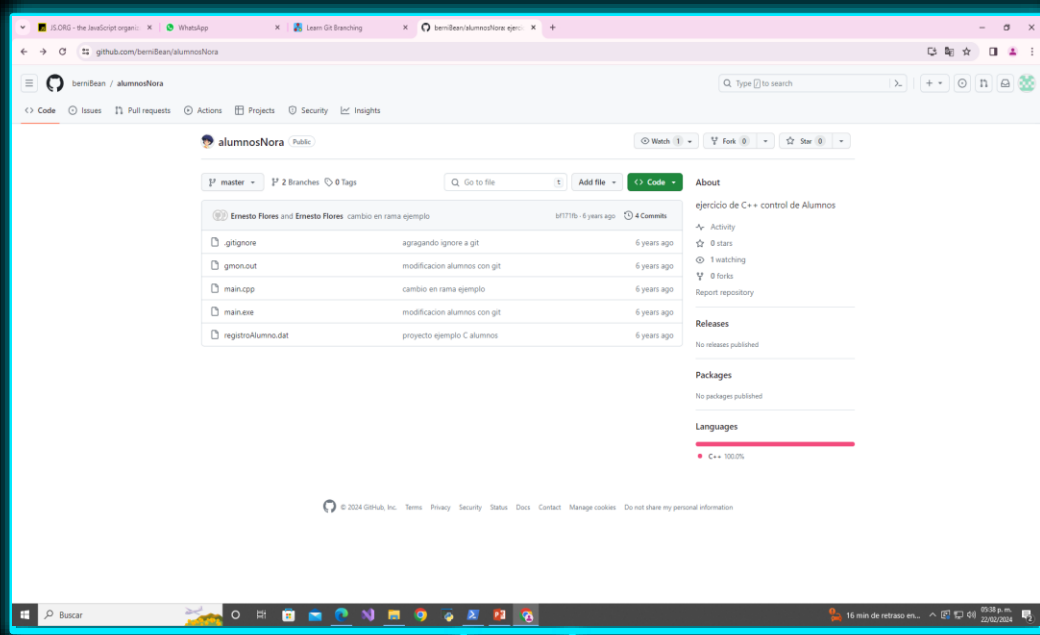
CLASE 3



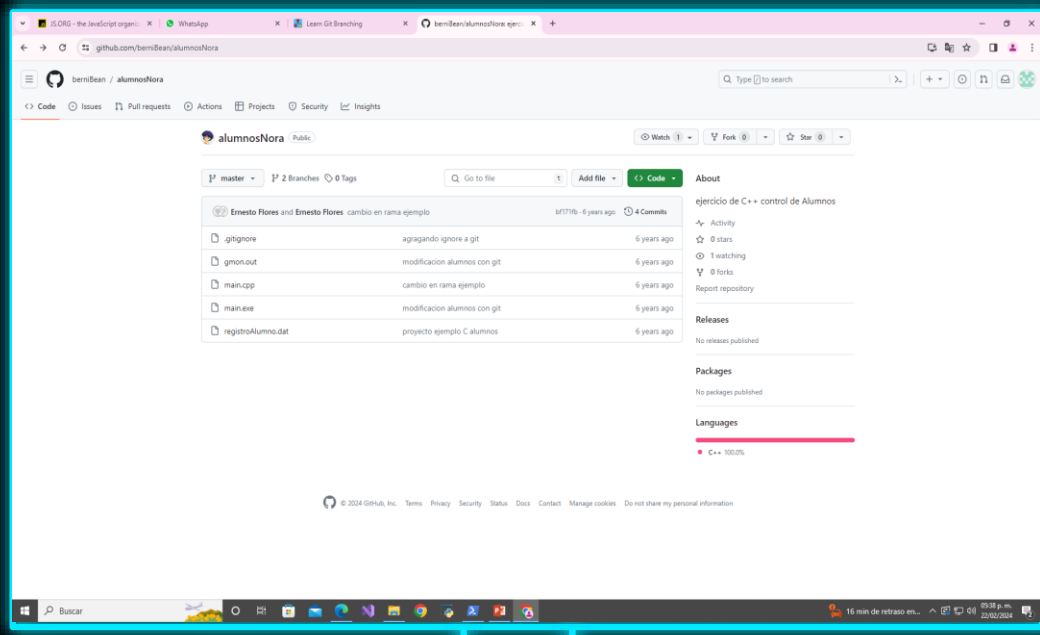
CLASE 3



CLASE 3:



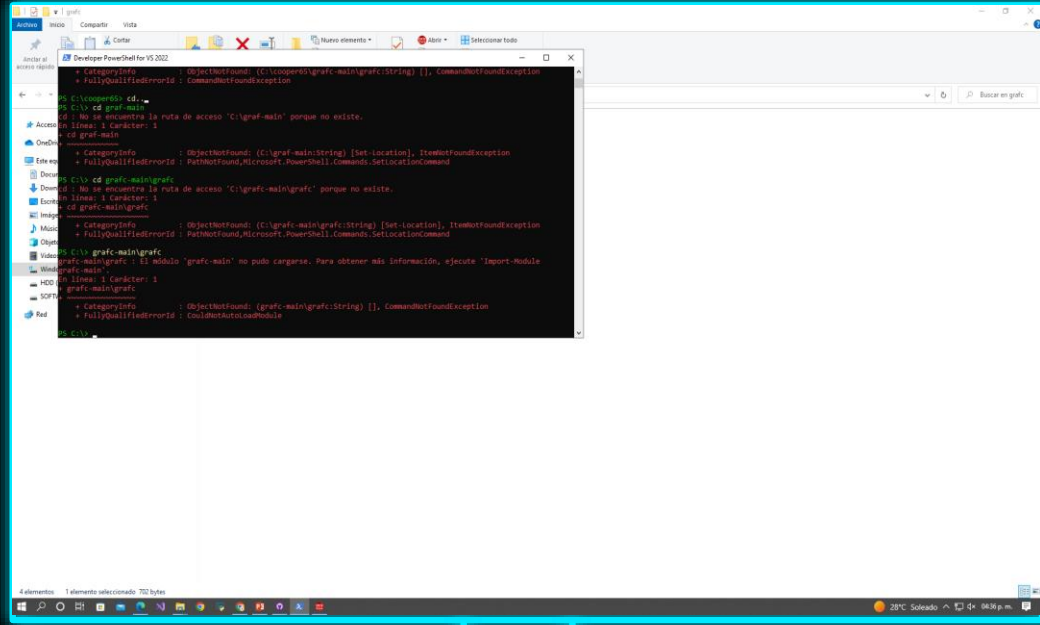
CLASE 3



CLASE

04

CLASE 4



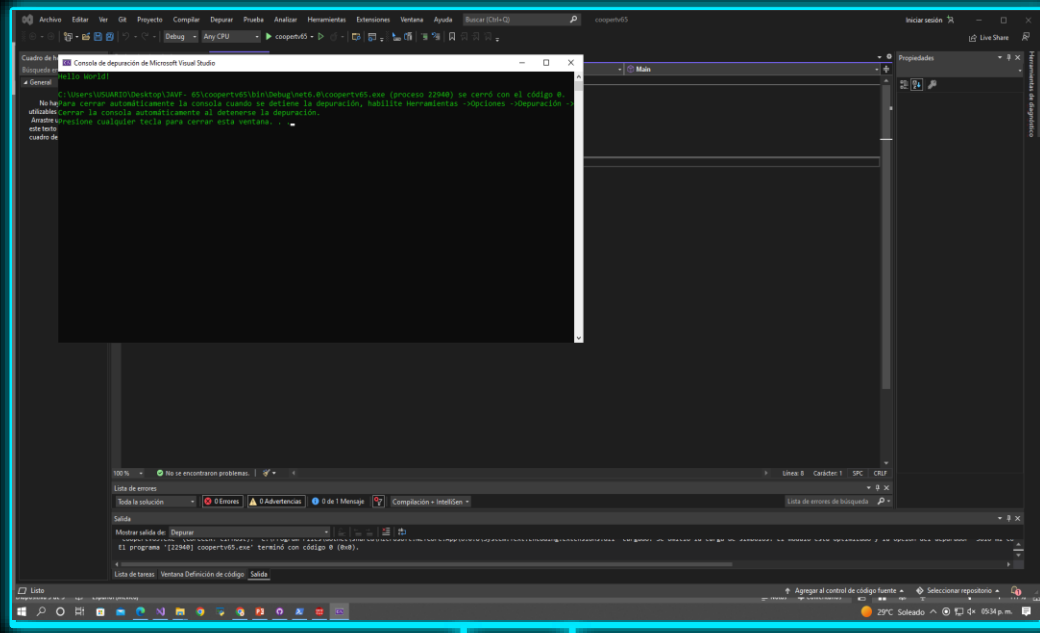
```
PS C:\> cd .
PS C:\> cd graf-main
cd : No se encuentra la ruta de acceso 'C:\graf-main' porque no existe.
En línea: 1 Carácter: 1
+ cd graf-main
+ ~~~~~
+ CategoryInfo          : (ObjectNotFound: (C:\graf-main:String) [Set-Location], ItemNotFoundException)
+ FullyQualifiedErrorId : PathNotFound,Microsoft.PowerShell.Commands.SetLocationCommand

PS C:\> cd graf-main\grafc
cd : No se encuentra la ruta de acceso 'C:\graf-main\grafc' porque no existe.
En línea: 1 Carácter: 1
+ cd graf-main\grafc
+ ~~~~~
+ CategoryInfo          : (ObjectNotFound: (C:\graf-main\grafc:String) [Set-Location], ItemNotFoundException)
+ FullyQualifiedErrorId : PathNotFound,Microsoft.PowerShell.Commands.SetLocationCommand

PS C:\> grafc-main\grafc
grafc-main\grafc : El módulo 'grafc-main' no pudo cargarse. Para obtener más información, ejecute 'Import-Module
grafc-main'.
En línea: 1 Carácter: 1
+ grafc-main\grafc
+ ~~~~~
+ CategoryInfo          : (ObjectNotFound: (grafc-main\grafc:String) [], CommandNotFoundException)
+ FullyQualifiedErrorId : CouldNotAutoLoadModule

PS C:\>
```


CLASE 4



CLASE 4

The screenshot shows a web browser displaying the GitHub repository page for 'Clase3_Cooper' by user 'JuanCooper'. The browser's address bar shows the URL 'github.com/JuanCooper/Clase3_Cooper'. The repository page includes a navigation bar with links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The repository name 'Clase3_Cooper' is displayed as 'Public'. Below the repository name, there are buttons for 'Pin', 'Unwatch', 'Fork', and 'Star'. The main content area shows a list of commits, with the most recent one being 'Add files via upload' by 'JuanCooper' 2 minutes ago. The commit details show files 'LICENSE', 'Program.vb', 'README.md', and 'clase 3.pptx'. The 'README' section is expanded, showing the repository name 'Clase3_Cooper', the user 'Coopeer Good', the location 'Hala Madrid', and a description: 'Este proyecto es para sacar 10 y norecurarla esto fue hecho por Cooper del grupo 6TV'. The right sidebar contains sections for 'About', 'Releases', and 'Packages', all indicating no content is published yet.

Resumen Puebla 0-2 Querétaro

JuanCooper/Clase3_Cooper

github.com/JuanCooper/Clase3_Cooper

JuanCooper / Clase3_Cooper

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Clase3_Cooper Public

Pin Unwatch Fork Star

main 1 Branch 0 Tags

Go to file Add file Code

About

No description, website, or topics provided.

Readme

Unlicense license

Activity

0 stars

1 watching

0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Commits

Commit	Author	Time
3d00def	JuanCooper	2 minutes ago
Initial commit	JuanCooper	4 days ago
Add files via upload	JuanCooper	2 minutes ago
Update README.md se me olvidó los espacios	JuanCooper	4 days ago
Subí mi archivo de PPT	JuanCooper	4 days ago

README Unlicense license

Clase3_Cooper

Coopeer Good

Hala Madrid

Este proyecto es para sacar 10 y norecurarla esto fue hecho por Cooper del grupo 6TV

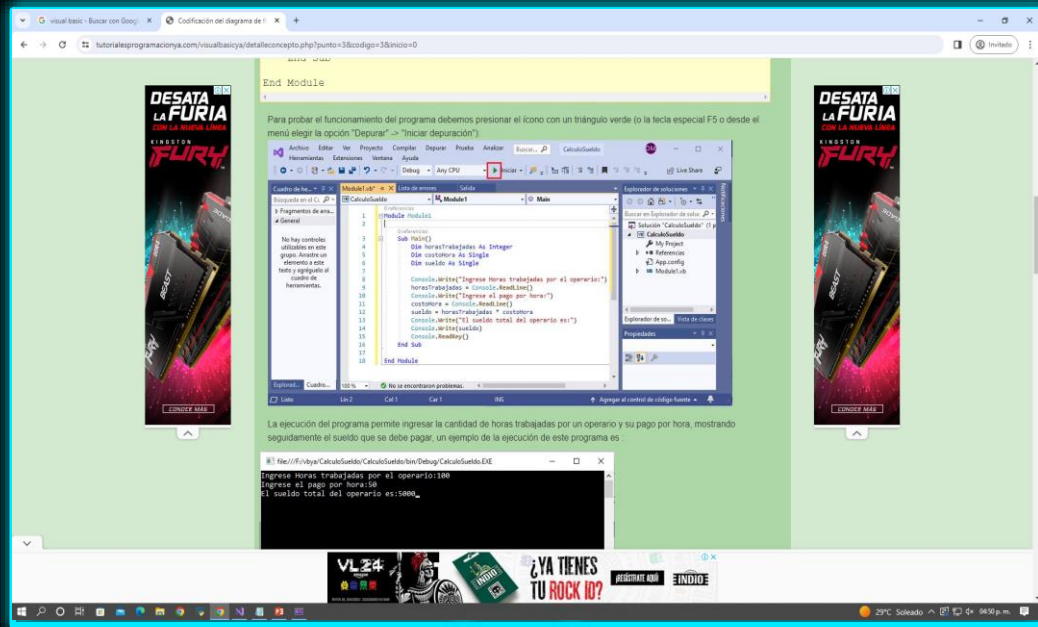
© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

29°C Soleado 6:40 p.m.

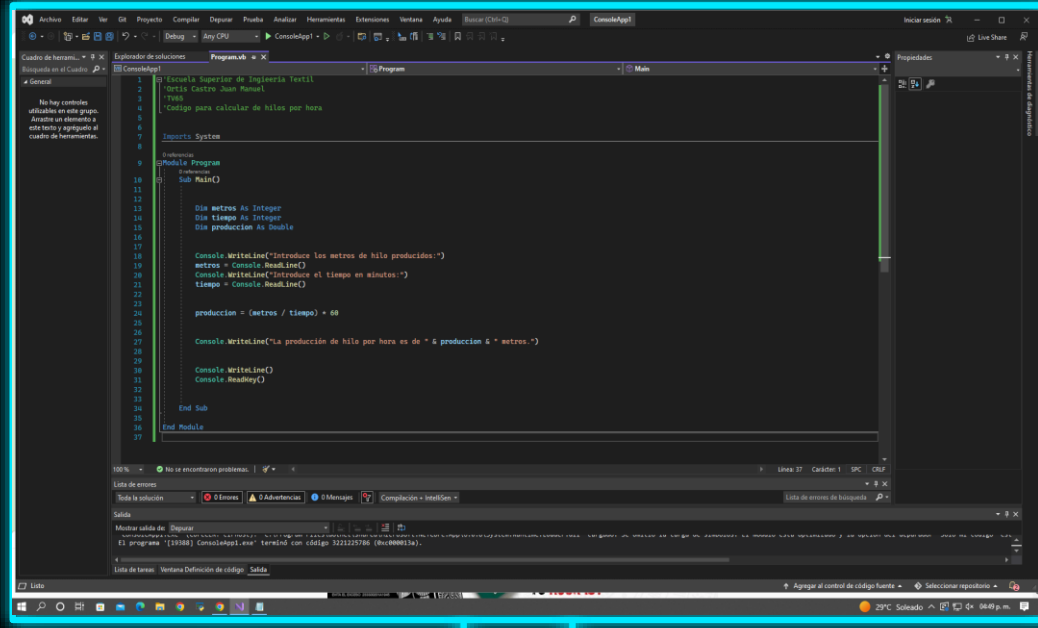
CLASE

05

CLASE 5



CLASE 5



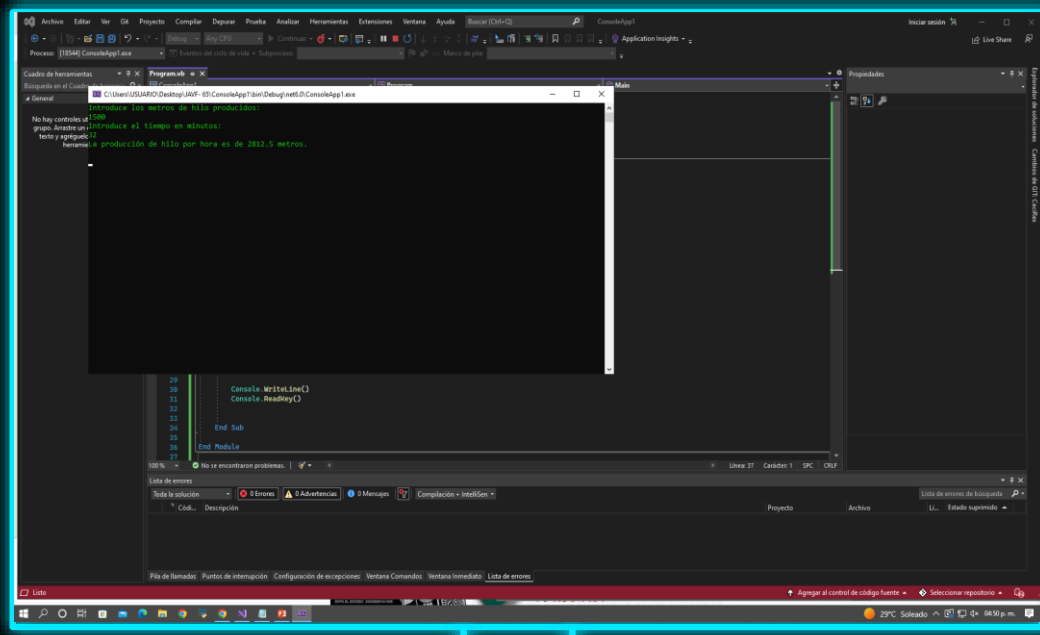
The screenshot shows the Visual Studio Code IDE with a C# project named 'ConsoleApp1'. The code is written in Spanish and calculates the production of threads in meters per hour based on input from the user.

```
1 Escuela Superior de Ingeniería Textil
2 Juan Castro Juan Manuel
3 1985
4 Código para calcular de hilos por hora
5
6
7 Imports System
8
9 namespace Program
10 {
11     class MainClass
12     {
13     public Main()
14     {
15         Dia metros As Integer
16         Dia tiempo As Integer
17         Dia produccion As Double
18
19         Console.WriteLine("Introduce los metros de hilo producidos:")
20         metros = Console.ReadLine()
21         Console.WriteLine("Introduce el tiempo en minutos:")
22         tiempo = Console.ReadLine()
23
24         produccion = (metros / tiempo) * 60
25
26         Console.WriteLine("La producción de hilo por hora es de " & produccion & " metros.")
27
28         Console.WriteLine()
29         Console.ReadLine()
30         Console.ReadKey()
31     }
32     End Sub
33 }
34 End Module
```

The bottom of the window shows the 'Output' window with the following text:

```
Mostrar salida de: Debug
El programa "C:\Program Files\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.29.30133\bin\Hostx64\x64\cl.exe" se ejecutó correctamente. El resultado de la compilación es: 0. El programa "C:\Program Files\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.29.30133\bin\Hostx64\x64\cl.exe" se ejecutó correctamente. El resultado de la compilación es: 0. El programa "C:\Program Files\Microsoft Visual Studio\2019\Community\VC\Tools\MSVC\14.29.30133\bin\Hostx64\x64\cl.exe" se ejecutó correctamente. El resultado de la compilación es: 0.
```

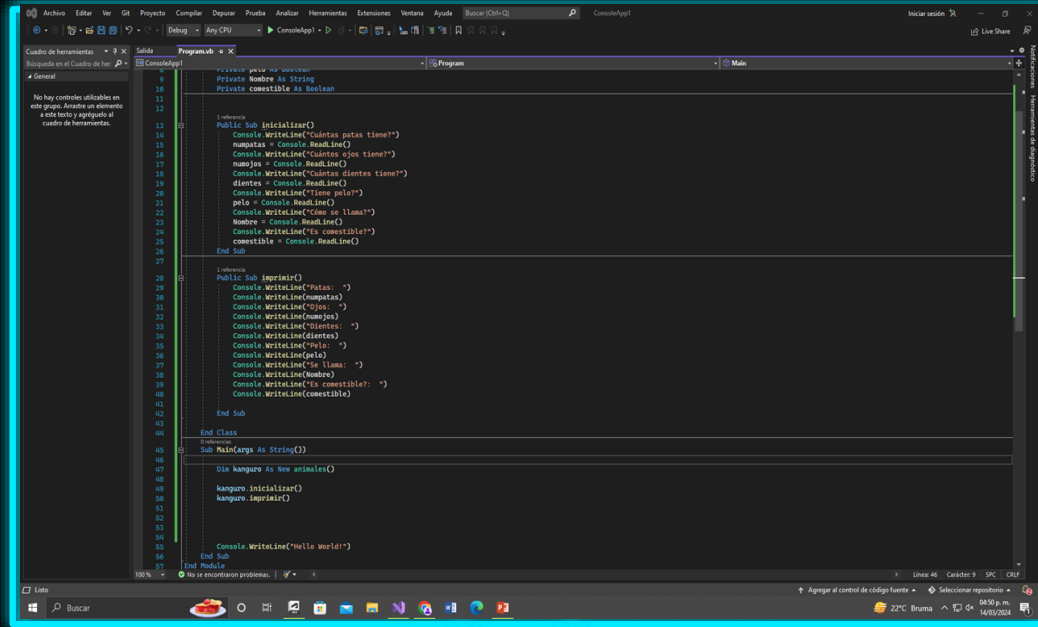
CLASE 5



CLASE

07

CLASE 7



The screenshot shows a Visual Studio Code editor window with a C# program. The program defines a `Kangaro` class with properties for `patas`, `ojos`, `dientes`, `pelu`, `llama`, `comestible`, and `comestible?`. It includes methods for `Inicializar` and `Imprimir`. The `Main` method creates a `Kangaro` object and calls `Inicializar` and `Imprimir`.

```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
        }
    }
}
```

The code in the image is as follows:

```
using System;

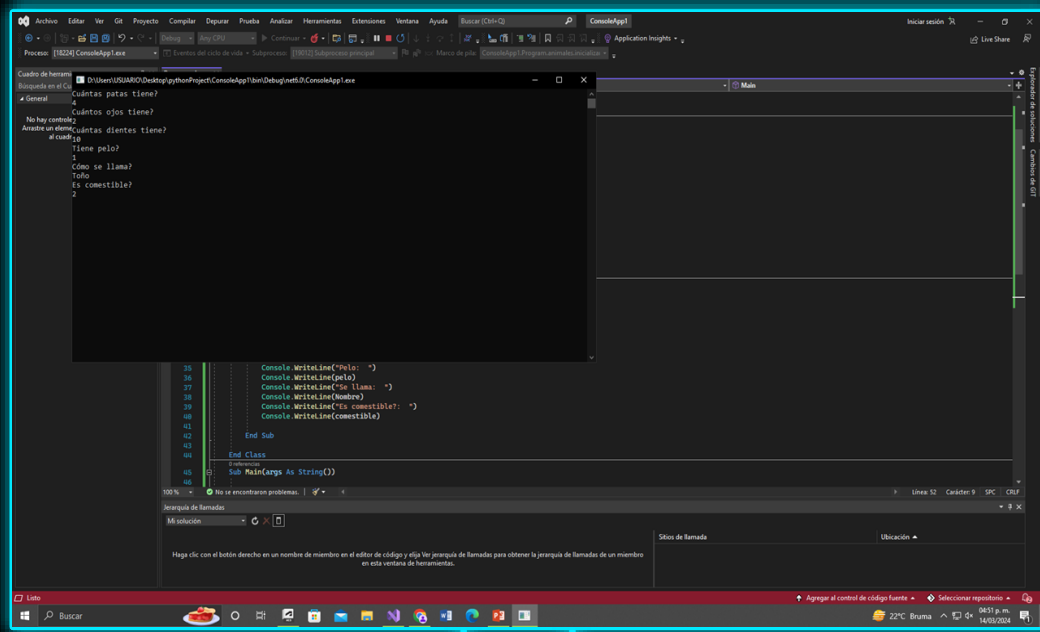
namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
        }
    }
}
```

The code in the image is as follows:

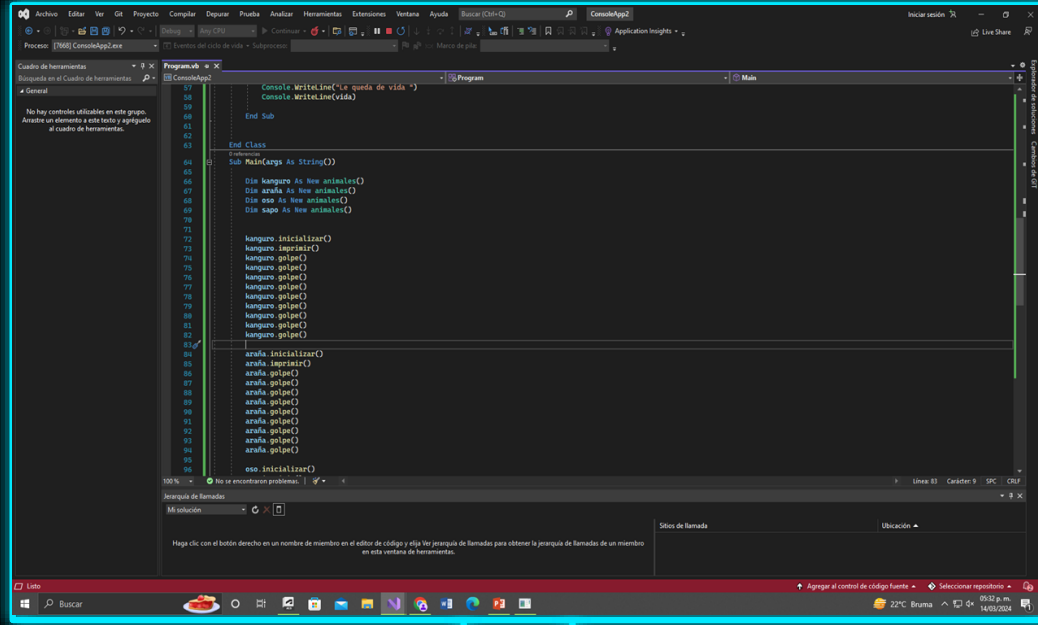
```
using System;

namespace ConsoleApp1
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
        }
    }
}
```


CLASE 7



CLASE 7



```
1 Console.WriteLine("¿A qué vida de vida?")
2 Console.WriteLine(vida)
3
4 End Sub
5
6 End Class
7
8 Private Sub Main(args As String())
9     Dim kangaro As New animales()
10    Dim araña As New animales()
11    Dim sapo As New animales()
12
13    kangaro.inicializar()
14    kangaro.imprimir()
15    kangaro.golpe()
16    kangaro.golpe()
17    kangaro.golpe()
18    kangaro.golpe()
19    kangaro.golpe()
20    kangaro.golpe()
21    kangaro.golpe()
22
23    araña.inicializar()
24    araña.imprimir()
25    araña.golpe()
26    araña.golpe()
27    araña.golpe()
28    araña.golpe()
29    araña.golpe()
30    araña.golpe()
31    araña.golpe()
32
33    sapo.inicializar()
34
35 End Sub
```

100% No se encontraron problemas.

Jerarquía de llamadas

MéTODO

Haga clic con el botón derecho en un nombre de miembro en el editor de código y elija Ver jerarquía de llamadas para obtener la jerarquía de llamadas de un miembro en esta ventana de herramientas.

Símbolos de llamada

Ubicación

Windows Taskbar: Lito, Buscar, 22°C Bruma, 09:32 p.m., 14/09/2024