

PRÁCTICA 1: Visual Studio – Instalación y Primeros programas

1. Antes de instalar Visual Studio (VS) en tu equipo, investiga sobre él y responde a las siguientes preguntas:

- ¿Cuántos tipos de VS hay para Windows?
 - Hay 3 tipos: Community, Professional y Enterprise
- ¿Qué diferencia hay entre ellos?
 - Community: De uso gratuito, código abierto y bueno para desarrolladores independientes.
 - Professional: De pago, con beneficios de dicha suscripción y útil para equipos pequeños de desarrolladores
 - Enterprise: También de pago, con beneficios y útil para proyectos de alta calidad y con equipos de proyecto más densos
- ¿Cuál te conviene instalar? ¿Y por qué?
 - El que se recomienda instalar normalmente es el Community debido a que su uso es gratuito y es útil para estudiantes como nosotros
- ¿Qué lenguajes de programación soporta la versión que has elegido?
 - Los principales que soporta son C, C++, .NET, JavaScript/TypeScript, Node.js y Python entre otros
- Una de las características de Visual Studio es que facilita el uso de “control de versiones”. ¿Qué es el “control de versiones”? ¿Qué ventajas o facilidades ofrece? ¿En qué situaciones crees que será útil manejar el “control de versiones”? (Pon algún ejemplo)
 - El control de versiones es una herramienta que nos permite gestionar nuestros archivos y carpetas de como puede ser por ejemplo, un proyecto. A través de esta característica podremos mantener actualizados nuestros archivos y carpetas sin que tenga que depender de nosotros. Uno de los ejemplos mas comunes es Git, una herramienta que nos permite administrar nuestros ficheros y carpetas de la manera más organizada posible

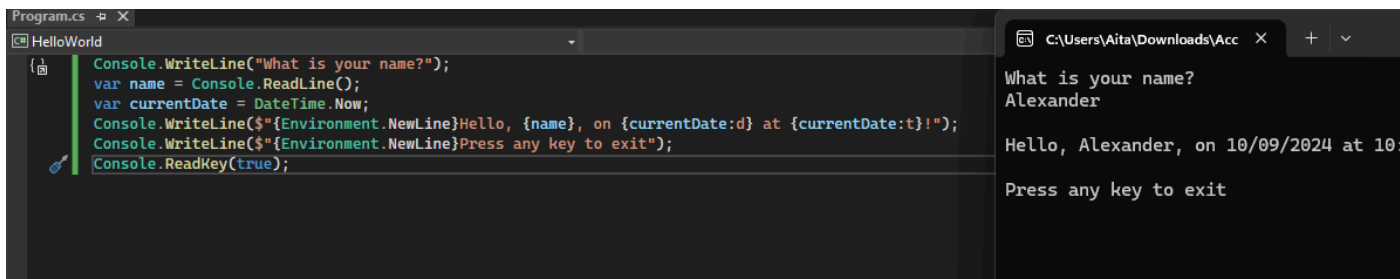
2. Ahora sí, descarga e instala en tu equipo el tipo de Visual Studio que hayas elegido y responde a las siguientes preguntas:

- ¿Cuáles son los requisitos del sistema de Visual Studio 2022 con respecto al sistema operativo y al hardware? (Sistemas operativos compatibles y Hardware)
 - Soporta todo tipo de Sistema Operativo Windows hasta el Windows 10 además de Windows Server, todos ellos de 64bits
 - Procesador ARM64 o x64
 - 4 GB de memoria, como mínimo
 - 2 vCPU y 8 GB de RAM, como mínimo.
 - Espacio en disco duro: mínimo de 850 MB y hasta 210 GB de espacio disponible, en función de las características instaladas; las instalaciones típicas requieren entre 20 y 50 GB de espacio libre
 - Tarjeta de vídeo que admita una resolución de pantalla mínima de WXGA (1366 x 768); Visual Studio funcionará mejor con una resolución de 1920 x 1080 o superior.
- ¿Qué requisitos adicionales se deben tener en cuenta?
 - Se necesitan derechos de administrador para instalar o actualizar Visual Studio.
 - Consulte la Guía del administrador de Visual Studio para ver más consideraciones e instrucciones sobre cómo instalar, implementar, actualizar y configurar Visual Studio en una organización.
 - Se requiere .NET Framework 4.7.2 o una versión superior para instalar Visual Studio. (.NET Framework 4.5.2 o posterior es necesario para instalar Visual Studio 17.8 y versiones anteriores.) Visual Studio requiere .NET Framework 4.8 para ejecutarse. Si .NET Framework 4.8 aún no está instalado, se instalará durante la configuración.
 - El desarrollo de aplicaciones Windows universales, incluidos el diseño, la edición y la depuración, requiere Windows 10. Windows Server 2019 y Windows Server 2016 pueden utilizarse para compilar

aplicaciones Windows universales desde la línea de comandos.

- Se requiere el entorno de ejecución WebView2 para instalar Visual Studio. Si aún no está instalado, se instalará durante la configuración. Se producirá un error en la instalación de Visual Studio si WebView2 no se puede instalar correctamente (ya sea debido a un error de instalación o a restricciones de directiva de grupo de instalación o actualización).
- Team Foundation Server 2019 Office Integration requiere Office 2016, Office 2013 u Office 2010.
- No se recomienda habilitar Smart App Control, una característica de Windows, en las máquinas de desarrollo. Cualquier configuración distinta de "desactivado" podría afectar negativamente al rendimiento de Visual Studio.
- ¿Cumple tu equipo los requisitos para instalar la versión de Visual Studio 2022 que has elegido?
 - Si cumple los requisitos de instalación

3. ➤ Crear una aplicación de consola de .NET con Visual Studio:



```
Program.cs
HelloWorld
{
    Console.WriteLine("What is your name?");
    var name = Console.ReadLine();
    var currentDate = DateTime.Now;
    Console.WriteLine($"{Environment.NewLine}Hello, {name}, on {currentDate:d} at {currentDate:t}!");
    Console.WriteLine($"{Environment.NewLine}Press any key to exit");
    Console.ReadKey(true);
}
```

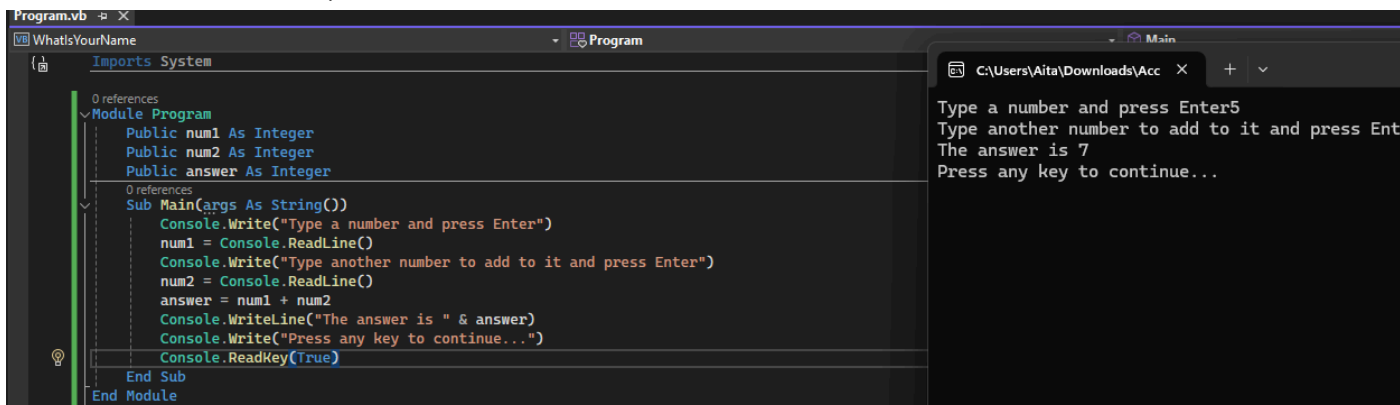
C:\Users\Aita\Downloads\Acc X + v

What is your name?
Alexander

Hello, Alexander, on 10/09/2024 at 10:00

Press any key to exit

4. ➤ Crear una aplicación de consola de Visual Basic sencilla:



```
Program.vb
What's Your Name
Imports System

Module Program
    Public num1 As Integer
    Public num2 As Integer
    Public answer As Integer

    Sub Main(args As String())
        Console.WriteLine("Type a number and press Enter")
        num1 = Console.ReadLine()
        Console.WriteLine("Type another number to add to it and press Enter")
        num2 = Console.ReadLine()
        answer = num1 + num2
        Console.WriteLine("The answer is " & answer)
        Console.WriteLine("Press any key to continue...")
        Console.ReadKey(True)
    End Sub
End Module
```

C:\Users\Aita\Downloads\Acc X + v

Type a number and press Enter5

Type another number to add to it and press Enter7

The answer is 7

Press any key to continue...

5. ➤ Crear un proyecto de aplicación de consola de C++:

```
// ProyectoPrueba001.cpp : This file contains the 'main' function.
//
#include <iostream>

int main()
{
    std::cout << "Hello World!\n" <<std::endl;
    return 0;
}
```

Hello World!

D:\DAM3\AccesoADatos\ProyectoPrueba001\x64\Debug\ProyectoPrueba001.exe
To automatically close the console when debugging stops, enable the option 'Automatically close the console when debugging stops' under the Debugging tab in the Project Properties dialog.
Press any key to close this window . . .|

6. ➤ Crear una calculadora de consola en C++:

```
#include <iostream>
#include "Calculator.h"
using namespace std;

int main()
{
    double x = 0.0;
    double y = 0.0;
    char oper = '+';
    double result = 0.0;

    cout << "Calculator Console Application" << endl << endl;
    cout << "Introduce la operacion a realizar. Formato: a+b | a-b | a*b | a/b" << endl;

    Calculator c;
    while (true)
    {
        cin >> x >> oper >> y;
        result = c.Calculate(x, oper, y);
        cout << "Result" << " of " << x << oper << y << " is: " << result << endl;
    }

    return 0;
}
```

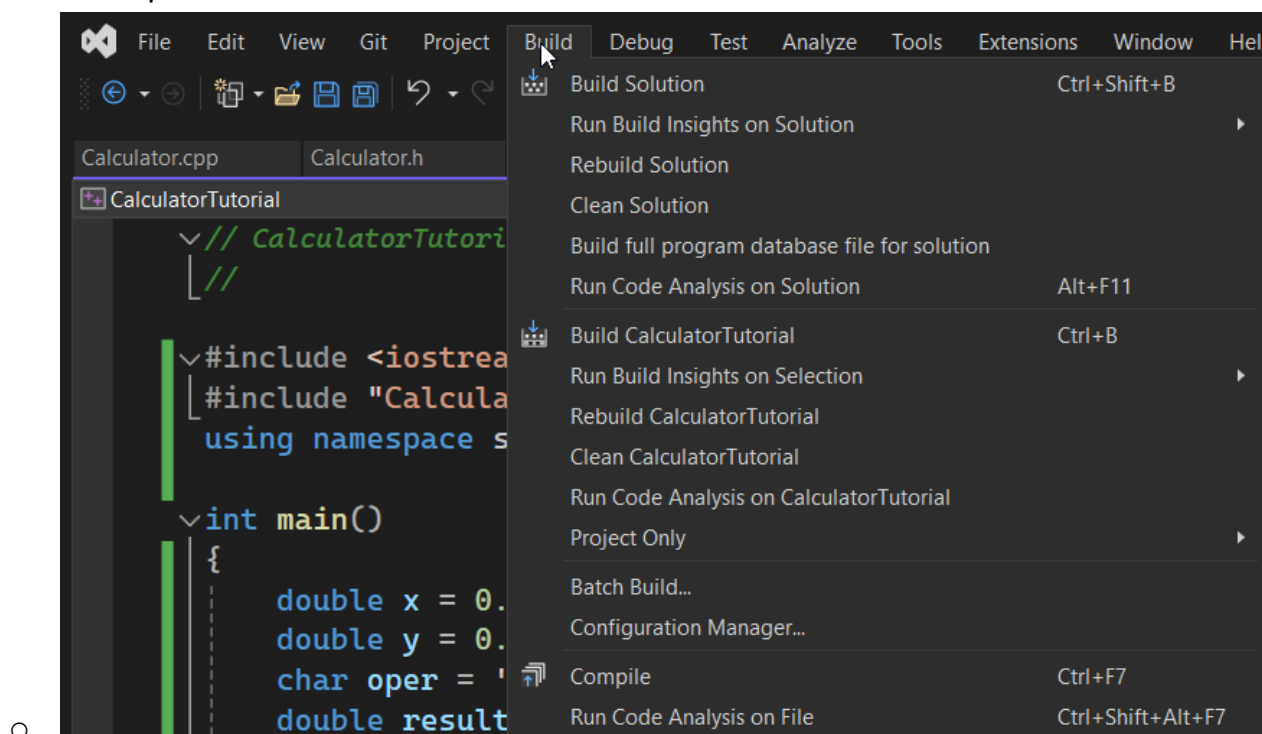
Calculator Console Application

Introduce la operacion a realizar. Formato: a+b | a-b | a*b | a/b

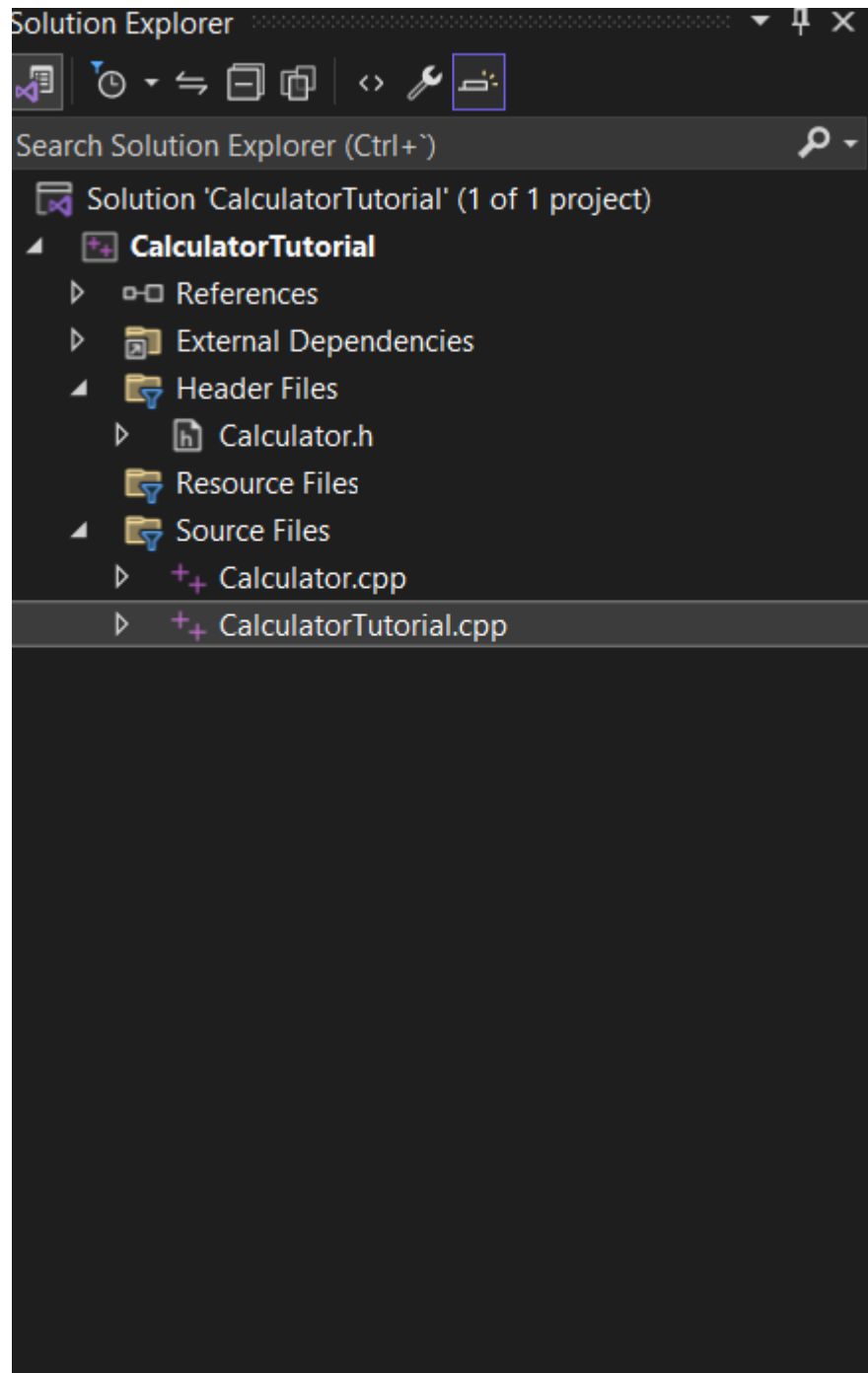
5
/
5
Result of 5/5 is: 1

Extra

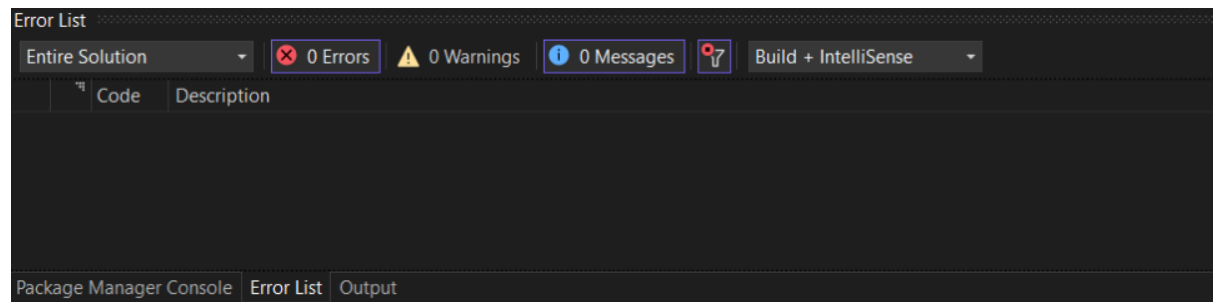
- ¿Cómo puedes compilar tu código? Es decir, ¿a qué parte de la ventana de Visual Studio debes dirigirte para compilar?



- ¿En qué parte de la ventana de VS puedes ver los proyectos, soluciones y ficheros de los mismos?



-
- ¿En qué parte de la ventana de VS se muestran los errores y bugs que pueda haber en el código?



D) Recogiendo algunos de los conceptos aprendidos...

1. ¿Qué es Visual Studio y para qué se utiliza?
 - a. Visual Studio es un entorno de desarrollo desarrollado por Microsoft. Se utiliza principalmente para desarrollar aplicaciones de software.
2. ¿Qué diferencia hay entre un proyecto y una solución?
 - a. Un proyecto es una unidad básica de desarrollo donde se almacenan todos nuestros ficheros, recursos, librerías, clases, etc..., mientras que una solución son múltiples proyectos
3. ¿Qué es el archivo del proyecto?
 - a. El archivo de proyecto en Visual Studio es un archivo que contiene la información y configuración específica sobre un proyecto individual dentro de una solución
4. ¿Qué es .NET? ¿Y para qué se utiliza? ¿Qué lenguajes de programación se pueden utilizar con .NET?
 - a. .NET es un entorno común para crear aplicaciones que pueden ejecutarse en diferentes sistemas operativos, como Windows, Linux, o mac. Y sus principales lenguajes de programación son toda la familia de C y Visual Basic
5. ¿Qué es Visual Basic .NET? ¿Y para qué se utiliza?
 - a. Es un lenguaje de programación que se utiliza para programar distintas aplicaciones para distintos dispositivos
6. Si quieres, puedes añadir alguna característica que te parezca interesante de Visual Studio y que no se haya mencionado en esta práctica. (Opcional)

- a. Creo que el hecho de mencionar que tienes una posibilidad de gestión de control de versiones es algo a destacar ya que se usa frecuentemente
- 7. Indica si te has encontrado alguna dificultad durante la realización de la práctica y cuáles han sido.
 - a. No me he encontrado ninguna dificultad a la hora de realizar la práctica