

Entornos de desarrollo

Práctica 3: Instalación y configuración de un IDE



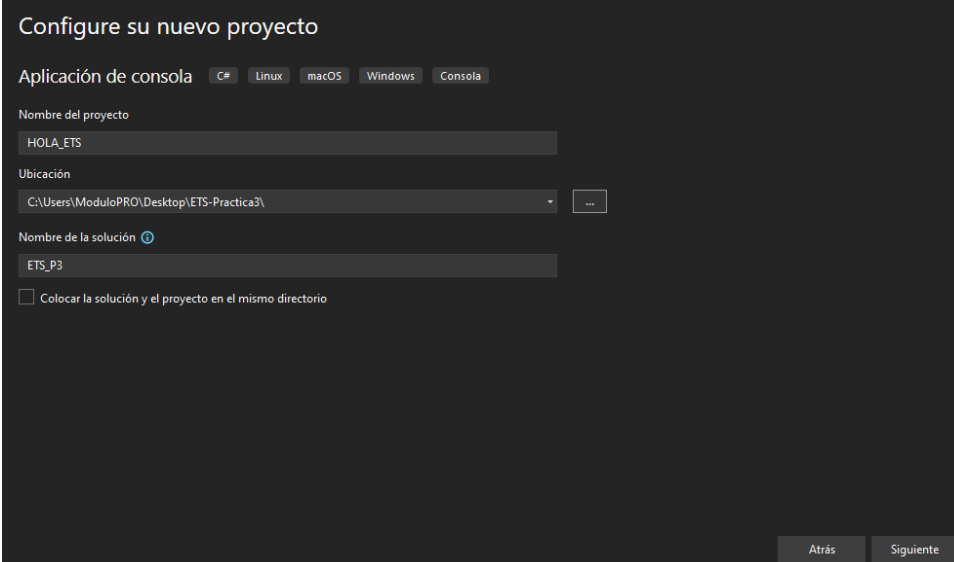
1. Cuando abres el VS te pide que abras una solución o un proyecto. ¿Cuál es la diferencia entre ellos?

Una solución es un contenedor que Visual Studio usa para organizar uno o más proyectos relacionados. Cuando se abre una solución, Visual Studio carga automáticamente todos los proyectos que contiene la solución.

Fuente:

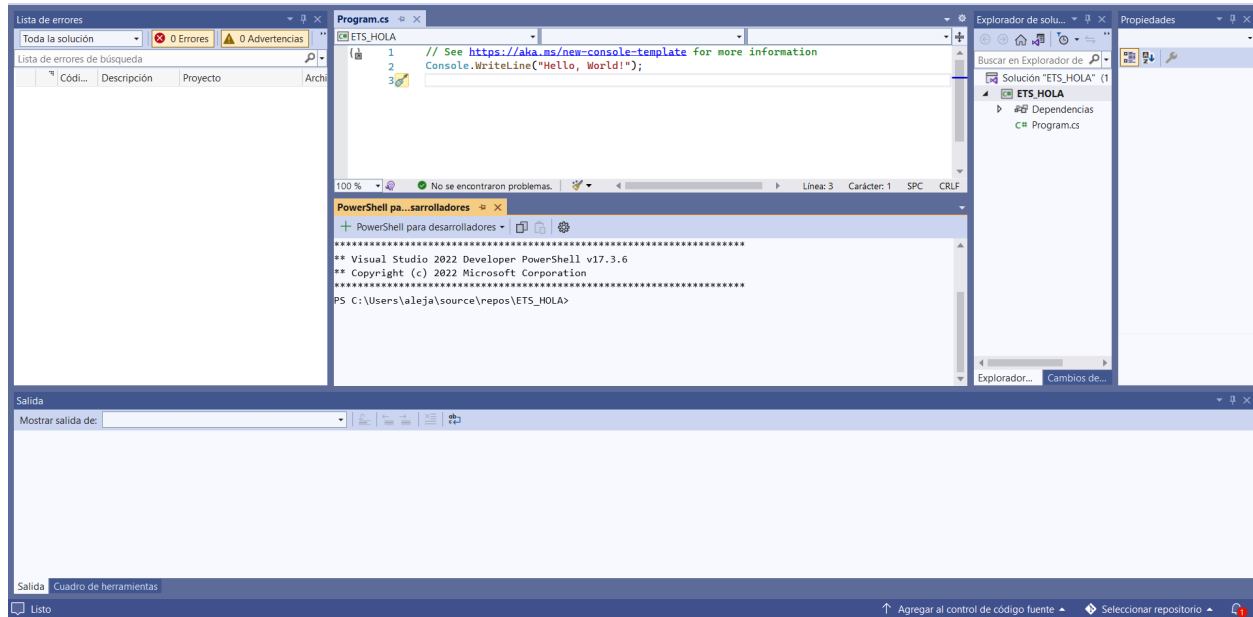
<https://learn.microsoft.com/es-es/visualstudio/get-started/tutorial-projects-solutions?view=vs-2022>

Crea una solución llamada **ETS_P3** y dentro de ella un proyecto llamado **HOLA_ETTS**. Guarda el proyecto en una carpeta que no sea la que viene por defecto

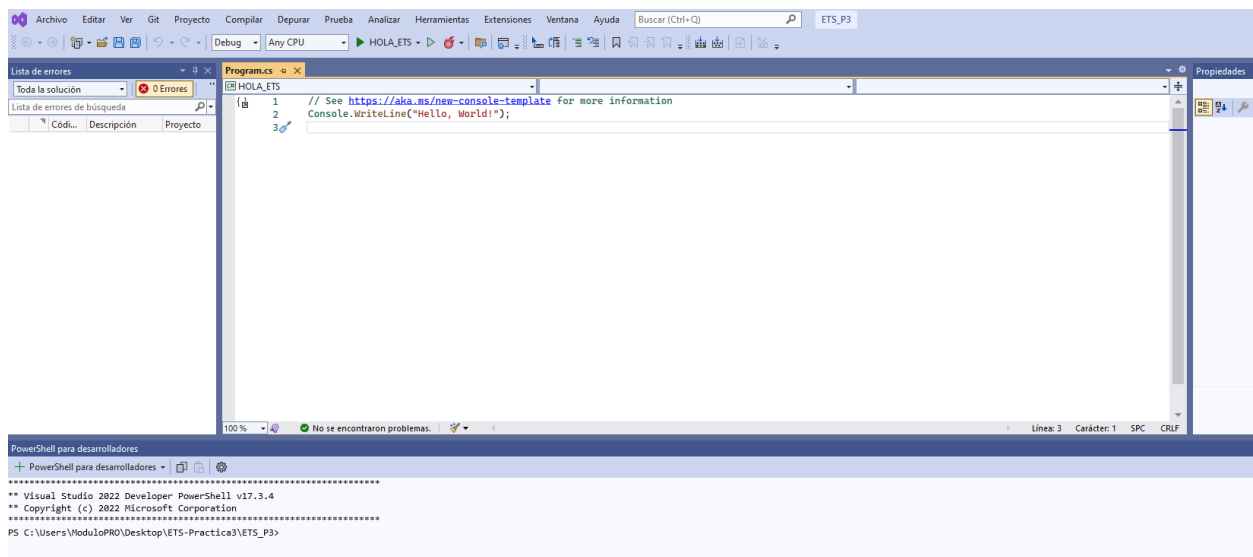


2. Dos de las opciones del menú superior que nos permiten modificar la disposición de los elementos de la interfaz son el apartado de **Ventana** y de **Ver**. Todos los elementos que ves en la pantalla (editor de texto, cuadro de herramientas, explorador de soluciones, salida...) son ventanas que se pueden eliminar y añadir, podéis ver una lista de las que hay en el apartado de **Ver**. Realiza estas tareas:

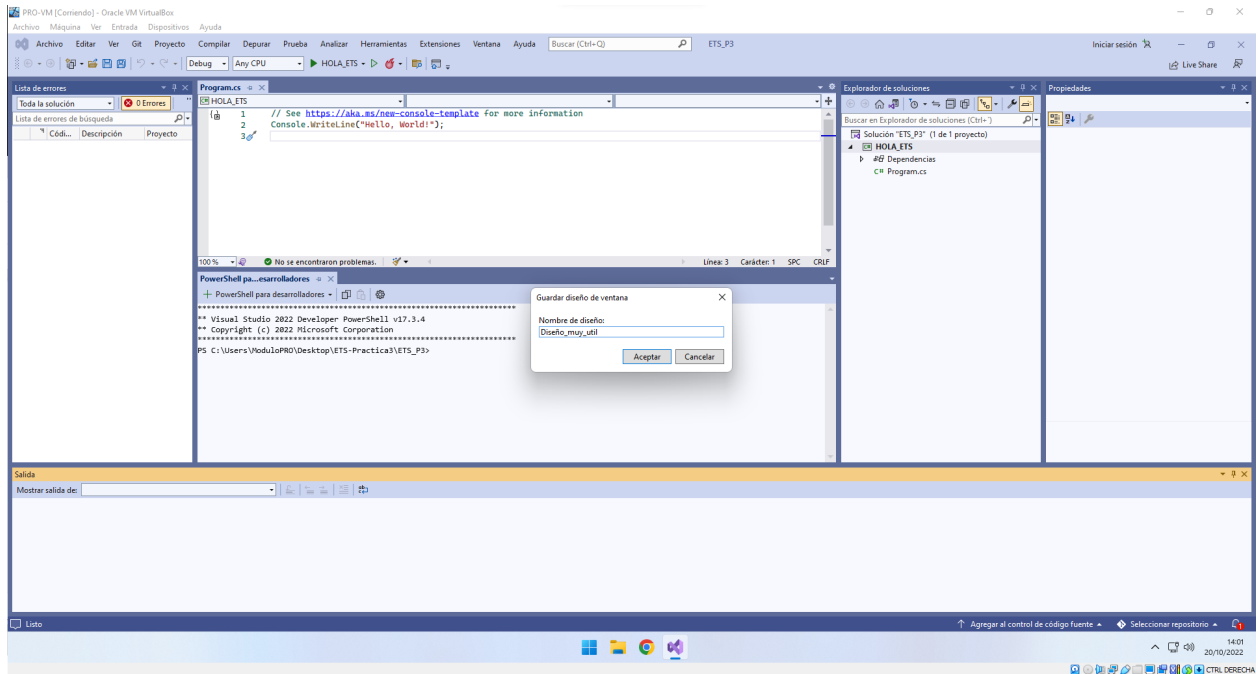
- Añade la ventana de **Lista de errores** y de **terminal**.
- Elimina la ventana de Explorador **de soluciones**.



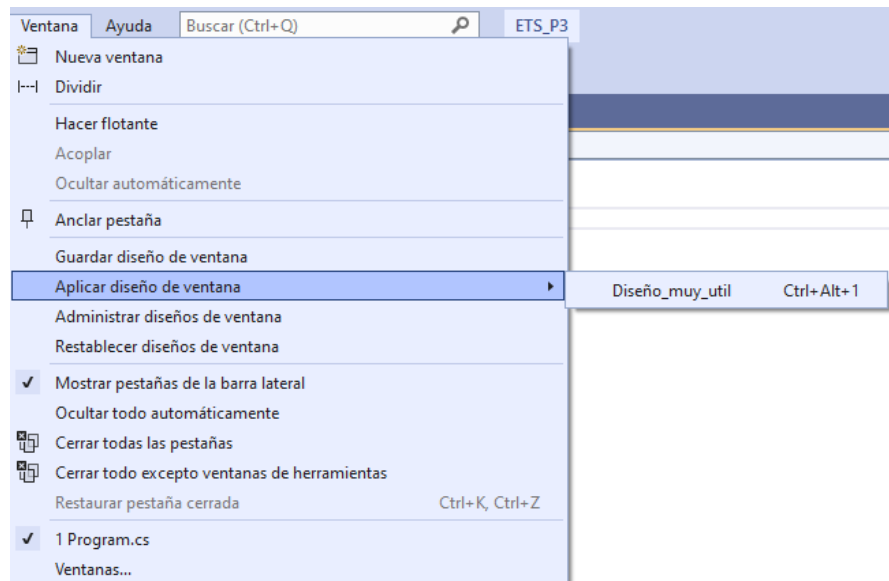
- Las ventanas las puedes reordenar como quieras, seleccionando una de ellas y arrastrando hacia donde quieres moverla. Puedes moverla debajo/encima de otra, al lado, dentro del mismo espacio... Genera una interfaz igual a esta



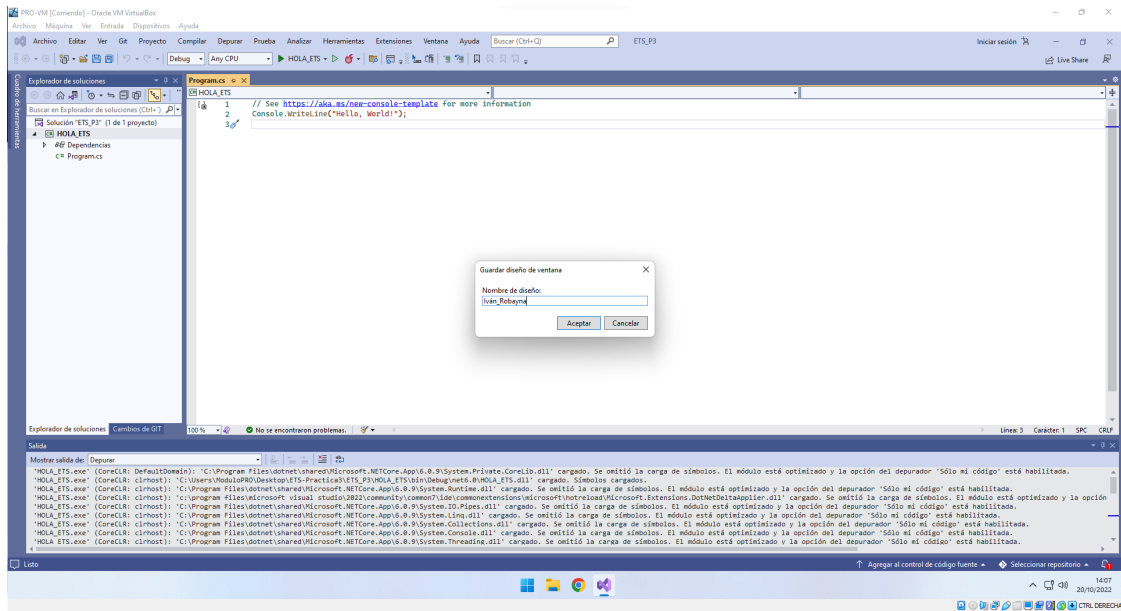
- La disposición de ventanas se puede guardar y recargar en cada momento. Para ello ve a **Ventana** y guarda este diseño tan útil que acabas de recrear, con el nombre **diseño_muy_util**.



- Restablece el diseño por defecto, y carga el diseño_muy_util.



- Crea una disposición de ventanas con la que te encuentres cómodo, y guardarlo con tu nombre.



3. Otra opción dentro de ventana es la posibilidad de modificar la barra de herramientas, que son esos iconos debajo de la barra superior, representando atajos a diferentes acciones (abrir fichero, guardar...)

- Añade la opción de compilar en la barra de herramientas. ¿Cómo aparece?

Al añadir a la barra de herramientas la opción de compilar, se añaden los siguientes iconos a la barra:



4. En el apartado de *Herramientas* puedes modificar diferentes atributos de nuestro IDE. Investiga cómo puedo:

- Modificar el tema del IDE a modo oscuro.
- Cambiar la fuente del texto normal sin formato a comic sans color amarillo muy chillón
- Ahora cambia la fuente de los comentarios a Elephant color rosa. ¿La fuente es común o sólo puedo cambiar el color?
- Modifica el interlineado a 2.0, y permite que se pueda ver los espacios en blanco.

```
... Console.WriteLine("El número es positivo.");  
}  
  
else  
{  
  
... Console.WriteLine("El número es negativo.");  
  
... //comentario para probar el color ...  
}
```

Modificación del tema a oscuro, con la fuente cambiada y el color cambiado, y con la fuente de los comentarios cambiada a color rosa. Solo se puede establecer una fuente común así que no se puede tener diferentes fuentes para el texto normal y para los comentarios. También está modificado el interlineado a 2.0 y vagamente pero se aprecian al final los espacios en blanco

- Activar y desactivar la opción para que se enumeren las líneas de código.

```
1 namespace Proyecto01  
2 {  
3     0 referencias  
4     ... class Program  
5     ... {  
6         0 referencias  
7         ... static void Main(String[] args)  
8         ... {
```

A la izquierda se puede apreciar las líneas de código numeradas.

- En **Editor de texto > C# > estilo de código** puedes informar al IDE cómo autocompletar o guiar al programador al escribir el código, avisándote, o incluso mostrarse como error alguna decisión tomada: no usar llaves en un condicional, no usar paréntesis, el calificador this...
 - ¿Qué diferencia ves si modificas algún calificador relacionado con el uso de paréntesis en las condicionales?

Puedes definir que cada grupo de condición vaya entre paréntesis para poder diferenciar ambos grupos con más claridad.

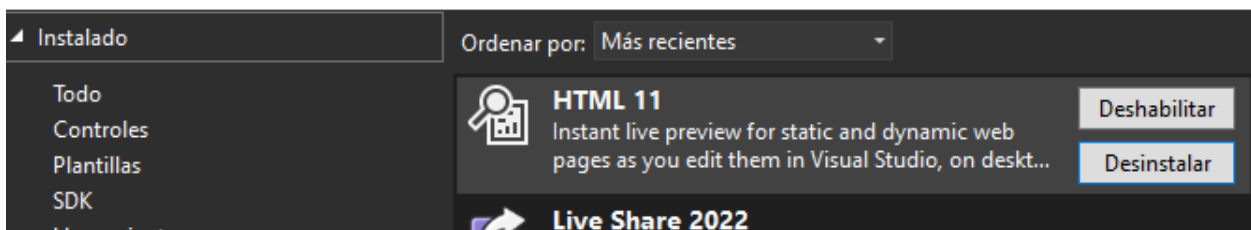
5. Las clases se están mezclando y ahora desde la asignatura de Lenguajes de Marcas te piden que uses el VS para programar páginas web en HTML. Te han soplado que puedes descargar **Plugins** que te faciliten la programación en HTML. Para ello:

- Define qué es un **Plugin**.

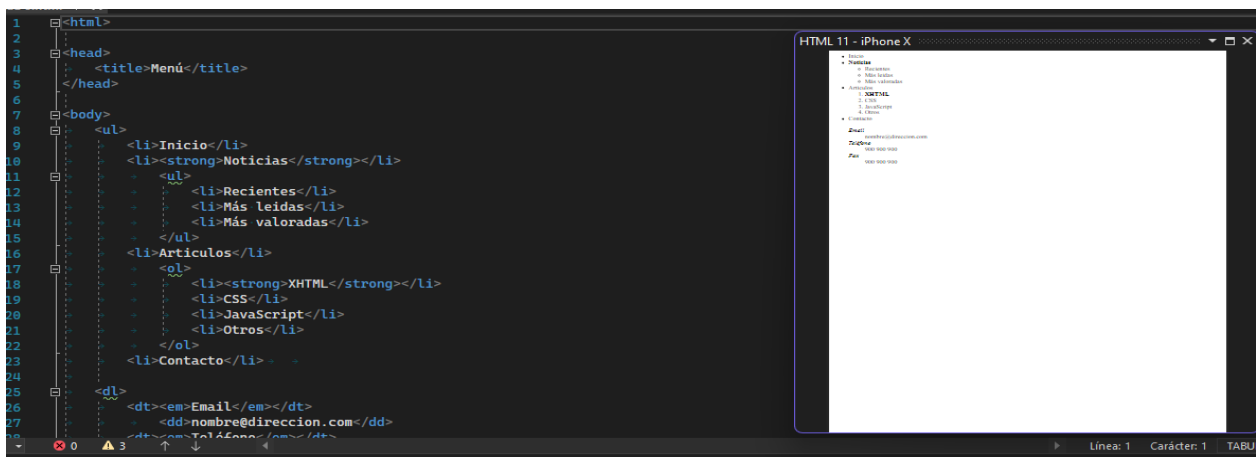
Los plugins son extensiones que permiten personalizar y mejorar la experiencia que tenemos, ya sea añadiendo nuevas características, mejorando las ya existentes o ambas. Pueden ser tanto extensiones muy simples como extensiones realmente complejas que añadan muchísimas funcionalidades.

- Abre la pestaña de extensiones del VS, e instala el plugin **HTML-11**.

Administrar extensiones



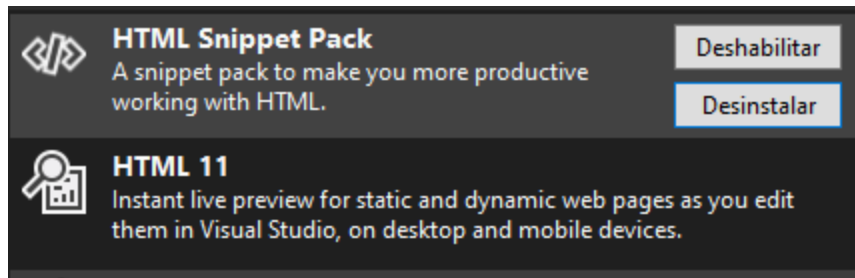
- Una vez instalado, abre un fichero .html (puedes abrir una carpeta que tengas con varios ficheros .html, dándole a File > Open) y explica qué hace dicho plugin.



Su principal función es la previsualización de tu página web HTML en diferentes navegadores o dispositivos

- ¿Qué es un **Snippet** de código? Instala un plugin que permita gestionar snippets de HTML.

Un snippet de código quiere decir un “fragmento de código” que puedes reutilizar para trabajar más rápido y eficaz, como por ejemplo, un snippet de HTML que te permita cerrar automáticamente las etiquetas.

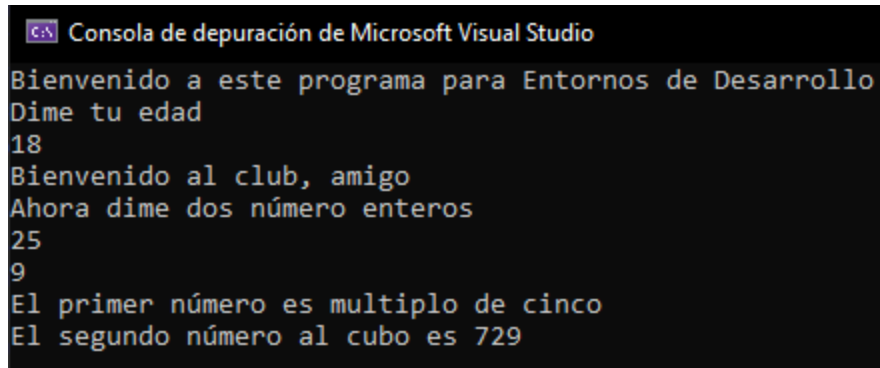


- Desinstala ambos plugins.

6. Crea un proyecto en el que muestre por pantalla un mensaje de bienvenida, y le pida la edad. Si es mayor de edad, muestra el mensaje *bienvenido al club, amigo*, y en caso contrario muestra el mensaje *lo siento chaval*. Sólo en el caso en el que el usuario sea mayor de edad, pídele dos números enteros. Del primero indica si es múltiplo de 5, y del segundo calcula su cubo.






```
1  using System;
2
3  namespace Ejercicio02
4  {
5      0 referencias
6      class Ejercicio02
7      {
8          0 referencias
9          static void Main(String[] args)
10         {
11             int Edad;
12             int num1, num2;
13             Console.WriteLine("Bienvenido a este programa para Entornos de Desarrollo");
14             Console.WriteLine("Dime tu edad");
15             Edad = Convert.ToInt32(Console.ReadLine());
16             if (Edad >= 18)
17             {
18                 Console.WriteLine("Bienvenido al club, amigo");
19                 Console.WriteLine("Ahora dime dos número enteros");
20                 num1 = Convert.ToInt32(Console.ReadLine());
21                 num2 = Convert.ToInt32(Console.ReadLine());
22                 num1 = num1 % 5;
23                 if (num1 == 0)
24                 {
25                     Console.WriteLine("El primer número es multiplo de cinco");
26                 }
27                 else
28                 {
29                     Console.WriteLine("El primer número no es multiplo de cinco");
30                 }
31                 num2 = (int)Math.Pow(num2, 3);
32                 Console.WriteLine("El segundo número al cubo es " + num2);
33             }
34             else
35             {
36                 Console.WriteLine("Lo siento chaval");
37             }
38         }
39     }
```

- Una vez tengas este programa, lánzalo desde el IDE, pulsando la tecla F5 o dándole a la opción de Run en la barra de herramientas.

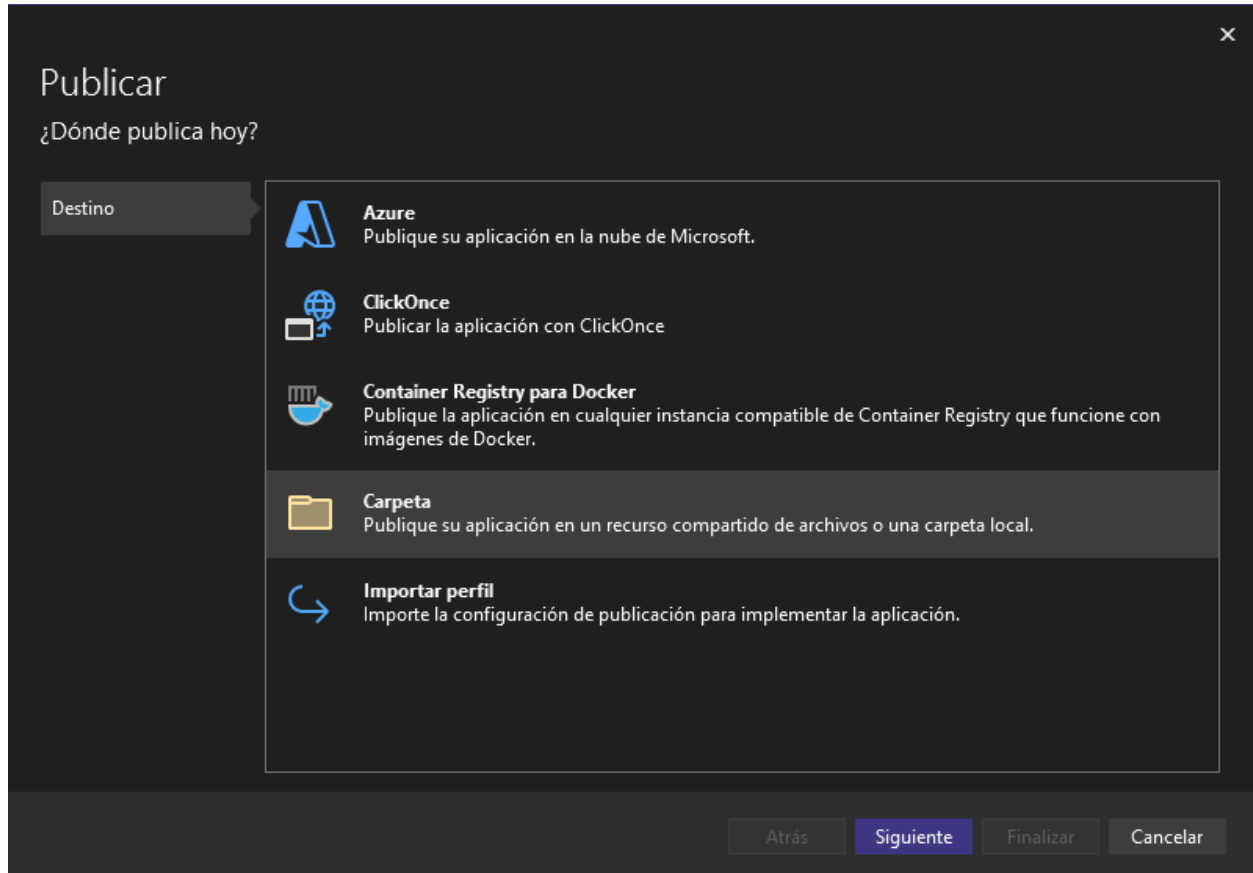


```
C:\> Consola de depuración de Microsoft Visual Studio
Bienvenido a este programa para Entornos de Desarrollo
Dime tu edad
18
Bienvenido al club, amigo
Ahora dime dos número enteros
25
9
El primer número es multiplo de cinco
El segundo número al cubo es 729
```

- Compila el proyecto y busca el fichero .exe. Lo encontrarás en la carpeta del proyecto, dentro de la carpeta **bin**.

Nombre	Fecha de modificación	Tipo	Tamaño
 Tareas de Clase.deps.json	26/10/2022 18:11	Adobe After Effect...	1 KB
 Tareas de Clase.dll	01/11/2022 15:51	Extensión de la ap...	6 KB
 Tareas de Clase.exe	01/11/2022 15:51	Aplicación	145 KB
 Tareas de Clase.pdb	01/11/2022 15:51	Program Debug D...	11 KB
 Tareas de Clase.runtimeconfig.json	26/10/2022 18:11	Adobe After Effect...	1 KB

- Crea el ejecutable en otra carpeta mediante la opción de **Publicar**. ¿Qué significa esta opción? ¿En qué diferentes plataformas puedes publicar? Hazlo en una carpeta en tu escritorio.

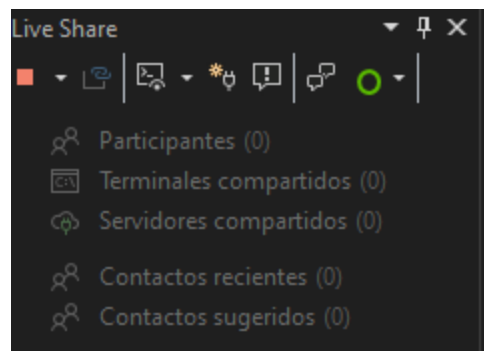


Seleccionamos la opción de carpeta y simplemente ponemos nuestro escritorio en la ruta y lo creamos.

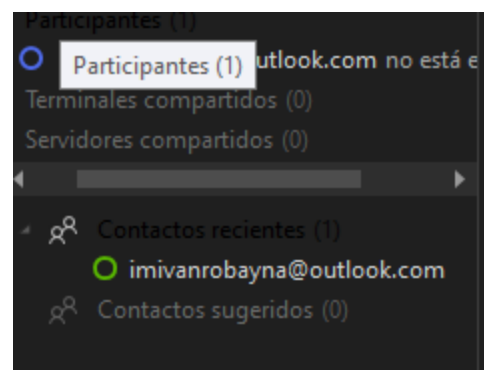
Ejercicio 2: (1.5 p)

La programación en parejas es un estilo de programación en el que dos personas se sientan juntas a programar. En general uno programa, y el otro mira. Aunque parece raro, es una práctica beneficiosa ya que añade un par de ojos para revisar el código, facilita la comunicación en voz alta, lo que ayuda a identificar errores... Microsoft sacó hace ya unos años la herramienta llamada **Live Share**, la cual permite compartir una hoja de trabajo y que varias personas la modifiquen en tiempo real (similar a un fichero de Google Docs). Realizad un informe sobre la instalación (si fuese necesaria) y de preparación de la sesión. Tras ello, haced un vídeo corto (10 – 20 segundos) en el que trabajáis de forma remota para hacer un programa sencillo de prueba, que reciba un entero de teclado y diga si es par o impar.

- La instalación del Live Share no es necesaria ya que el plugin viene por defecto instalado en las versiones actuales. Lo único que hay que hacer es loguearte con una cuenta de Microsoft o de GitHub y mandarle a la persona con la que vas a compartir el trabajo en enlace de invitación.



Antes y después de unirse



El video viene en el zip aparte.

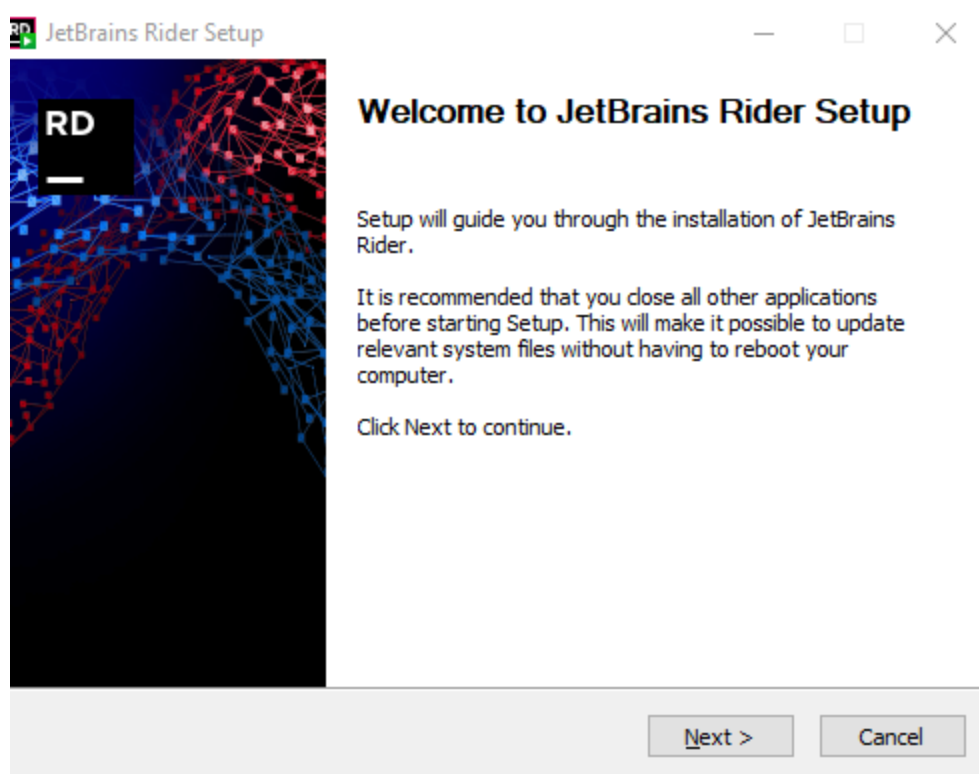
Ejercicio 3 (1.5 p)

El VSC no es el único IDE que hay orientado a C# o C++. En este apartado tenéis que instalar otros IDE:

- [Rider](#): El IDE de C# de la empresa JetBrains, responsables de otros IDE como IntelliJ o Android Studio.
- [Visual Studio Code](#): Otro IDE de Microsoft, más versátil que el Community y multilenguaje.

Documenta el proceso de instalación, y ejecutad el mismo programa que hayáis hecho en el último apartado del tema anterior. Una vez hecho esto, sintetiza cuales son las similitudes/diferencias entre los 3 IDEs, ¿hay alguno que os guste más que el otro?

Para descargar Rider procedemos a ir a la página oficial y darle al apartado de descargar. Elegimos el sistema operativo y descargamos el ejecutable para instalarlo.



Seguimos los pasos típicos como elegir la carpeta de instalación o si queremos que el programa abra ciertas extensiones automáticamente.

Visual Studio Code es bastante parecido a Visual Studio (obviamente), aunque a mi gusto tiene la interfaz un poco más limpia así de primeras y es más amigable con el usuario, aunque al final con el nivel de personalización que tienen estos IDEs pues más allá de las diferencias en las funcionalidades que tienen cada uno creo que el tema estético y de interfaz queda en segundo plano. Rider tiene su propia consola integrada en el programa y eso me parece una mejora a la hora de ejecutar programas, pero su sistema para detectar errores me parece bastante pobre ya que estuve varios minutos dando vueltas por un fallo que me daba y se debía a que me faltaba una mayúscula, error que en Visual Studio me hubiese indicado sin problema.

BIBLIOGRAFÍA

[How to Download and Install Visual Studio - YouTube](#)

[Rider: El IDE .NET multiplataforma de JetBrains](#)

[C# If ... Else \(w3schools.com\)](#)