

# C# Fundamentals

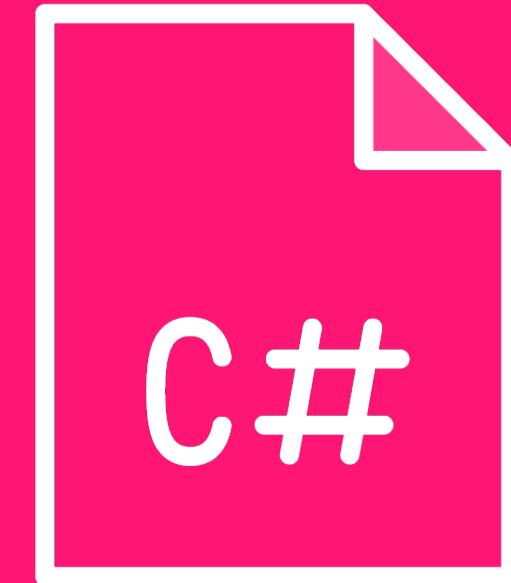
## Getting Started with C# and .NET



**Gill Cleeren**

CTO Xebia Microsoft Services Belgium

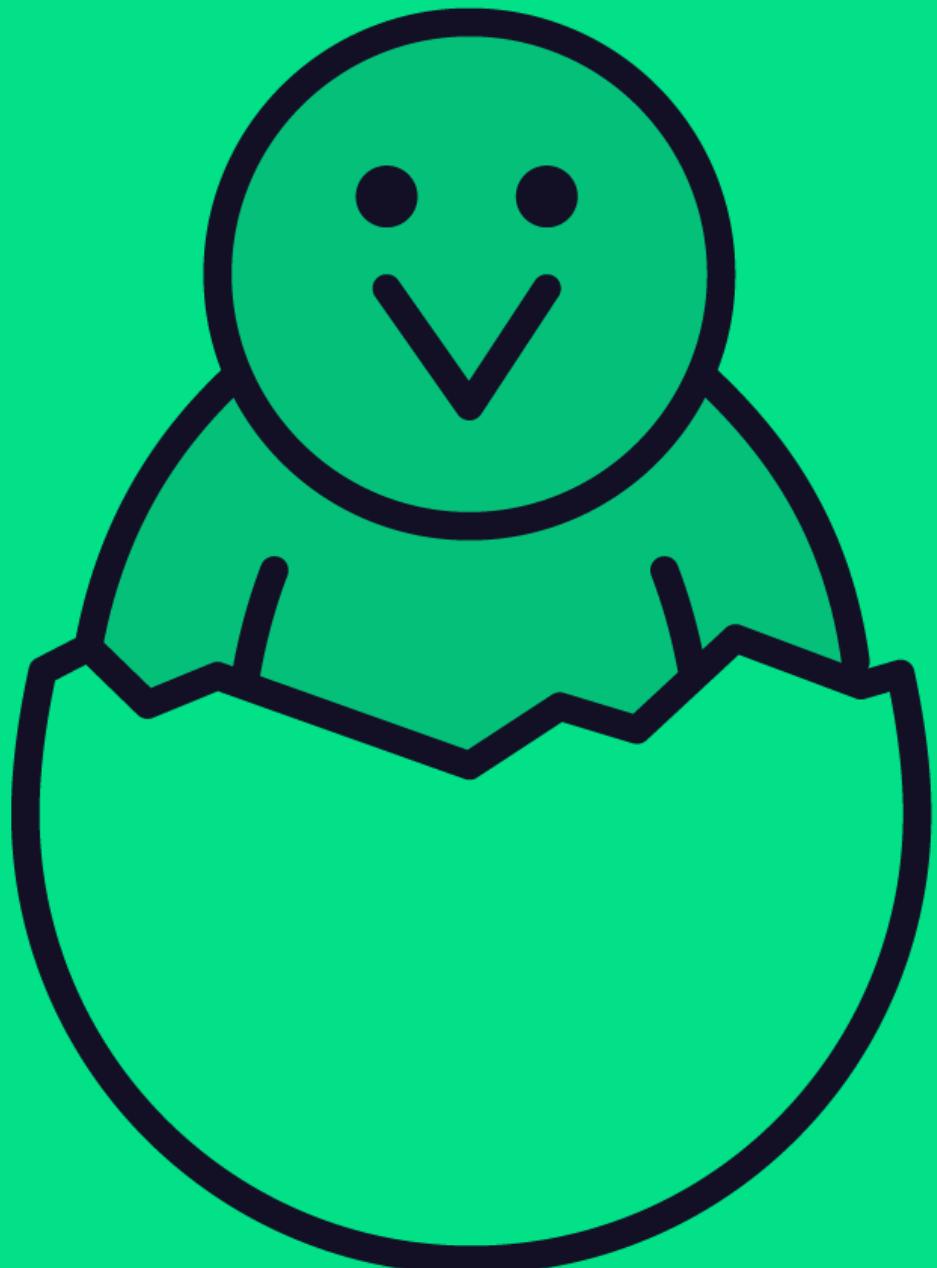
@gillcleeren



**Learning C# is a great choice!**

**C# is part of .NET**





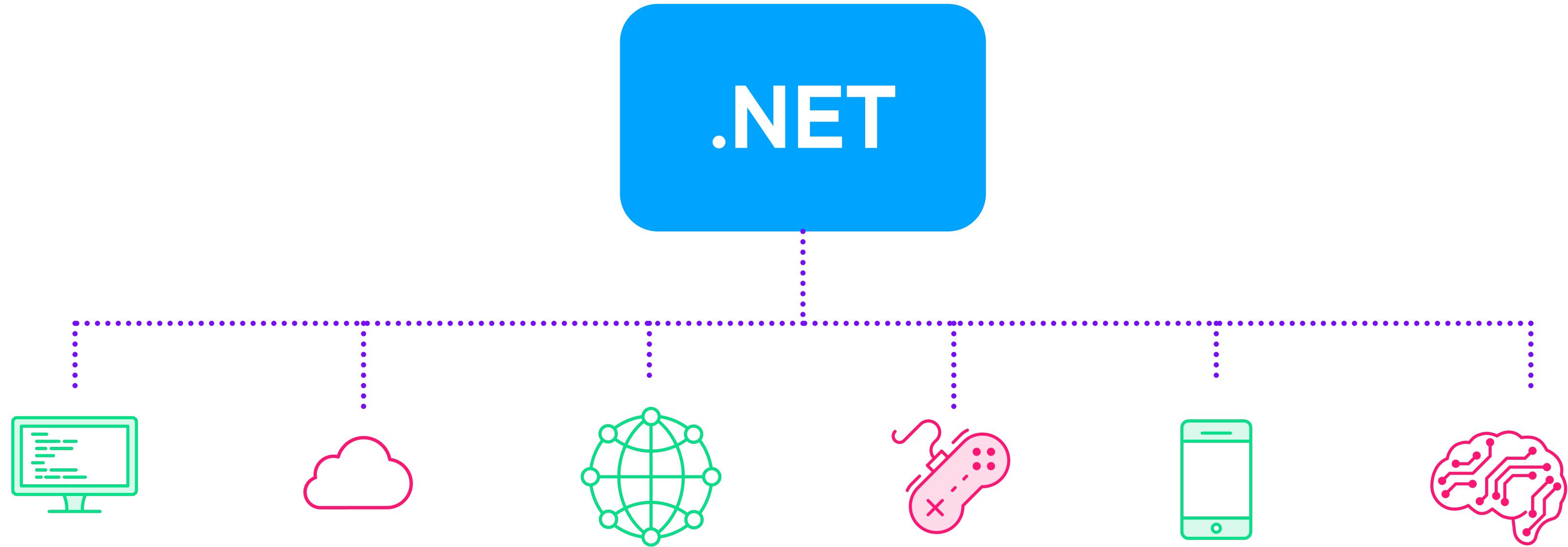
**Hello .NET**

**Developer platform**

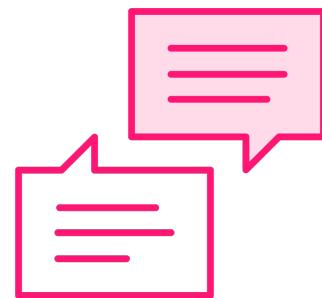
**Open source**



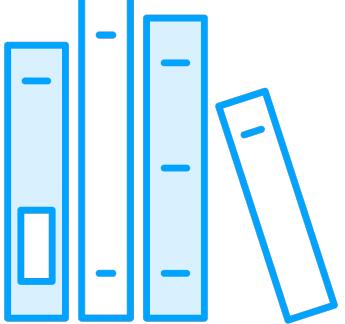
# The Unification of .NET



# The .NET Ecosystem



Languages



Libraries



Tools





.NET 8

Cross-platform





# The Languages of .NET

C#

Visual Basic

F#





No C# knowledge  
is required!



# What Is Programming?

★★★★★ By Simon Allardice

This is an accessible, language-agnostic course on the fundamentals of programming. Learn how to get started programming - in any language.



## Bethany's Pie Shop HR

- HR Management software

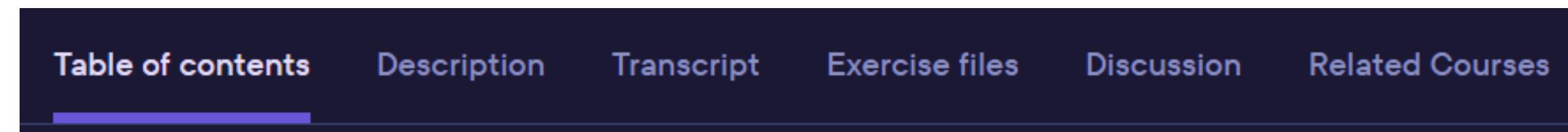




Follow along  
with everything I'm  
doing!



# Downloads for This Course



# Version Check



**This version was created by using:**

- C# 12
- .NET 8
- Visual Studio 2022 Community Edition



# Version Check



**This course is 100% applicable to:**

- C# 10, C# 11, C# 12 and C# 13
- .NET 6, .NET 7, .NET 8 and .NET 9
- Visual Studio 2022 (any edition)

**Most code will work fine with older versions of C#**



# Overview



**Hello C#**

**Setting up your environment**

**Building your first C# program**

**Debugging our code**

**Building applications using VS Code and the CLI**

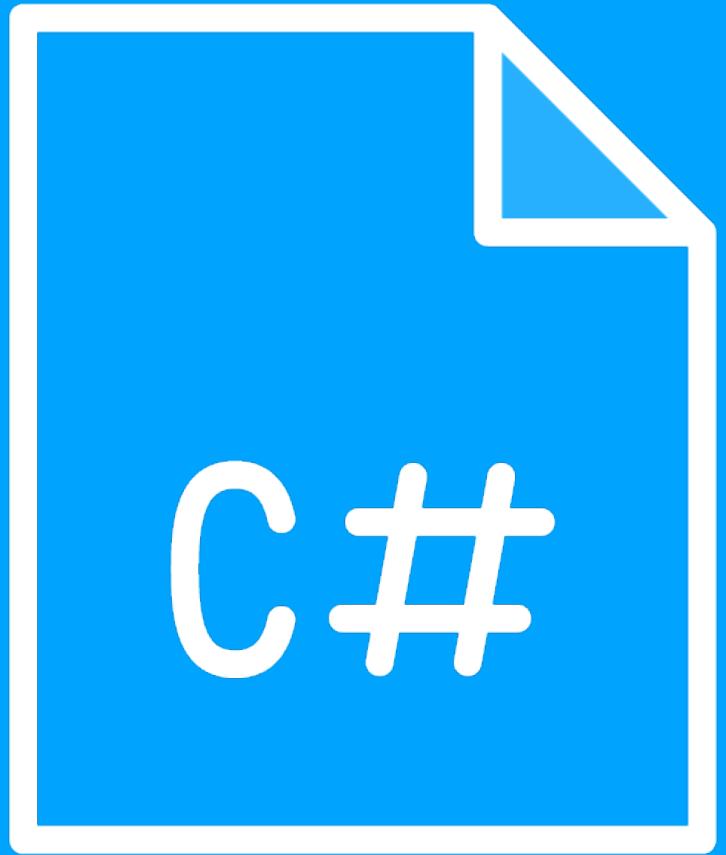
**Browsing the docs**





**Hello C#**





# Hello C#

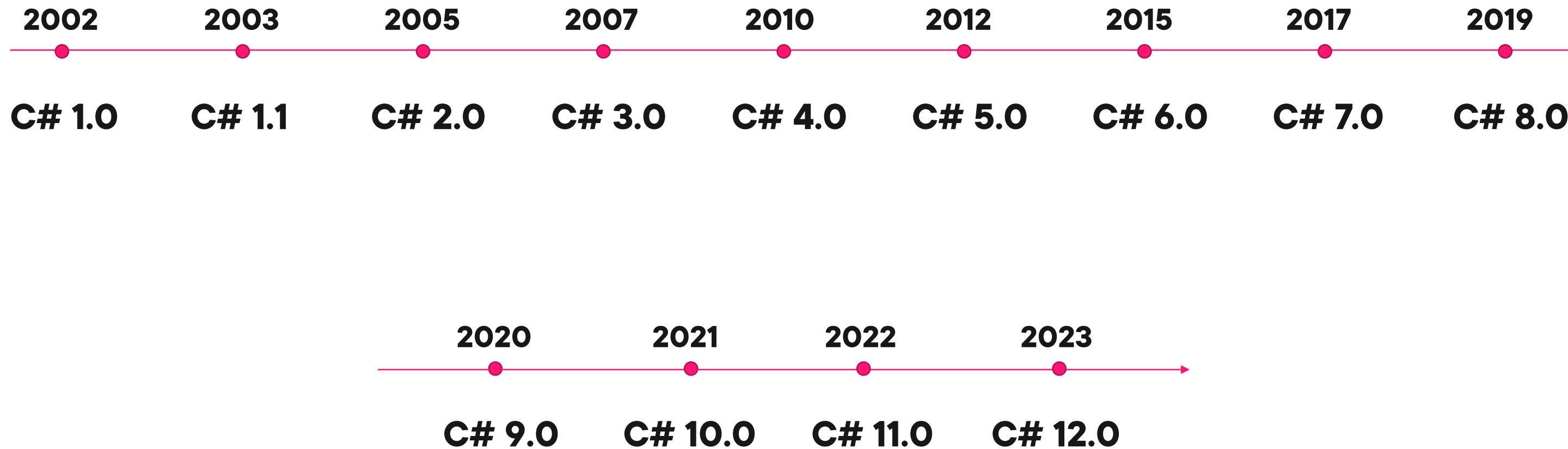
Main language for .NET development

Object-oriented and type-safe  
programming language

Familiar for C, C++, Java and JS  
developers



# C# Is Actively Maintained



# Hello C#

**Console applications**

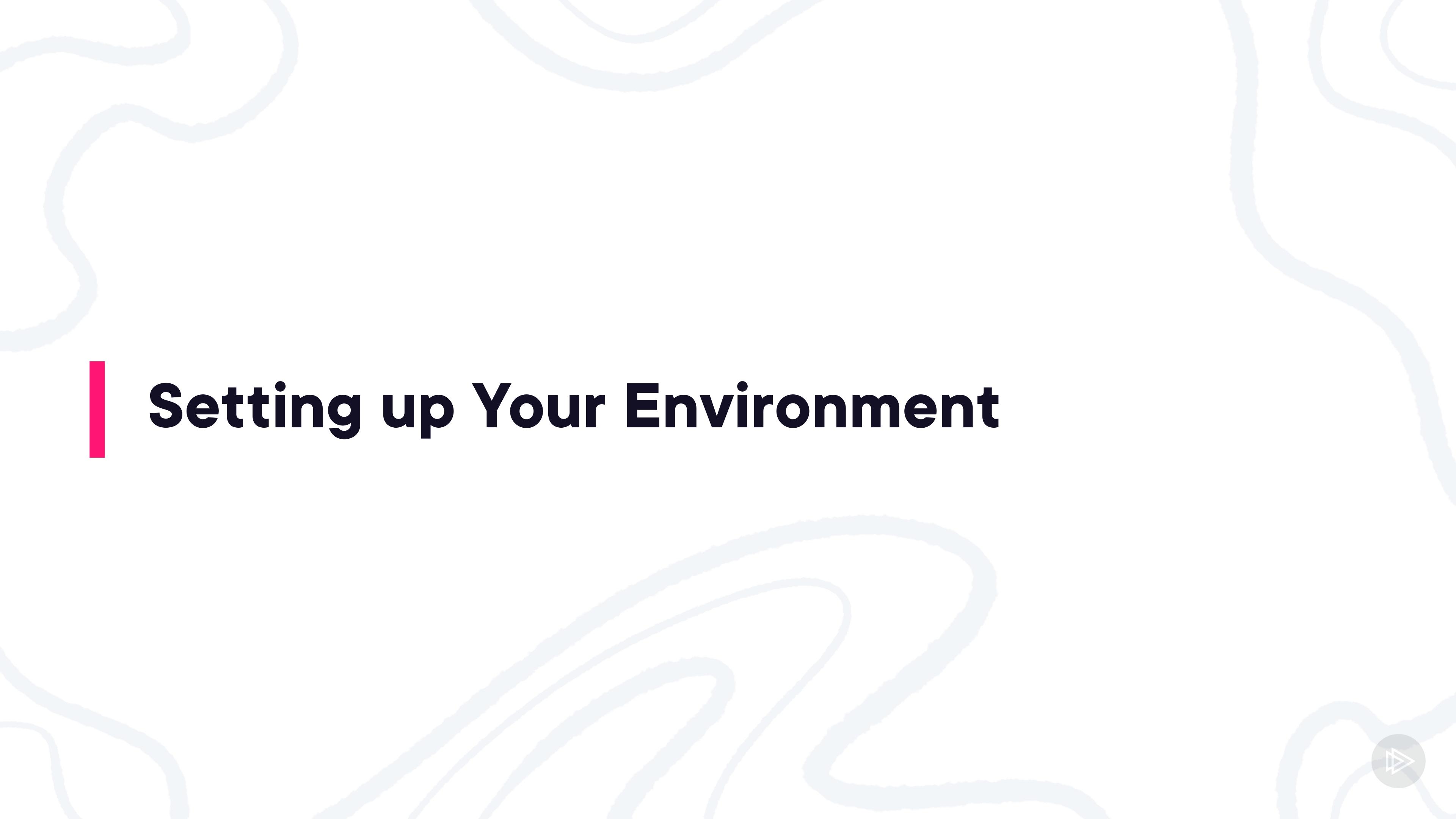
**Desktop applications**

**Web applications**

**Mobile applications**

**Services**





# Setting up Your Environment

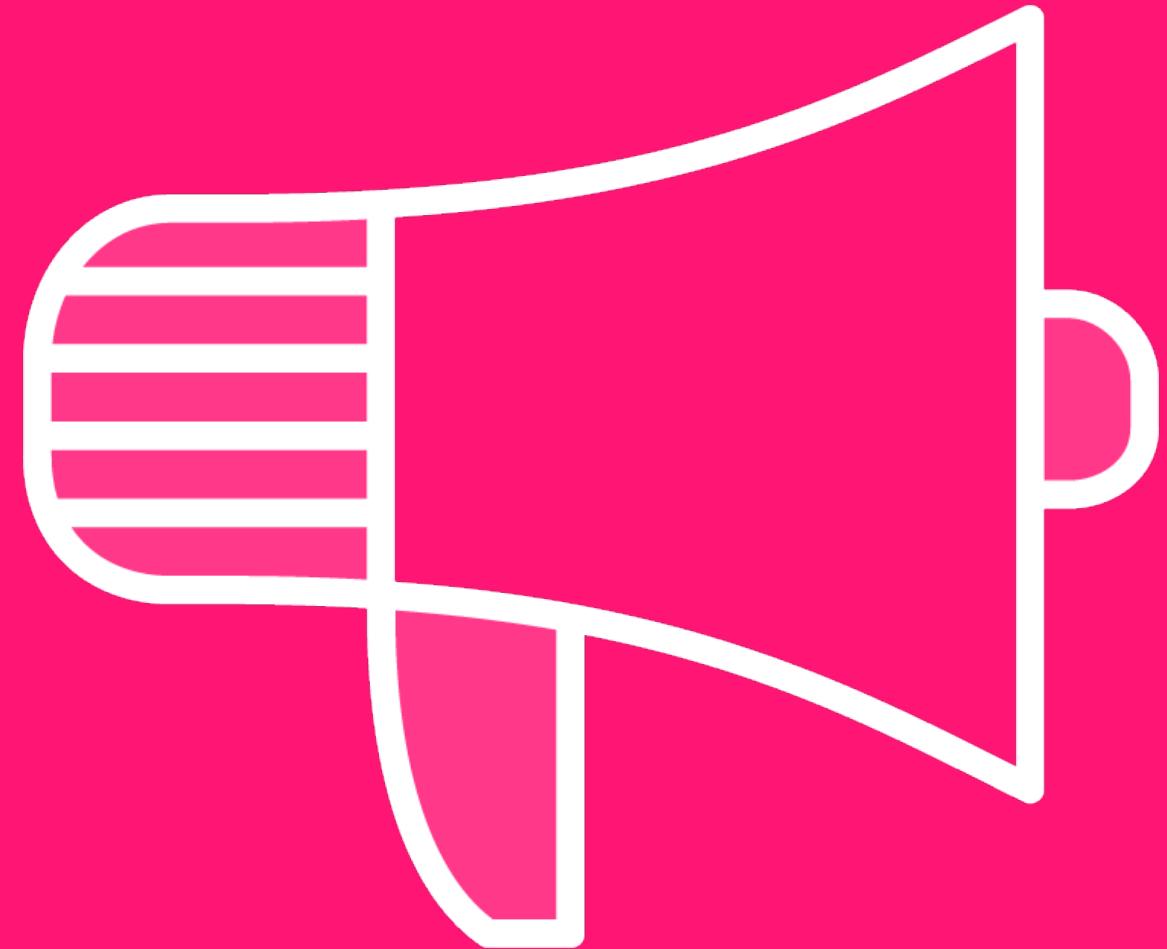


# Building .NET Applications with C#

Visual Studio 2022  
Windows

.NET CLI &  
Visual Studio Code  
All platforms





# Introducing Visual Studio 2022

**Flagship IDE  
(Integrated Development Environment)**

**Many features**



# Visual Studio 2022 Editions

**Community**

Free to use

**Professional**

Paid  
Aimed at smaller  
teams

**Enterprise**

Paid  
Aimed at larger  
teams



# Demo



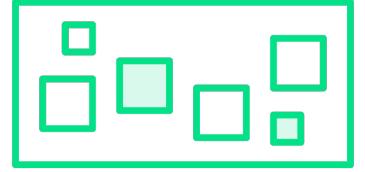
**Setting up your environment  
for C# development**



# Building Your First C# Program Using Visual Studio



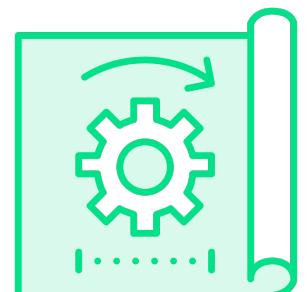
# Using Projects



**Containers for code files**



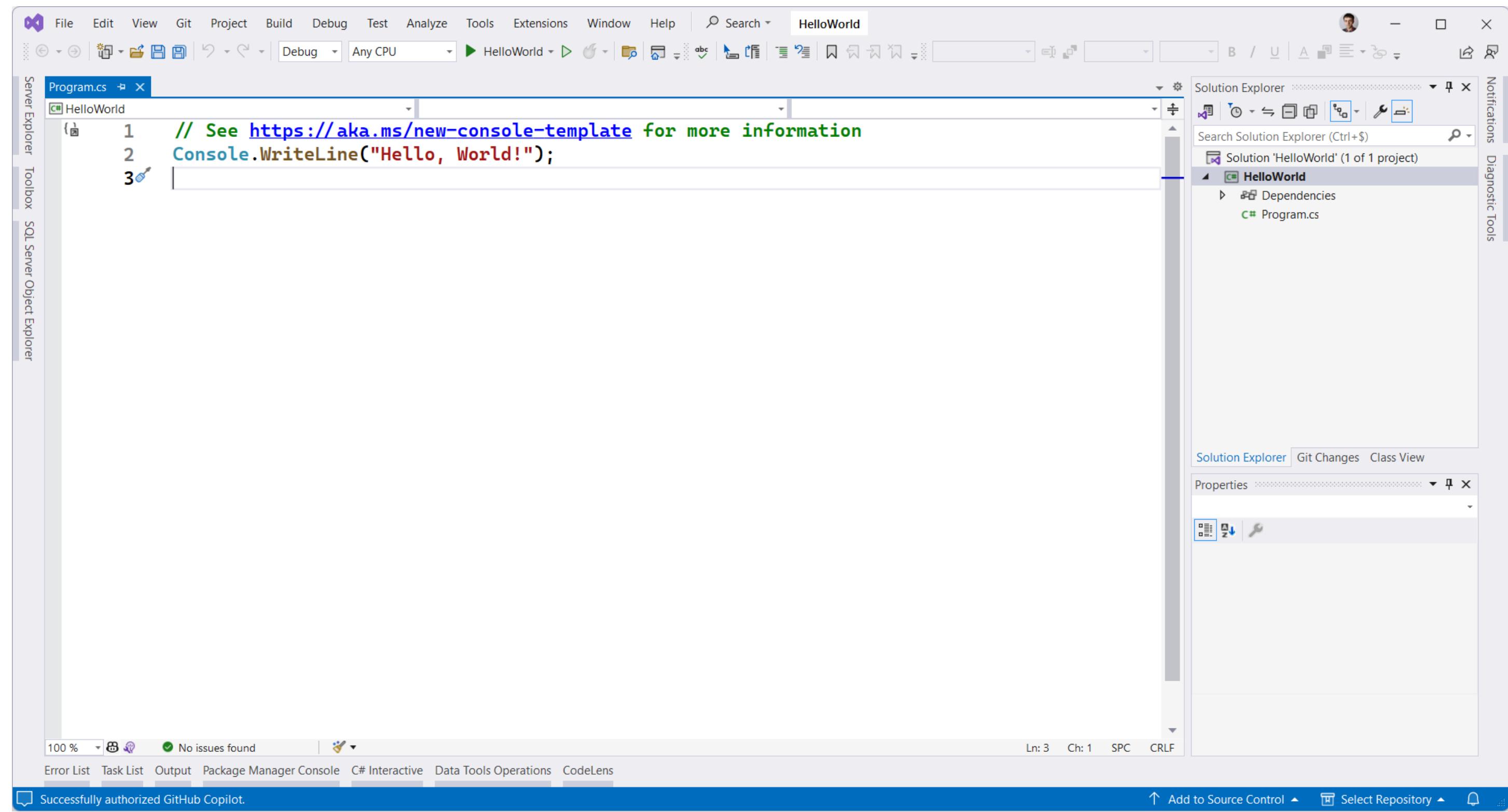
**Compiled into executable files (assembly)**



**Different templates**



# The Visual Studio 2022 Interface



# Demo



**Creating your first C# application**

**Running the application**



# Demo



**Writing C# code**

**Fixing a syntax error**



## Demo



### Looking at the generated files

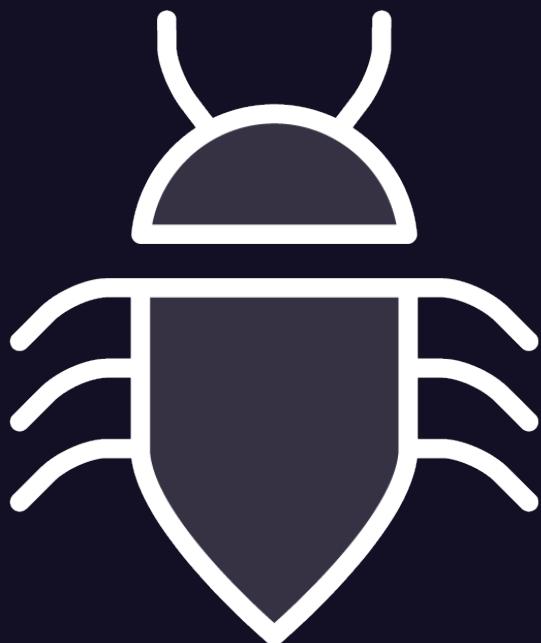
- Project
- Solution
- Executable file

### Executing our compiled application



# Debugging Our Code



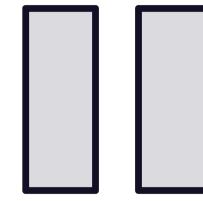


# We will make errors!

Debugging using Visual Studio



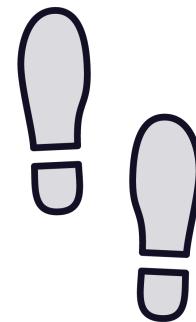
# Introducing Breakpoints



**Pause the running code**



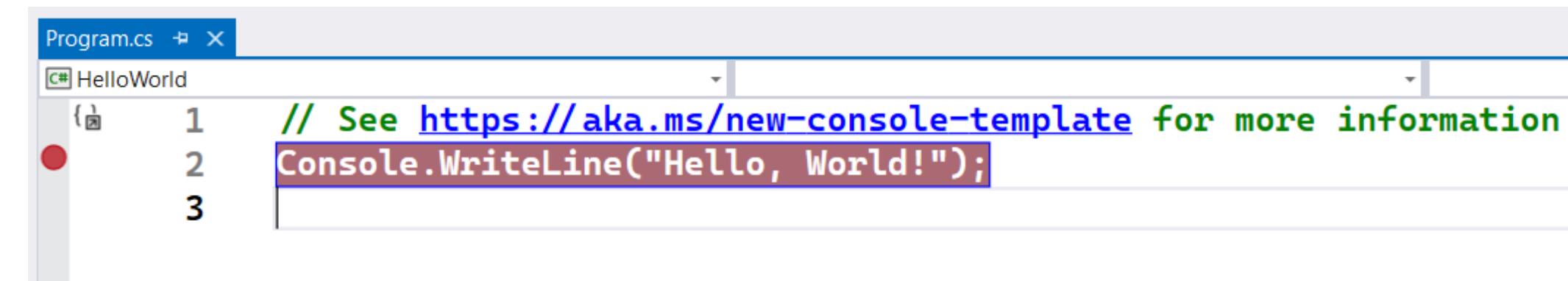
**Inspect the state of the running application**



**Step through the different lines of code**



# Adding a Breakpoint



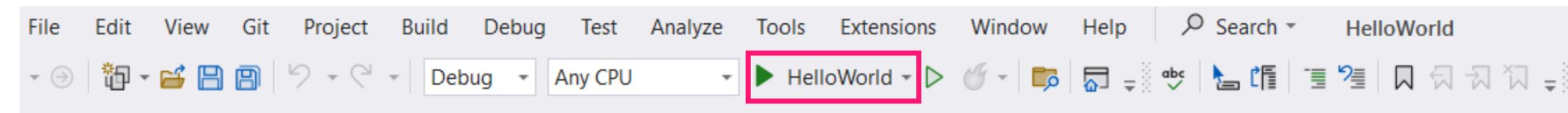
A screenshot of a code editor window titled "Program.cs". The code file is named "HelloWorld". The content of the file is:

```
1 // See https://aka.ms/new-console-template for more information
2 Console.WriteLine("Hello, World!");
3
```

A red circular breakpoint icon is positioned next to the first line of code (line 1). The line "Console.WriteLine("Hello, World!");" is highlighted with a blue selection bar.



# Running with the Debugging Attached



# Demo

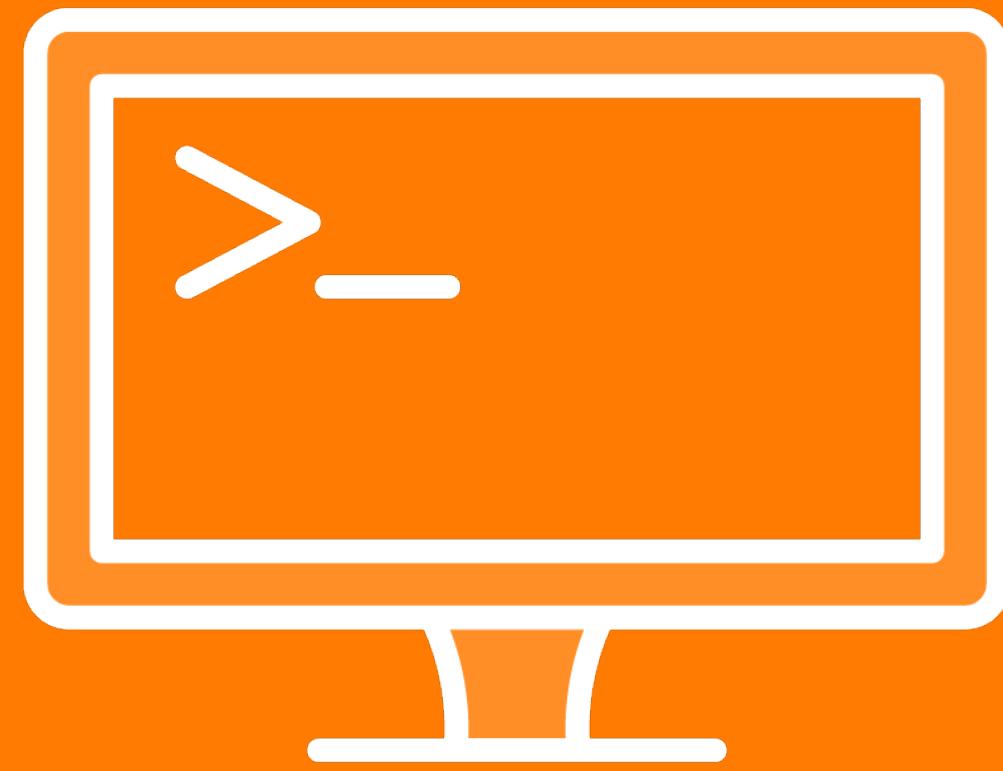


## Understanding the debugger



# Building applications using VS Code and the CLI





# Understanding the CLI

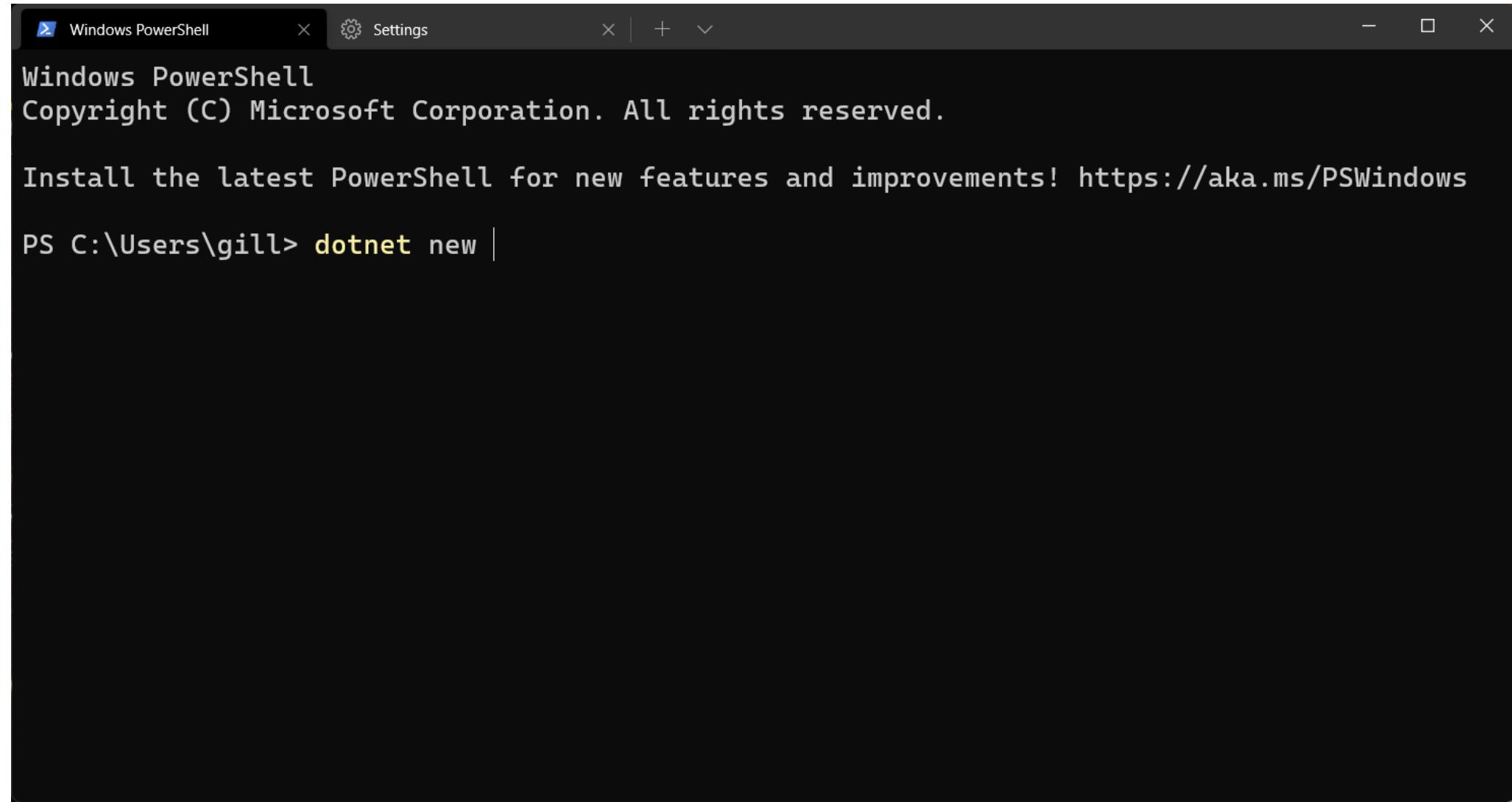
Command-line interface for .NET

Cross-platform tools

“dotnet” command with parameters



# Understanding the CLI



A screenshot of a Windows PowerShell window titled "Windows PowerShell". The window shows the standard PowerShell startup message: "Windows PowerShell", "Copyright (C) Microsoft Corporation. All rights reserved.", and a link to install the latest PowerShell. Below this, a command is being typed: "PS C:\Users\gill> dotnet new |". The window has a dark theme with white text.



# Creating a New Project

Create a new app                      Name of app

**dotnet new console -n "FirstProgram"**

Type of app



```
dotnet new
```

```
dotnet build
```

```
dotnet run
```

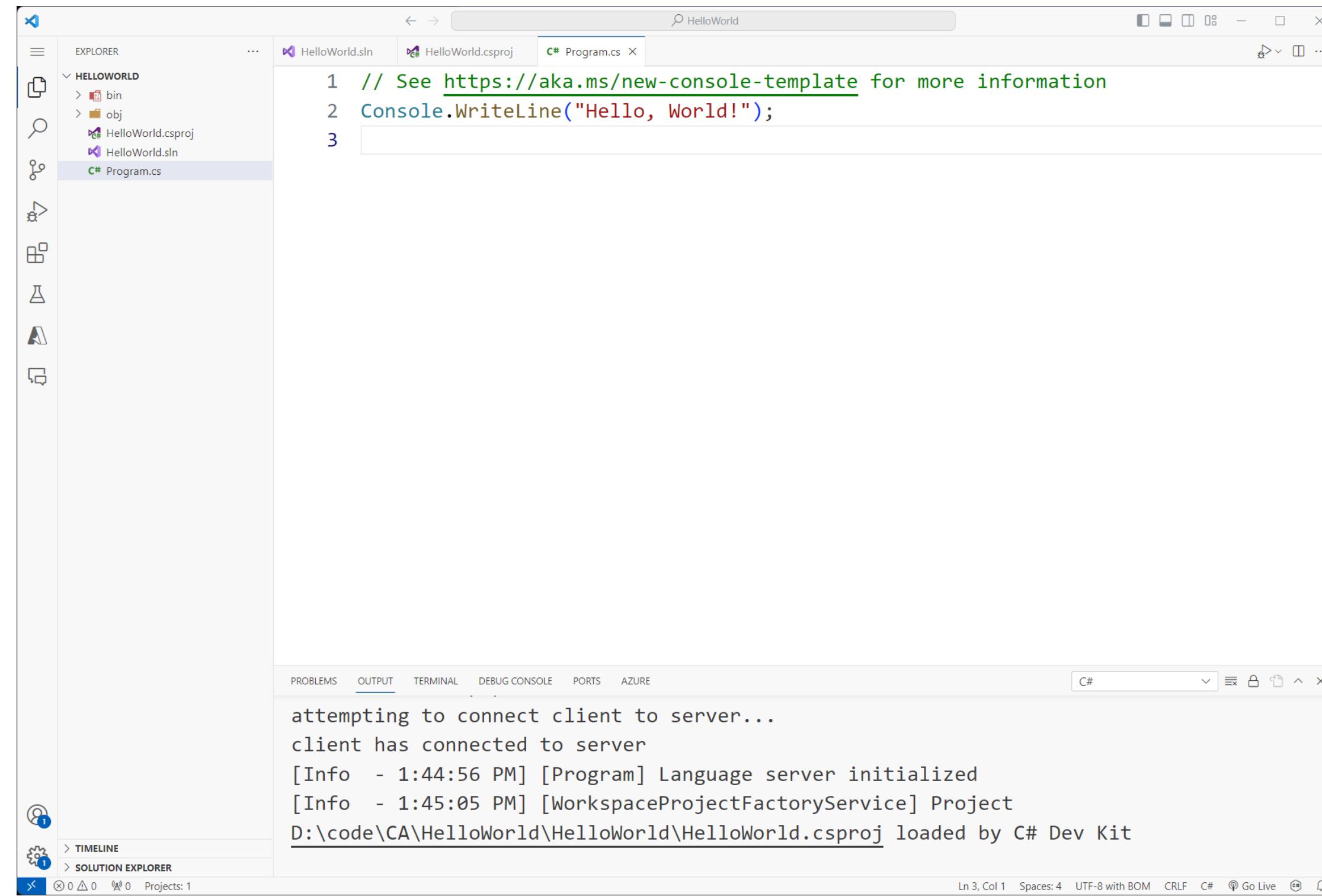
◀ **Create a new project**

◀ **Compile your application**

◀ **Execute your application**



# Writing Code with Visual Studio Code



```
// See https://aka.ms/new-console-template for more information
Console.WriteLine("Hello, World!");

```

attempting to connect client to server...  
client has connected to server  
[Info - 1:44:56 PM] [Program] Language server initialized  
[Info - 1:45:05 PM] [WorkspaceProjectFactoryService] Project  
D:\code\CA\HelloWorld\HelloWorld\HelloWorld.csproj loaded by C# Dev Kit



## Demo



**Installing the required tools**

**Creating an application using the CLI**

**Editing code with VS Code**

**Compiling using the CLI**

**Running with VS Code and the CLI**



# Using C# Dev Kit in VS Code

The screenshot shows the Visual Studio Code interface with the Marketplace extension search results for "C# Dev Kit".

**Left Sidebar:** Shows the Extensions Marketplace search results for "C#". The top result is "C# Dev Kit" by Microsoft, which is highlighted.

**Top Bar:** Shows the VS Code header with tabs like File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar containing "HelloWorldFromCLI".

**Extension Details Page:**

- Extension Name:** C# Dev Kit (v1.2.5, Pre-Release)
- Developer:** Microsoft ([microsoft.com](#))
- Downloads:** 1,708,814
- Rating:** ★★★★☆ (130)
- Description:** Official C# extension from Microsoft
- Status:** This extension is enabled globally.
- Actions:** Disable, Uninstall, Switch to Release Version

**Content Area:**

- Title:** C# Dev Kit for Visual Studio Code
- Text:** C# Dev Kit helps you manage your code with a solution explorer and test your code with integrated unit test discovery and execution, elevating your C# development experience wherever you like to develop (Windows, macOS, Linux, and even in a Codespace).
- Text:** This extension builds on top of the great C# language capabilities provided by the C# extension and enhances your C# environment by adding a set of powerful tools and utilities that integrate natively with VS Code to help C# developers write, debug, and maintain their code faster and with fewer errors. Some of this new tooling includes but is not limited to:

  - C# project and solution management via an integrated solution explorer
  - Native testing environment to run and debug tests using the Test Explorer
  - Roslyn powered language service for best in-class C# language features such as code navigation, refactoring, semantic awareness, and more
  - AI-Assisted development

- Section:** Quick Start
- Text:** 1. Install C# Dev Kit (The C# extension, the IntelliCode for C# Dev Kit, and the .NET Install Tool will automatically be installed)  
2. Open a folder/workspace that contains a C# project (.csproj) and the extension will activate.  
3. Check out the [C# Getting Started](#) documentation to learn more!
- Section:** Feature List & Walkthrough
- List:**
  - Project System / Solution Explorer
    - Solution Node Actions
    - Add Project
    - Build/Run Project
  - Code Editing (Uses the C# extension)
    - Refactoring
    - Code Navigation (Go To Definition/References)
    - Code Completions
    - Roslyn powered semantic awareness
  - Package Management
    - Automatic NuGet Restore

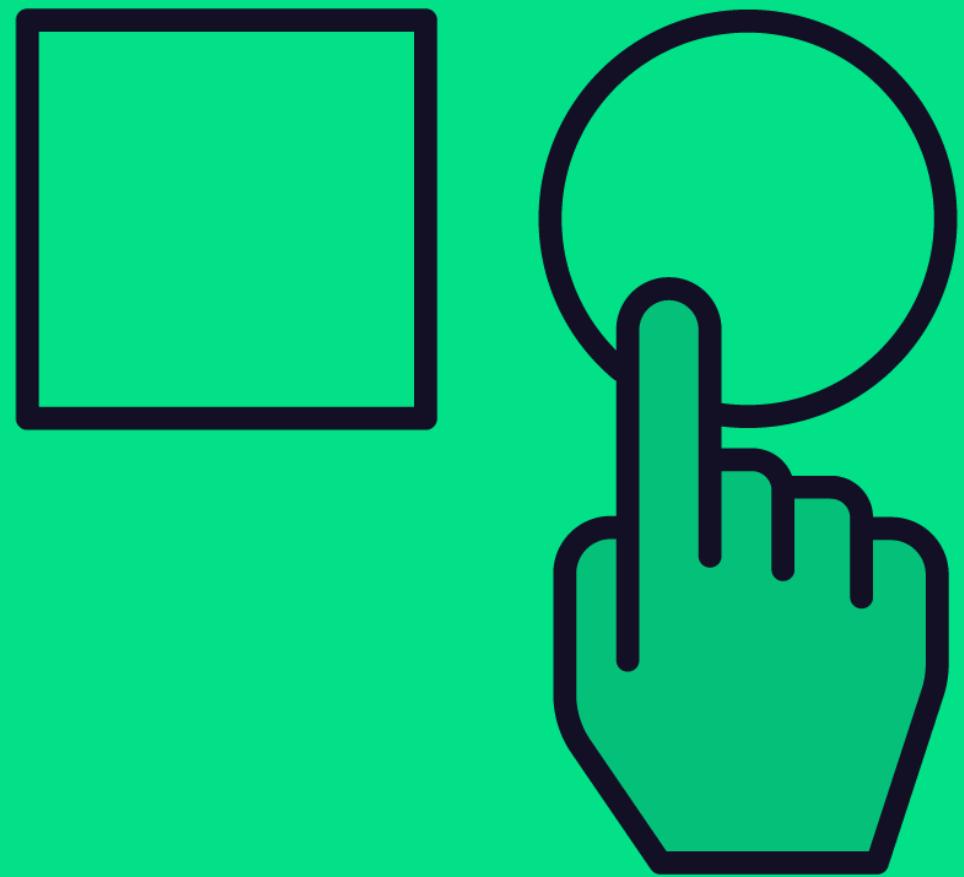
## Demo



### Using the C# Dev Kit in VS Code

- Creating a new project
- Writing C# code
- Running and debugging our code





## The choice is yours!

No matter what platform, you can  
write C# applications!





# Browsing the Docs



# Demo



## Finding information in the docs



## Summary



**C# is an object-oriented and type-safe language to write .NET applications**

**C# is actively maintained**

**C# can be used to build all types of .NET applications**

**Applications can be created using**

- Visual Studio
- VS Code and CLI



**Up Next:**

# **Learning the C# syntax**

---

