Block 1: Lektion 1/5

Intro, utvecklingsmiljö, grunder

Introduktion

Dagens upplägg

- Vad är C#
- Vad är .NET?
- Vad är Visual Studio?
- Dotnet CLI?
- C# Grunder
- Att Göra/Lästips

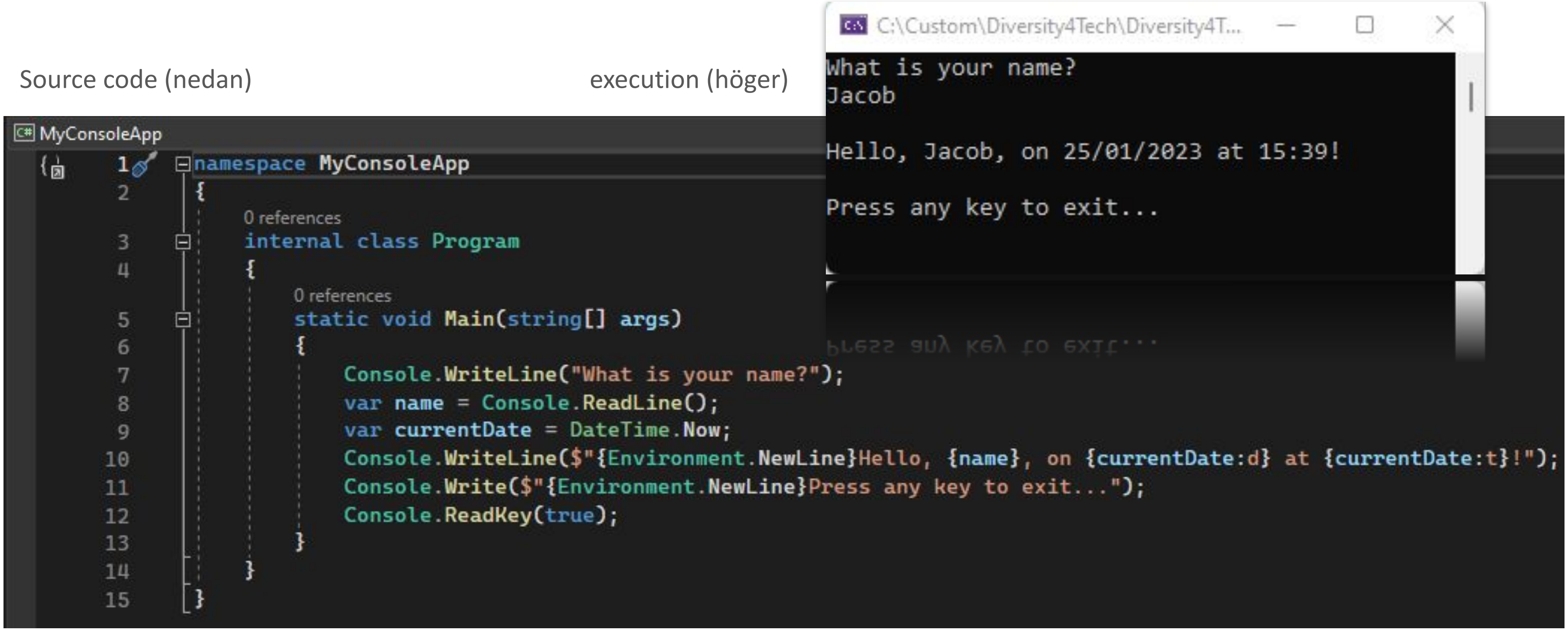
Vad är C# (CSharp)

- **Open-source:** Free to use, study, modify and contribute to.
- **Cross-platform:** Runs on any system with architecture/runtime specific .NET SDK (Software Development Kit) installed.
- **General-purpose language:** mobile apps, desktop apps, webapps, cloud-based services, enterprise software, games…
- **High-level programming language:** High abstraction (human readable) for increased productivity, maintainability…
- **Supports multiple paradigms/programming disciplines:** e.g. imperative, declarative, functional, generic, object-oriented, and component-oriented
- **Static typing (when type information is supplied):** is enforced (unless using the *dynamic* keyword, type must be declared at variable/object declaration) at compile time, which ensures safe interaction, mitigates the risk of experiencing unexpected errors during runtime.
- **Strong typing (strictness):** Strict rules for casting and conversion of types are enforced at compile time. The storage space that a variable of the type requires; min/max values it can represent; members that it contains; base type inheritance; interface implementations; permitted operations
- **Lexically (statically) scoped:** Performance and speed gains through Lexical scoping variables only accessible within their defined scope (where they were declared)
- **Managed code:** Requires a CLI (Common Language Infrastructure) to manage execution in platform specific runtime environment - NET 7.0 provides the CLI for C# 11.0

.NET 7.0 Console App (written in C#)

Source code (nedan)

execution (höger)



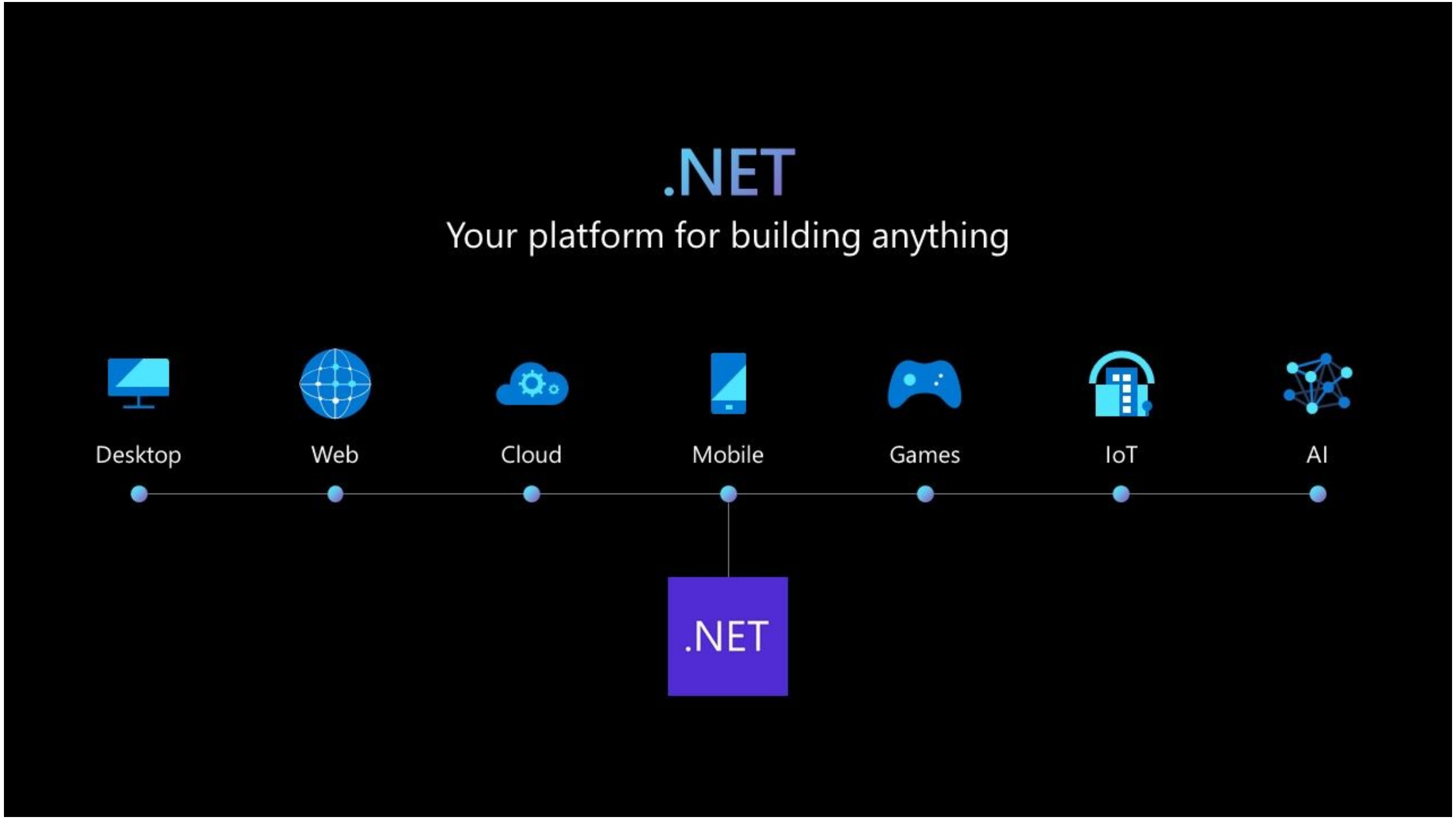
```csharp
namespace MyConsoleApp
{
    internal class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("What is your name?");
            var name = Console.ReadLine();
            var currentDate = DateTime.Now;
            Console.WriteLine($"{Environment.NewLine}Hello, {name}, on {currentDate:d} at {currentDate:t}!");
            Console.Write($"{Environment.NewLine}Press any key to exit...");
            Console.ReadKey(true);
        }
    }
}
```

```
What is your name?
Jacob

Hello, Jacob, on 25/01/2023 at 15:39!

Press any key to exit...
```

Vad är .Net?


.NET
Your platform for building anything
Desktop
Web
Cloud
Mobile
Games
IoT
AI
.NET

.NET 5+ (Current version: 7.0.2)

Open Source, Cross-Platform, managed computer software framework written in C++ and C#

.NET Evolution

- .Net 5+ (samlingsnamn)
- .Net Core 3.1 (cross-platform)
- .Net Framework 4.8 (win)

.NET SDK contains

The .NET SDK is a set of libraries and tools that allow developers to create .NET applications and libraries. It contains the following components that are used to build and run applications:

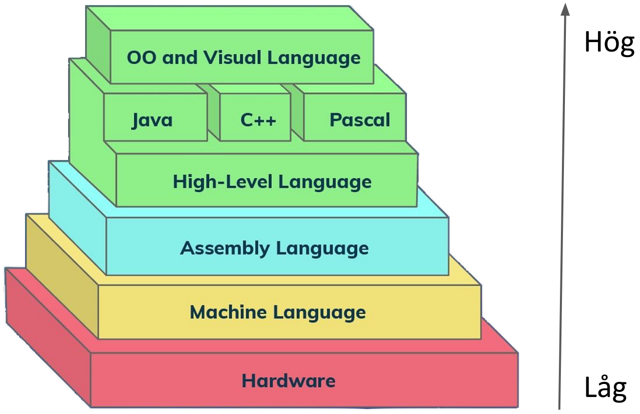
- The .NET CLI
- The .NET runtime (CLR) and libraries
- The dotnet driver

.NET features allow developers to productively write reliable and performant code.

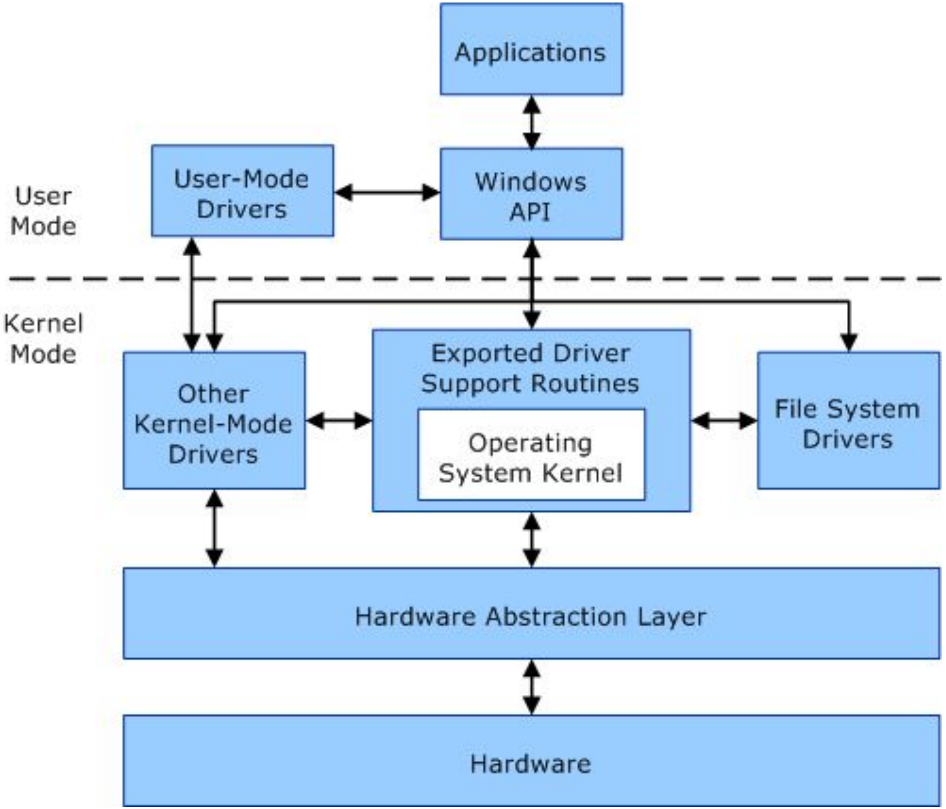
- Asynchronous code
- Attributes
- Reflection
- Code analyzers
- Delegates and lambdas
- Events
- Exceptions
- Garbage collection
- Generic types
- LINQ (Language Integrated Query).
- Parallel programming
- Type inference - C#, F#, Visual Basic.
- Type system
- Unsafe code

Så länge det finns en OS-specifik verison av .NET SDK installerad på platformen kan C# kod som skapats på en windows maskin även köras på en OSX eller Linux maskin. Detta då varje platform har rätt version av .NET runtime (för OS och CPU architecture) installerad. C# kod kompileras till IL (Intermediary language) som delvis innehåller “native machine code” som anropar .NET och låter den platformspecifika CLR (Common Language Runtime) kompilera koden till “native” instruktioner (maskin kod) som kan tolkas av platformens CPU.
Se .NET Glossary

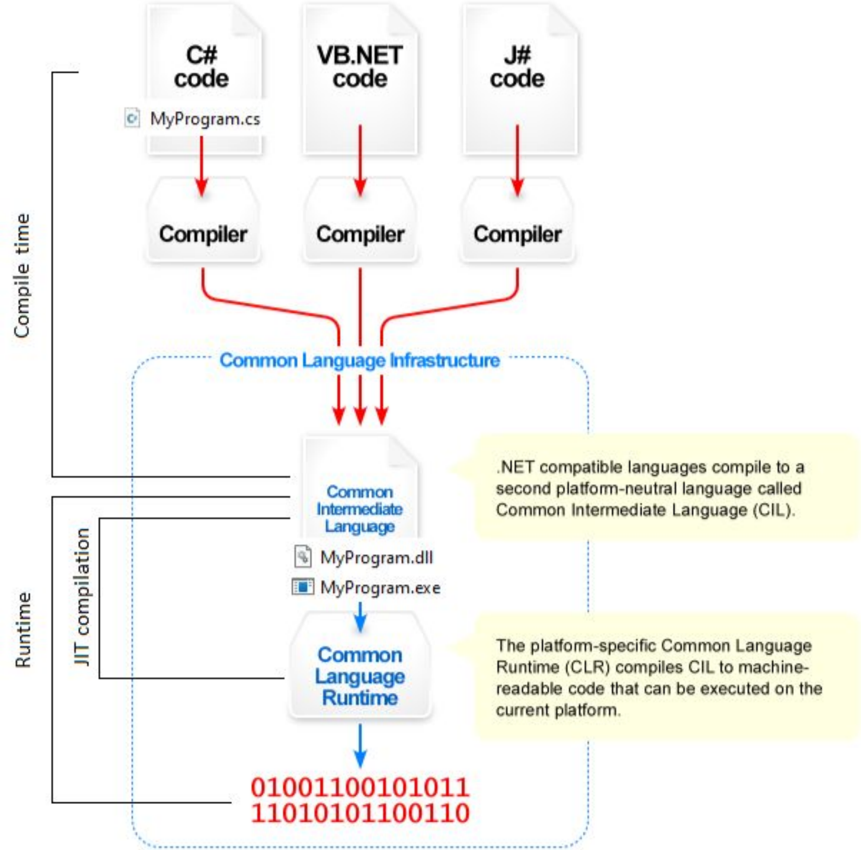
Abstraktionsnivå



Platforms exempel: Windows 11 x64 PC



Från “human readable” till “machine readable”

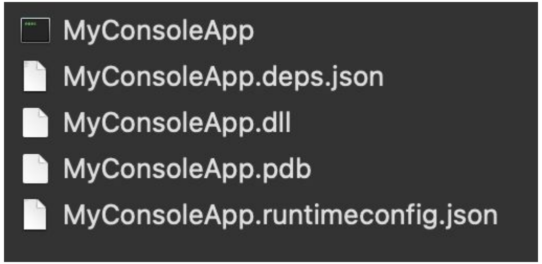


.NET compatible languages compile to a second platform-neutral language called Common Intermediate Language (CIL).

The platform-specific Common Language Runtime (CLR) compiles CIL to machine-readable code that can be executed on the current platform.

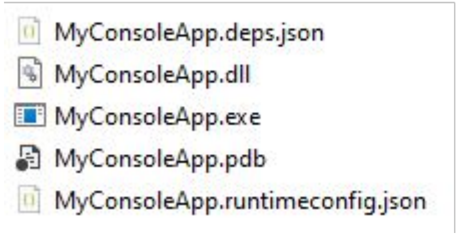
Compiler output on Unix/Win systems:

AArch64/Arm64 : Unix executable



MyConsoleApp
MyConsoleApp.deps.json
MyConsoleApp.dll
MyConsoleApp.pdb
MyConsoleApp.runtimeconfig.json

Winx64: Windows executable (.exe)



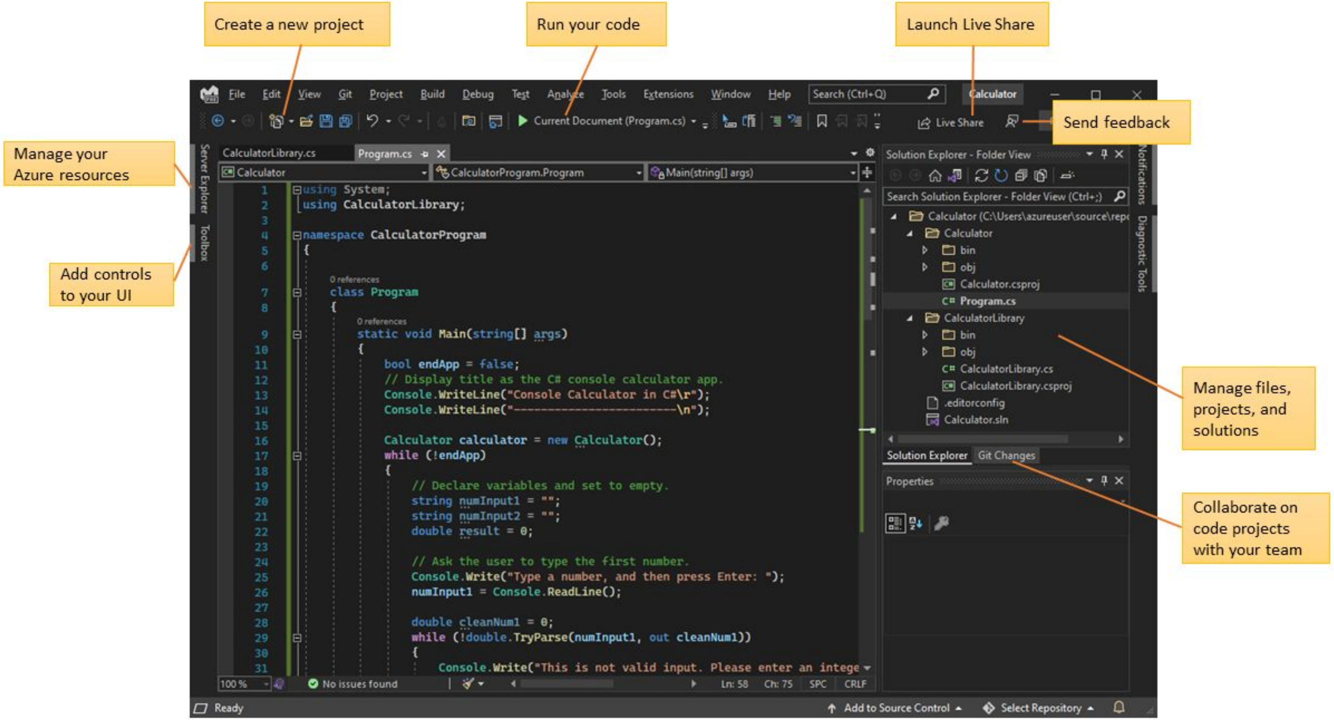
MyConsoleApp.deps.json
MyConsoleApp.dll
MyConsoleApp.exe
MyConsoleApp.pdb
MyConsoleApp.runtimeconfig.json

Rörigt?

Nu har vi börjat bygga en förståelse för vad .NET möjliggör för oss som utvecklare

Om det kändes som mycket att ta in kan vi vara extra glada över att Visual Studio (IDE) låter oss koda på utan att tänka så mycket på dessa aspekter
Men vad är då Visual Studio?

Utvecklingsmiljö - Integrated Development Environment (IDE)



Create a new project

Run your code

Launch Live Share

Send feedback

Manage your Azure resources

Add controls to your UI

Manage files, projects, and solutions

Collaborate on code projects with your team

```
using System;
using CalculatorLibrary;

namespace CalculatorProgram
{

    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            bool endApp = false;
            // Display title as the C# console calculator app.
            Console.WriteLine("Console Calculator in C#\r");
            Console.WriteLine("------------------------\n");

            Calculator calculator = new Calculator();
            while (!endApp)
            {
                // Declare variables and set to empty.
                string numInput1 = "";
                string numInput2 = "";
                double result = 0;

                // Ask the user to type the first number.
                Console.Write("Type a number, and then press Enter: ");
                numInput1 = Console.ReadLine();

                double cleanNum1 = 0;
                while (!double.TryParse(numInput1, out cleanNum1))
                {
                    Console.Write("This is not valid input. Please enter an intege
```

Microsoft Visual Studio 2022 (IDE)

Ett enda program med de verktyg och funktioner vi behöver för att effektiv utveckling .

Öka produktivitet genom att reducera manuell installation och integration av verktyg; förenkla och förbättra utvecklingsuplevelsen genom bl.a. debugging, varningar, felmeddelanden, scaffolding via GUI och integrerad terminal.

Några exempel på vad Visual Studio 2022 erbjuder (IDE documentation)

- Source code editor (Editor)
- Organise code in solutions and projects (Solution Explorer)
- Git & GitHub integration (Team Explorer)
- Command Line Interface (.NET CLI)
- Package Manager (.NuGet)
- Code completion tool (IntelliSense & IntelliCode)
- Build automation tools (MSBuild)
- Debugger (VS Debugger)

Att göra/Lästips

- https://www.w3schools.com/cs/index.php
- https://learn.microsoft.com/en-us/training/paths/csharp-first-steps/
- Work with data in C# - Training | Microsoft Learn
- Getting started with Visual Studio
- Create a .NET console application using Visual Studio - .NET | Microsoft Learn

Produktivitets tips:

- How to move around in the IDE - Visual Studio (Windows) | Microsoft Learn
- Quick Actions, light bulbs, and screwdrivers - Visual Studio (Windows) | Microsoft Learn
- C# IntelliSense - Visual Studio (Windows) | Microsoft Learn
- Code navigation commands - Visual Studio (Windows) | Microsoft Learn