

# Introduction to C#

## C# Programming

## In this chapter we will learn ...

- What C# and .NET are
- To install .NET on any platform
- To install VSCode
- To create and build simple C# Console App

# Introduction

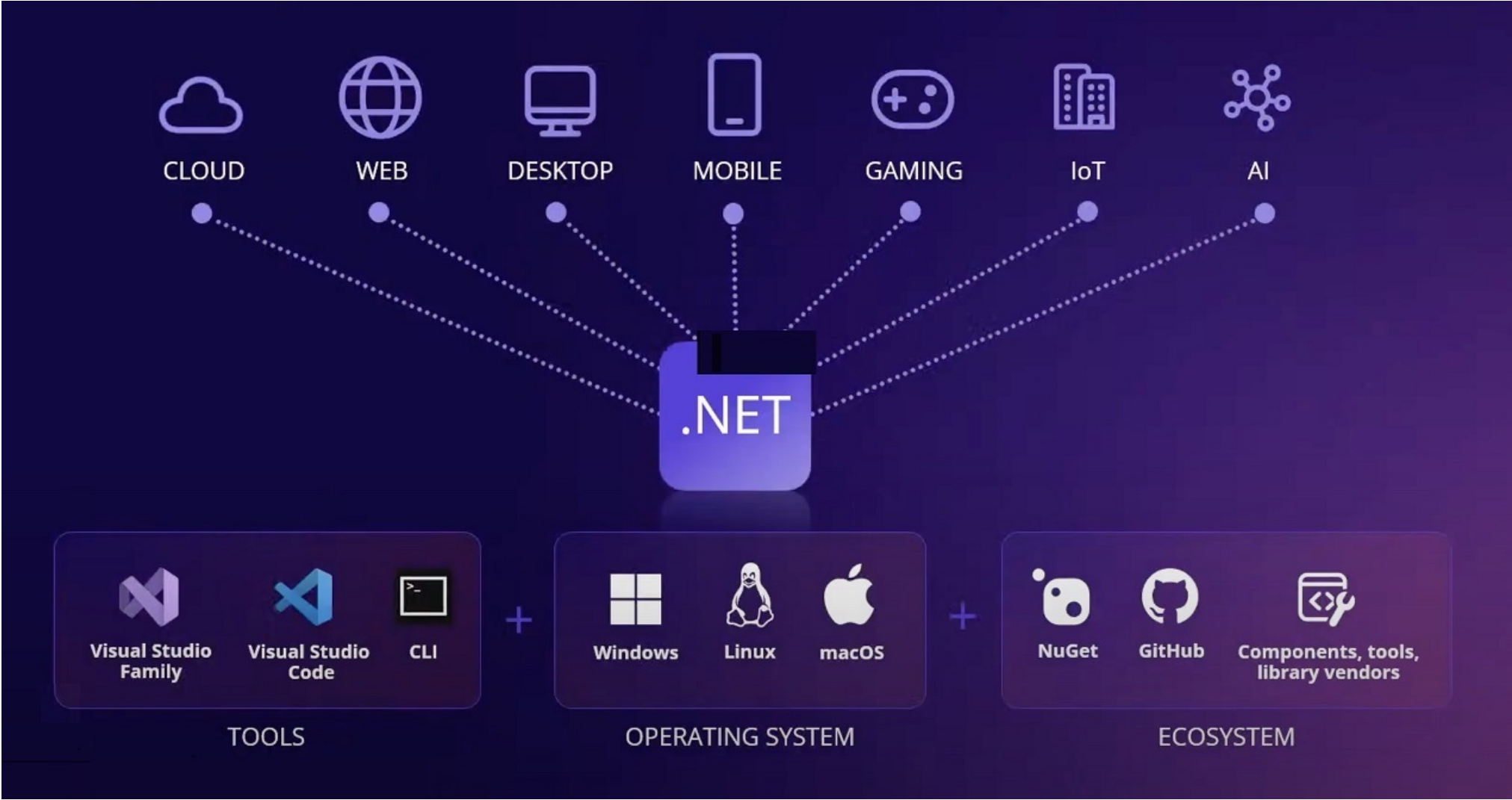
# Question

**What do we know about .NET?**

- Discuss



# What is .NET

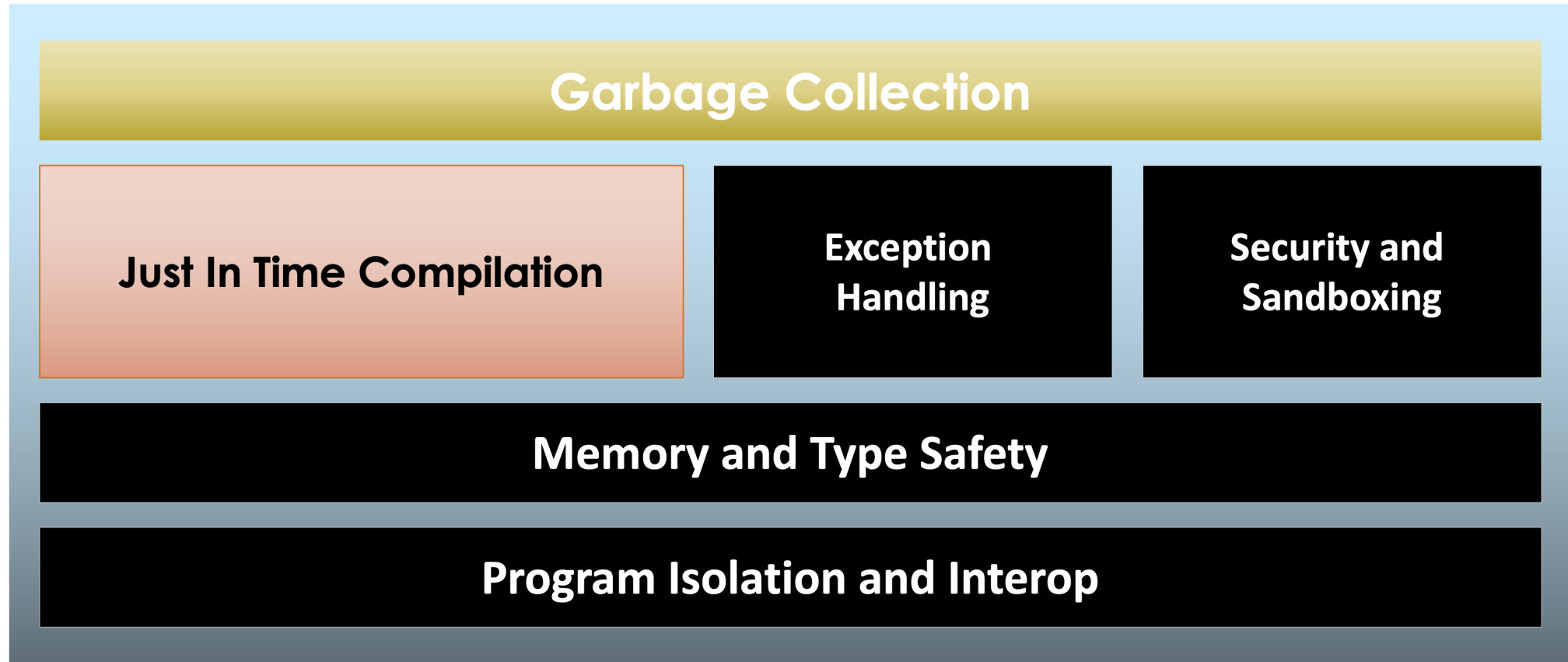


# What is .NET - Continued

## **.NET is the combination of**

- Extensive Re-Usable Base Class Library (BCL)
  - .NET Framework BCL,
  - .NET 5, 6 BCL
- Common Language Runtime (CLR) Environment
  - .NET Framework CLR
  - .NET 5, 6 CLR
- Command Line Interface
- .NET Standard
  - An API standard for the BCLs supported across platforms

# Core CLR



# Installing .NET

At least two options

- Download and install Visual Studio 2022 Community or
- Goto link below, download and install latest .NET for your platform (Windows, OSX, Linux)  
<http://dotnet.microsoft.com>
- Download and Install Visual Studio Code for your platform (Windows, OSX, Linux)



# Visual Studio 202?

**Is a**

- Fully Integrated Development Environment
- Lots of tools and helpers
- Build and Debug
- Integrates with Azure Devops
- High Quality Code Editing with Code Snippets and Intellisense

**It's not really Cross Platform**

- Best Version is for PC (is available for Mac)

# Visual Studio Code

## Is a

- Very Powerful Editor
- Support can be added for almost any language
- Debug and Build capabilities can be configured
- Lots of Extensions can be added to enhance capabilities

## It's Cross Platform

- Support Windows, Linux, OSX

# Using the CLI with .NET

## Creating a new project

- Below creates a console application (other app types are available)

```
dotnet new console
```

## Building and running

```
dotnet build // Build only  
dotnet run  // Build and Run  
dotnet test // Build and run Unit tests
```

# Projects and Solutions

**Project File** defines all of the project settings

- Has a .csproj (or .proj) extension
- Contains links to other projects, Services and Packages
- Stores build settings

**Solution** defines a collection of projects

- Has .sln extension
- Useful when used with Visual Studio as opening a SLN file load all projects at once

# Question

**What do we know about C#?**



## A bit about C#

C# Is case sensitive

**Uses** { and }  
to define code blocks

**Requires** ; to  
define the end  
of statements

C# (like C)  
programs have  
a Main function  
that are the  
staring point of  
the program

# First C# App

```
using System;

namespace FirstApp {
    class Program {

        public static Main(string[] args) {
            Console.WriteLine("Hello World");
        }
    }
}
```

- using - We will be using functions/classes from System namespace
- namespace - Unique group name for our types. Program is actually **FirstApp.Program**
- class - Defines a new data type called Program

## First C# App - since .NET 5.0 (Top level modules)

```
using System;  
  
Console.WriteLine("Hello World");
```



# Simple Console App

What is going on here and why? ...

```
using System;  
  
Console.Write("What is your name? ");  
  
string firstName = Console.ReadLine();  
  
Console.WriteLine($"Hello {firstName}");
```

## In this chapter we learned ...

- What C# and .NET are
- To install .NET on any platform
- To install VSCode
- To create and build simple C# Console App