

# COMO CREAR UN PROYECTO EN C#?

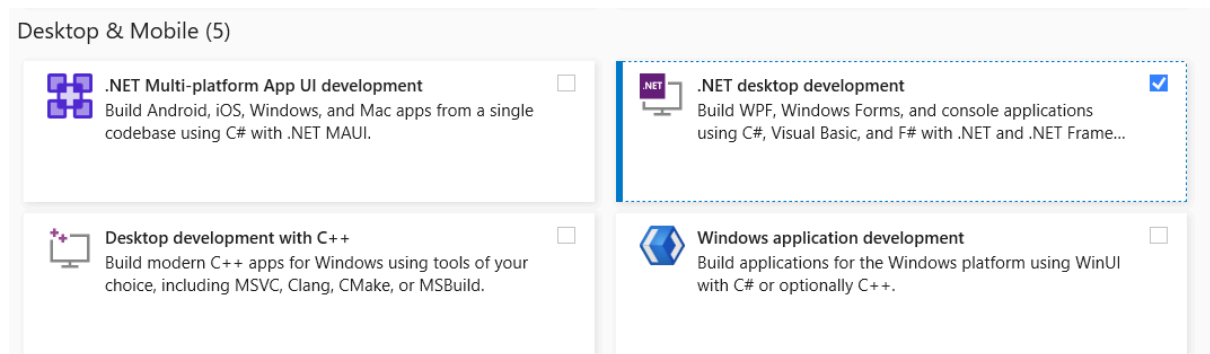
## #1 - Visual Studio

<https://visualstudio.microsoft.com/es/>



Seleccionamos la opción Community 2022.

Una vez que se instaló nuestra versión de VS, instalamos .NET.



A la derecha se mostrará el siguiente listado de dependencias:

## Installation details

- Visual Studio core editor ●
- ▼ .NET desktop development ●
  - ▼ Included
    - ✓ .NET desktop development tools
    - ✓ .NET Framework 4.7.2 development tools
    - ✓ C# and Visual Basic
  - ▼ Optional
    - ☒ Development tools for .NET
    - ☒ .NET Framework 4.8 development tools
    - ☒ Entity Framework 6 tools
    - ☒ .NET profiling tools
    - ☒ IntelliCode
    - ☒ Just-In-Time debugger
    - ☒ Live Share
    - ☒ ML.NET Model Builder
    - ☒ GitHub Copilot
    - ☒ Blend for Visual Studio
    - ☐ F# desktop language support
    - ☐ PreEmptive Protection - Dotfuscator

Importante: Si buscas una instalación más ligera, vamos a instalar solamente marcando las casillas: Development tools for .NET, .NET Framework 4.8 development tools, Entity Framework 6 tools, Just-In-Time debugger y Live Share.

## Visual Studio Installer

Installed Available



Visual Studio Community 2022

Pause

Downloading and verifying: 1.4 GB of 1.7 GB ( 6 MB/sec )  
81%

Installing: package 226 of 381

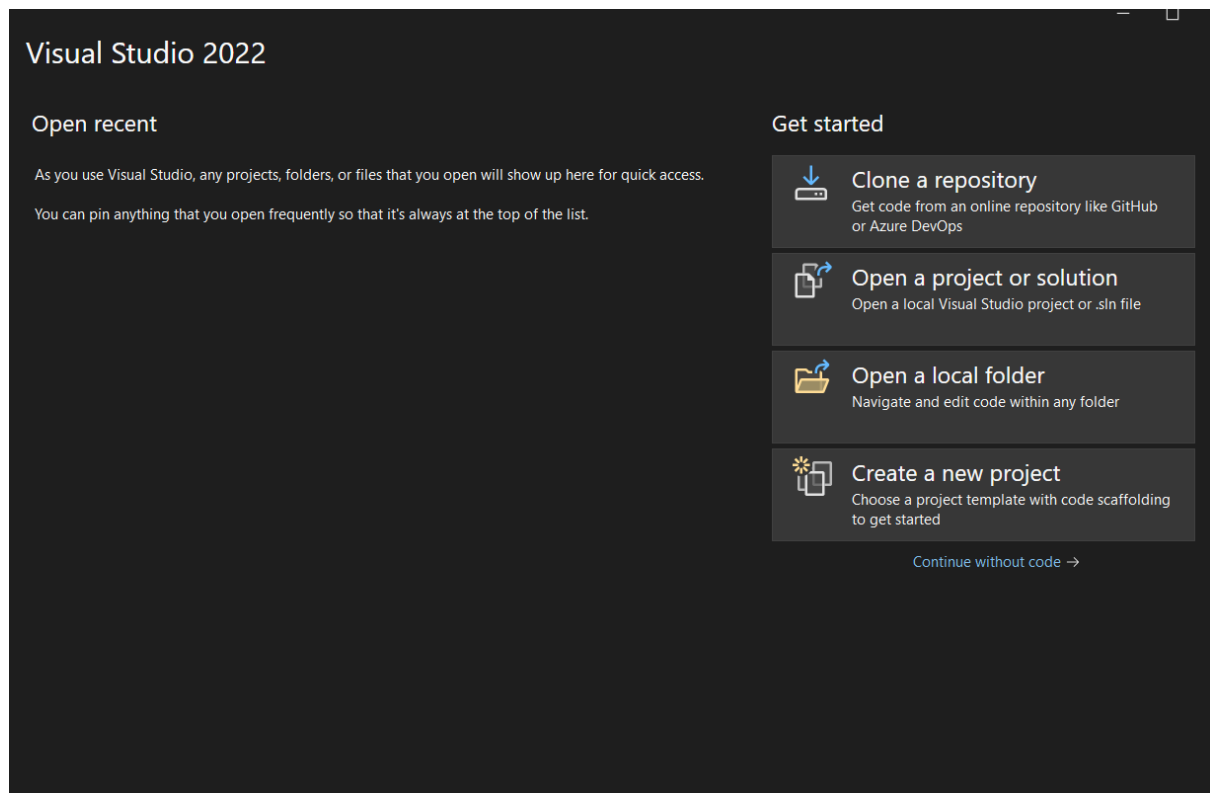
54%

Microsoft.VisualStudio.TestTools.TestPlatform.IDE

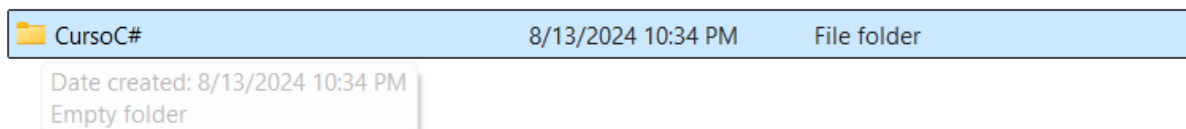
[Release notes](#)

Una vez que termine la instalación, vamos a “lanzar” nuestra aplicación. Nos pedirá que podemos iniciar sesión con una cuenta de Microsoft lo cual es opcional, así que lo podemos omitir.

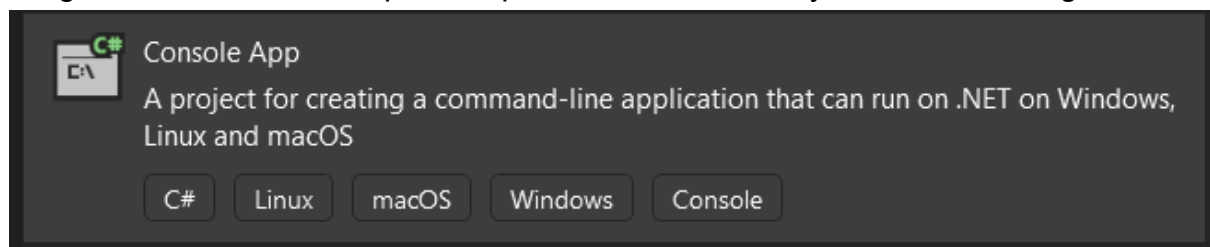
Nos pedirá establecer un color para el desarrollo de nuestra aplicación  
Luego de completar las configuraciones tendremos esta ventana:



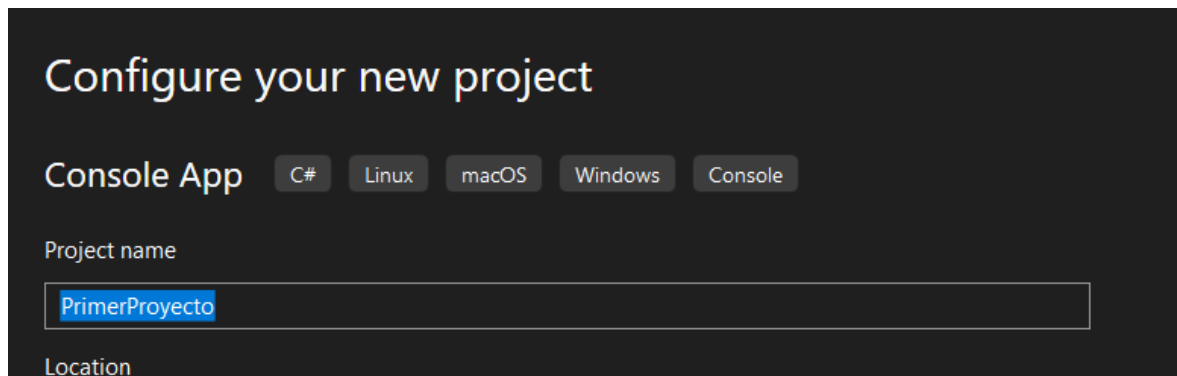
Antes de crear nuestro primer proyecto, vamos a crear una carpeta en nuestros archivos, donde almacenaremos todos nuestros códigos.



Una vez creada nuestra carpeta, daremos clic a “Crear nuevo proyecto”  
Luego, seleccionamos la opción “Aplicación de consola” y damos clic a “Siguiente”



Configuramos el nombre y la carpeta donde estará ubicado nuestro primer proyecto:



Configure your new project

Console App C# Linux macOS Windows Console

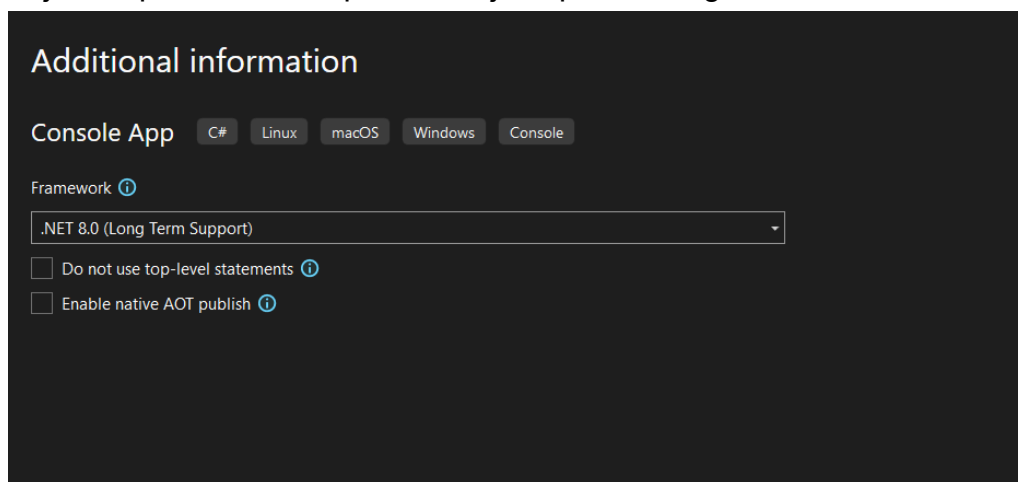
Project name

PrimerProyecto

Location

Damos clic a “siguiente”

Dejamos por default lo que nos deja la pantalla siguiente



Additional information

Console App C# Linux macOS Windows Console

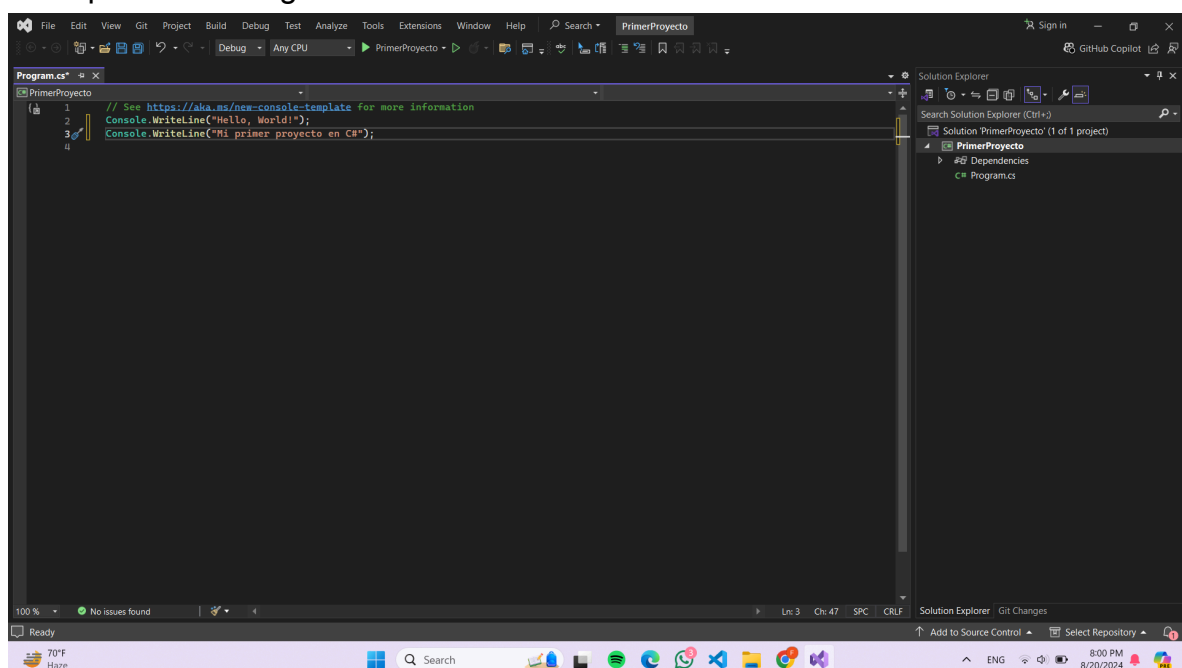
Framework .NET 8.0 (Long Term Support)

☐ Do not use top-level statements

☐ Enable native AOT publish

Y damos clic a “Crear” para finalizar.

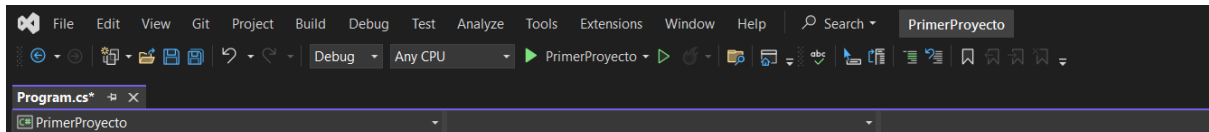
Nos aparecerá la siguiente interfaz:



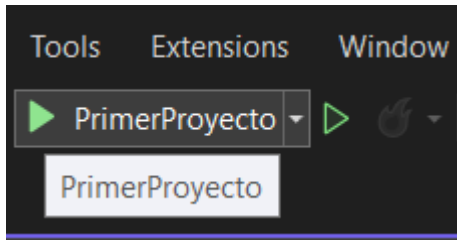
He agregado la siguiente línea de código:

```
Console.WriteLine("Mi primer proyecto en C#");
```

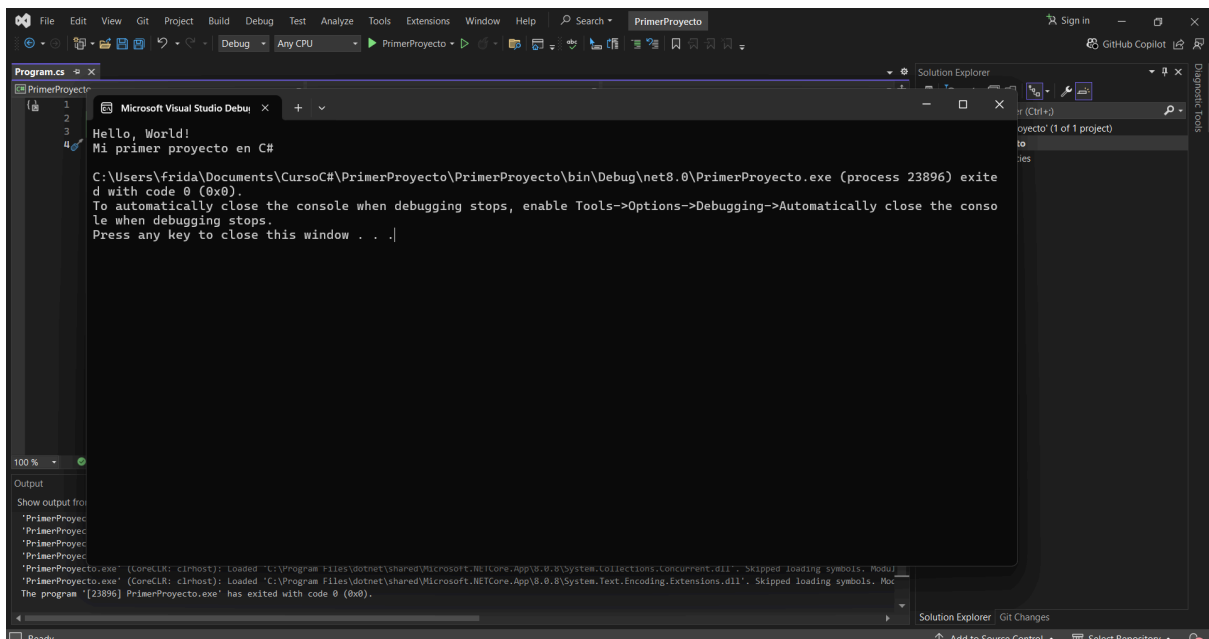
Vamos a ejecutar nuestro proyecto, ubicándonos en la parte superior:



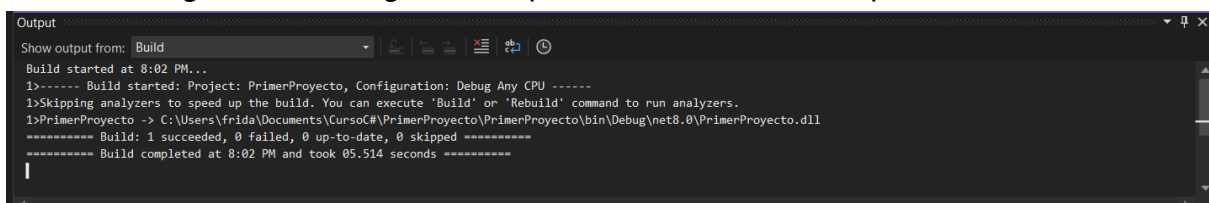
Daremos click al boton verde que está a lado del nombre del programa:



Y automáticamente se ejecuta sin errores y se mostrará la salida de nuestro código en la ventana de la terminal como se muestra a continuación:



Nuestro código no tuvo ningún error, podemos verlo en el output de Visual Studio:



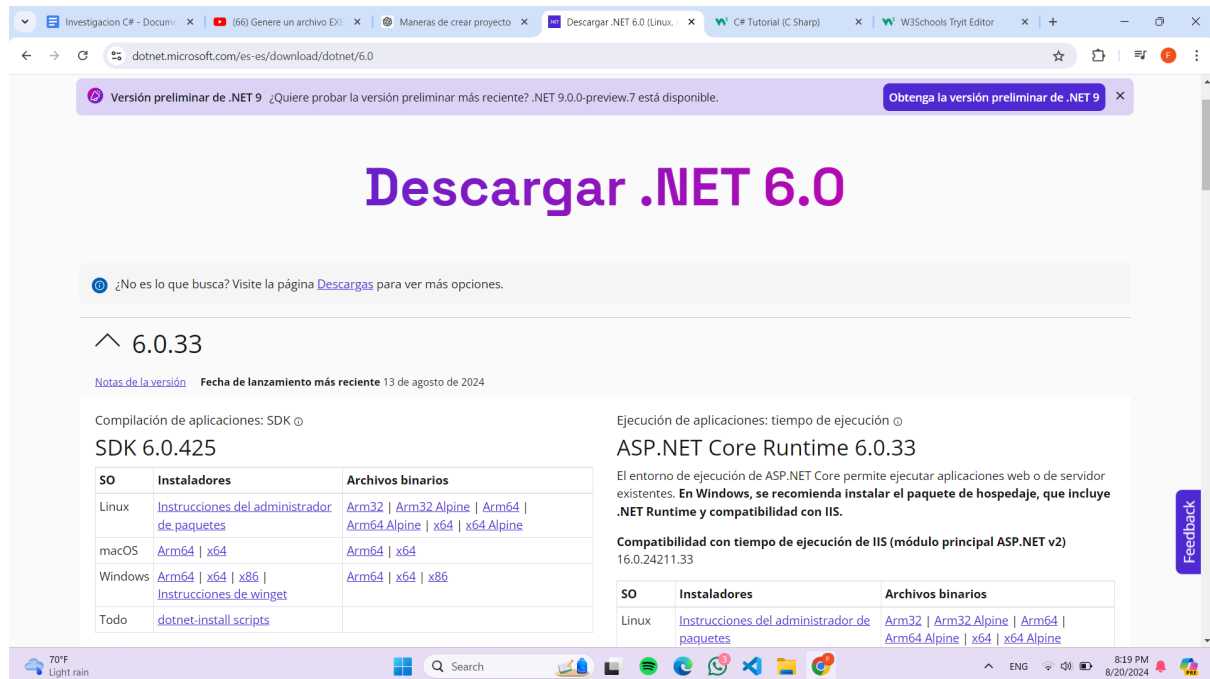
Para salir simplemente hacemos clic en “Archivo” y en la opción “Cerrar” o “Cerrar solución”.

## #2 - Visual Studio Code

Lo primero que haremos será instalar Visual Studio Code.

Una vez que lo hemos instalado, vamos a instalar .NET SDK:

<https://dotnet.microsoft.com/es-es/download/dotnet>



Dotnet.microsoft.com/es-es/download/dotnet/6.0

Obtenga la versión preliminar de .NET 9

# Descargar .NET 6.0

¿No es lo que busca? Visite la página [Descargas](#) para ver más opciones.

6.0.33

[Notas de la versión](#) Fecha de lanzamiento más reciente 13 de agosto de 2024

Compilación de aplicaciones: SDK

### SDK 6.0.425

SO	Instaladores	Archivos binarios
Linux	<a href="#">Instrucciones del administrador de paquetes</a>	<a href="#">Arm32</a>   <a href="#">Arm32 Alpine</a>   <a href="#">Arm64</a>   <a href="#">Arm64 Alpine</a>   <a href="#">x64</a>   <a href="#">x64 Alpine</a>
macOS	<a href="#">Arm64</a>   <a href="#">x64</a>	<a href="#">Arm64</a>   <a href="#">x64</a>
Windows	<a href="#">Arm64</a>   <a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Instrucciones de winget</a>	<a href="#">Arm64</a>   <a href="#">x64</a>   <a href="#">x86</a>
Todo	<a href="#">dotnet-install scripts</a>	

Ejecución de aplicaciones: tiempo de ejecución

### ASP.NET Core Runtime 6.0.33

El entorno de ejecución de ASP.NET Core permite ejecutar aplicaciones web o de servidor existentes. **En Windows, se recomienda instalar el paquete de hospedaje, que incluye .NET Runtime y compatibilidad con IIS.**

Compatibilidad con tiempo de ejecución de IIS (módulo principal ASP.NET v2) 16.0.24211.33

SO	Instaladores	Archivos binarios
Linux	<a href="#">Instrucciones del administrador de paquetes</a>	<a href="#">Arm32</a>   <a href="#">Arm32 Alpine</a>   <a href="#">Arm64</a>   <a href="#">Arm64 Alpine</a>   <a href="#">x64</a>   <a href="#">x64 Alpine</a>

70°F Light rain

Search

ENG 8:19 PM 8/20/2024

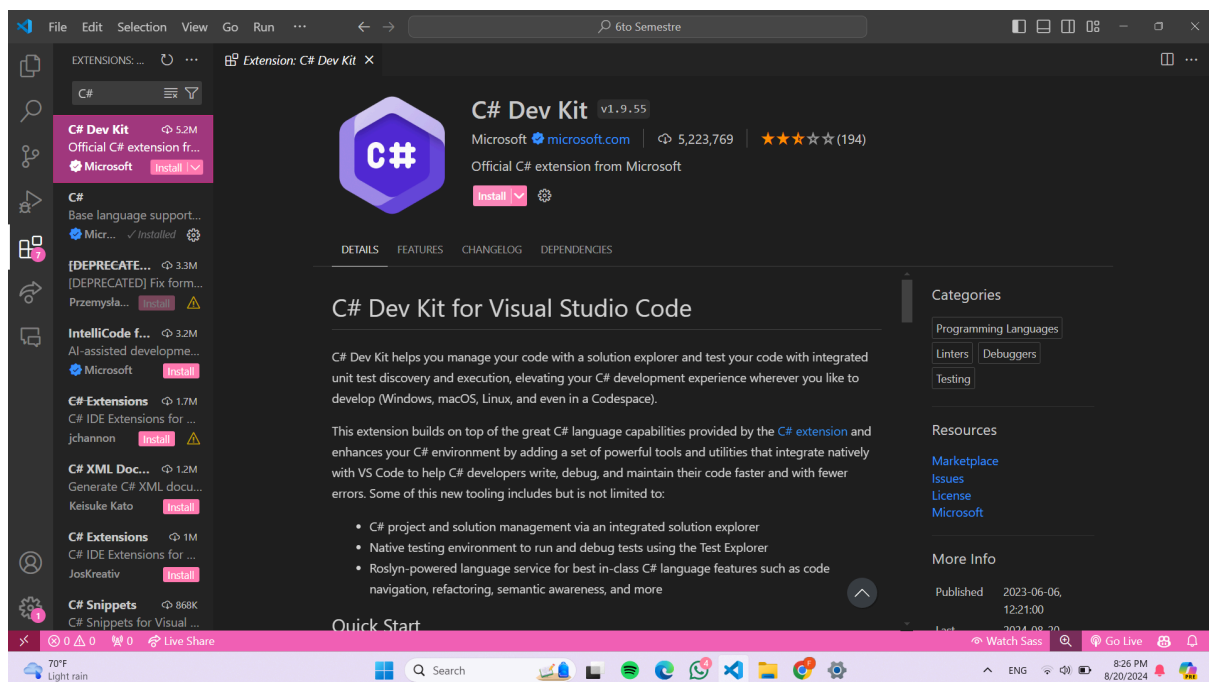
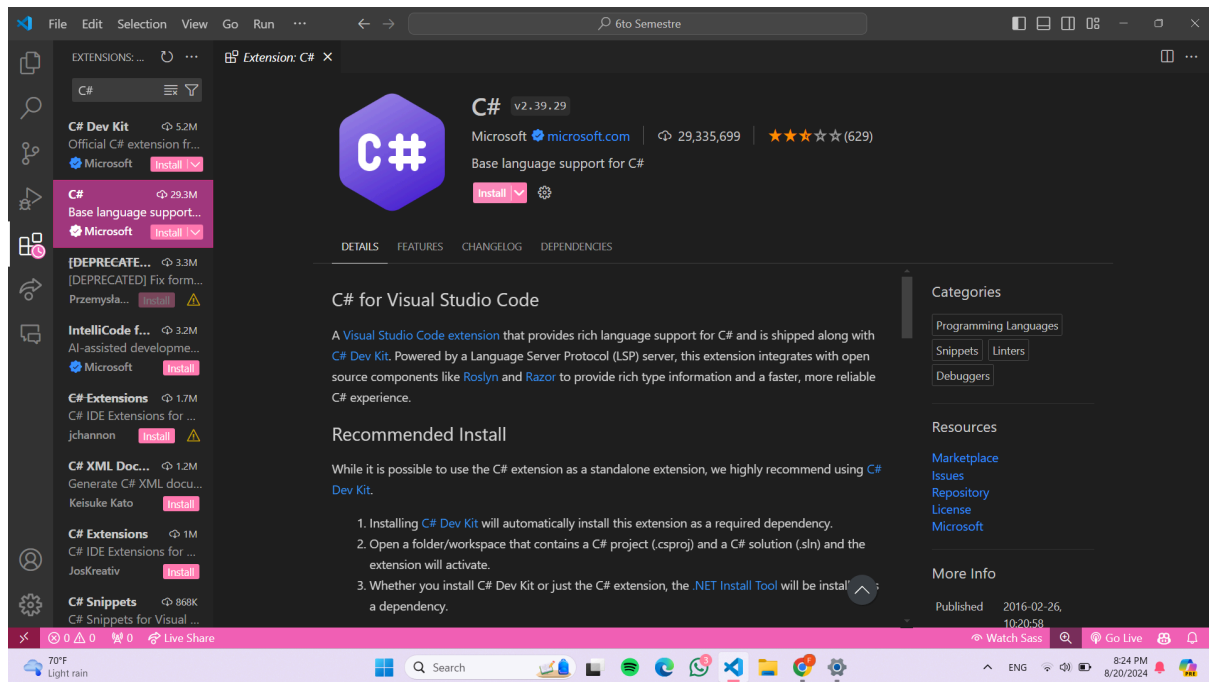
Feedback

Windows	<a href="#">Arm64</a>   <a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Instrucciones de winget</a>	<a href="#">Arm64</a>   <a href="#">x64</a>   <a href="#">x86</a>
---------	---	---

Descargamos x64 dependiendo de las capacidades de nuestra computadora, la opción más común es x64.

Ejecutamos lo que descargamos y ahora nos vamos a Visual Studio Code.

Iremos a las extensiones y vamos a instalar las siguientes:



Ahora, abriremos una nueva terminal y navegaremos hasta la carpeta donde almacenaremos nuestros códigos de C# y ejecutaremos el siguiente código:

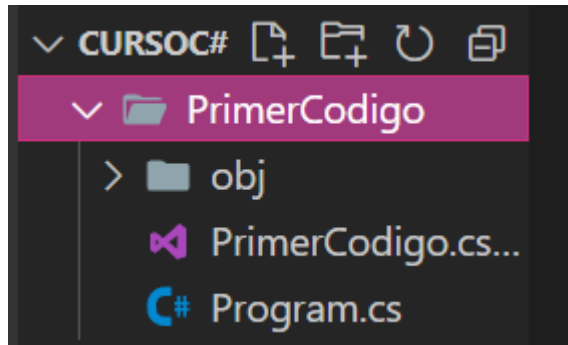
```
dotnet new console -n NombreDelProyecto
```

```
PS C:\Users\frida\Documents\CursoC#> dotnet new console -n PrimerCodigo

Welcome to .NET 8.0!
-----
SDK Version: 8.0.400

Telemetry
-----
```

Ahora podemos abrir la carpeta manualmente y nos mostrará lo siguiente:



Agregaremos una línea más de código:

```
C# Program.cs X
PrimerCodigo > C# Program.cs
1 // See https://aka.ms/new-console-template for more information
2 Console.WriteLine("Hello, World!");
3 Console.WriteLine("Mi primer codigo C# en VSC");
4
```

Y para ejecutarlo, nuevamente abriremos una terminal, asegurándonos de que estemos en la ruta correcta de nuestro proyecto, y vamos a ejecutar el siguiente comando:

*dotnet run*

```
● PS C:\Users\frida\Documents\CursoC#> cd PrimerCodigo
● PS C:\Users\frida\Documents\CursoC#\PrimerCodigo> dotnet run
Hello, World!
Mi primer codigo C# en VSC
○ PS C:\Users\frida\Documents\CursoC#\PrimerCodigo>
```



Ejecutando más líneas para familiarizarnos:

```
//Ahora vamos a ejecutar líneas mas elaboradas
string name = "Juan";
int age = 30;
Console.WriteLine("Hola, soy: " + name + " y tengo: " + age + " años");
```

PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS   COMMENTS

● PS C:\Users\frida\Documents\CursoC#\PrimerCodigo> **dotnet** run  
Hola, soy: Juan y tengo: 30 años

○ PS C:\Users\frida\Documents\CursoC#\PrimerCodigo> █

Fuentes

[https://www.w3schools.com/cs/cs\\_getstarted.php](https://www.w3schools.com/cs/cs_getstarted.php)

<https://www.youtube.com/watch?v=uuk5OBEHB10>