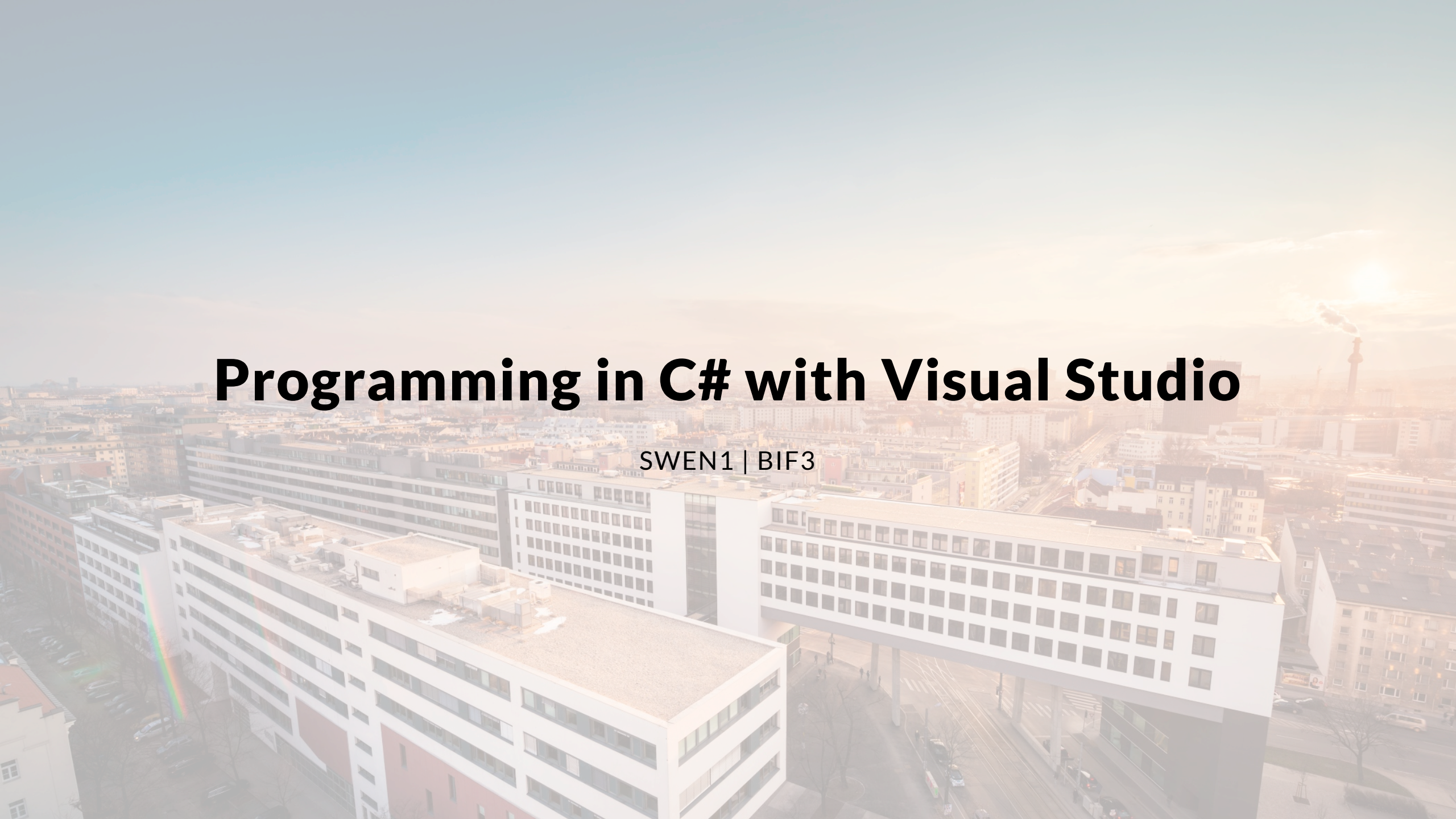


Programming in C# with Visual Studio

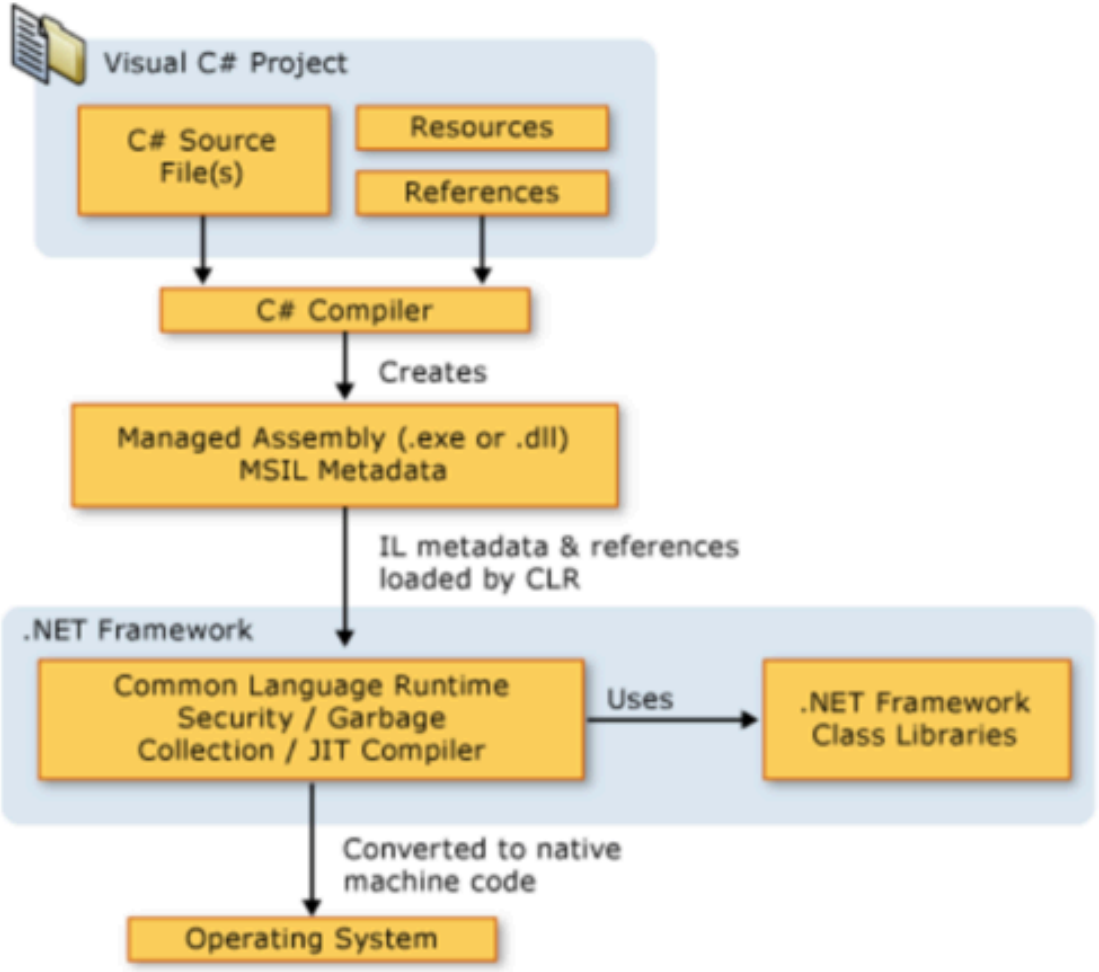
SWEN1 | BIF3

Compiling Code

The Big Picture

Building in Command line with .NET cli

- `dotnet new console`
- `dotnet build`
- `dotnet run`

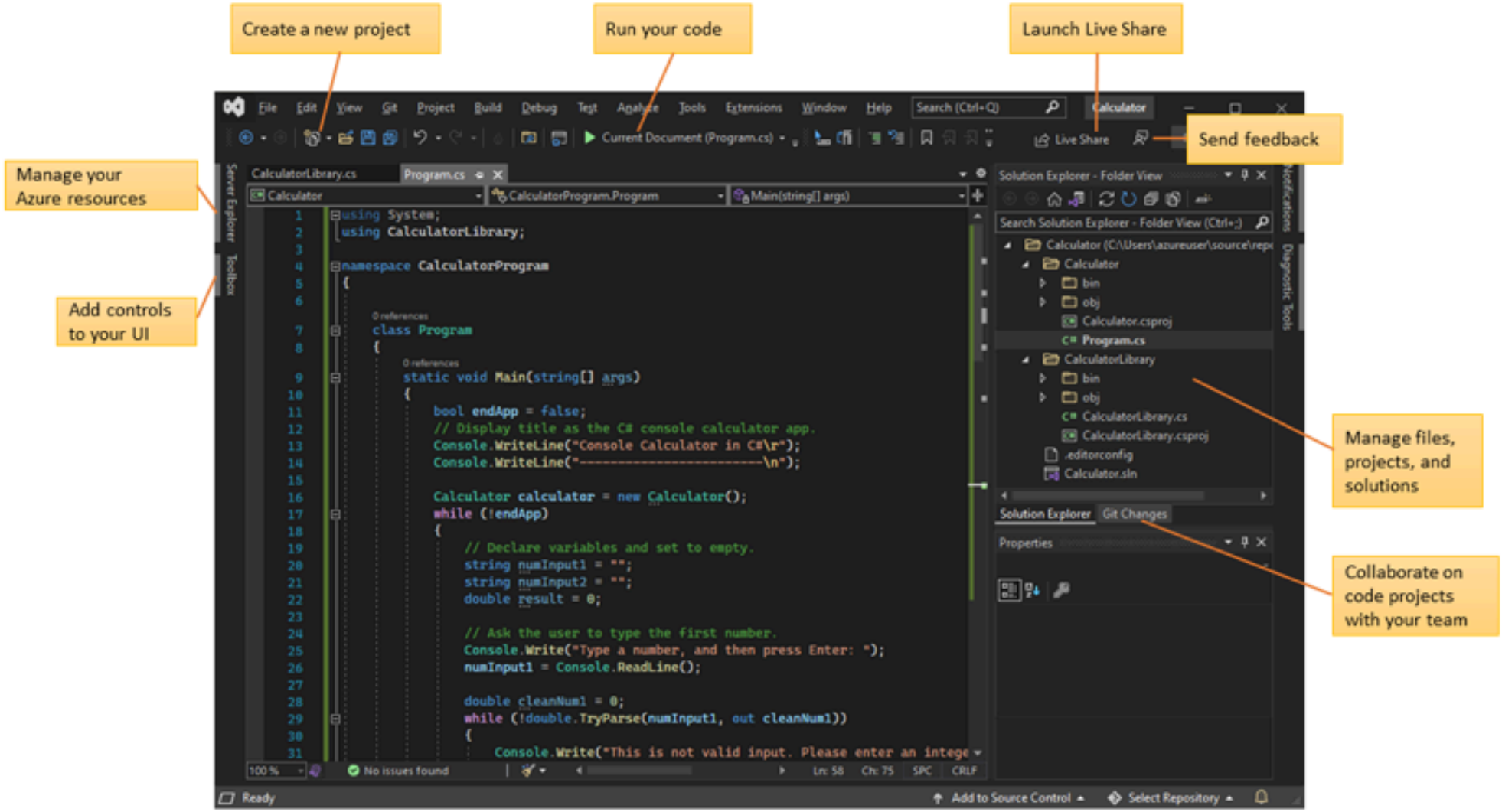


Microsoft-Docs: Introduction to the C# language and the .NET Framework published: 2015-07-20; accessed: 2020-07-13 https://docs.microsoft.com/en-us/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework

Visual Studio IDE

SWEN1 | BIF3

Visual Studio IDE



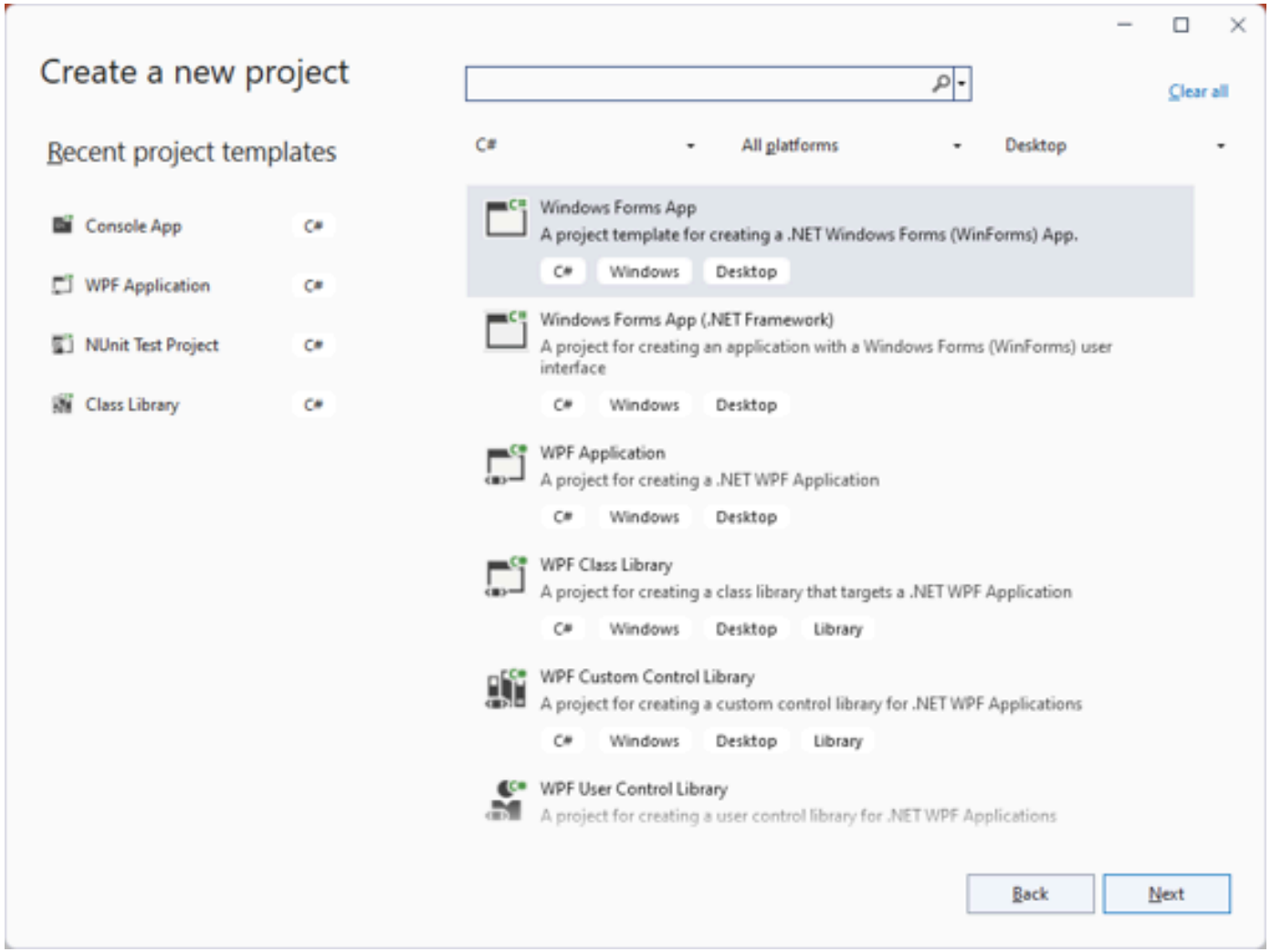
Create a Solution

Project/Solution-Templates for many kinds of applications, e.g.:

- Console App
- WPF Application
- ASP.Net

Further project-types:

- Class Library
- NUnit Test Project



Visual Studio - Editing & Refactoring

Action	Keys
Quick Actions / Refactoring Suggestions	`Alt` + `Enter` or `Ctrl` + `.`
Method Info	`Ctrl` + `K`, `Ctrl` + `I`
Comment / Uncomment	`Ctrl` + `K`, `Ctrl` + `C` / `Ctrl` + `K`, `Ctrl` + `U`
Delete Line (without copy)	`Ctrl` + `Shift` + `L`
Paste from keyboard buffer ring	`Ctrl` + `Shift` + `V`
Move Code Up / Down	`Alt` + `Up` / `Alt` + `Down`
Format Document / Selection	`Ctrl` + `K`, `Ctrl` + `D` / `Ctrl` + `K`, `Ctrl` + `F`
Surround with (.../if/try/foreach)	`Ctrl` + `K`, `Ctrl` + `S`
Rename	`Ctrl` + `R`, `Ctrl` + `R`
Encapsulate Field	`Ctrl` + `R`, `Ctrl` + `E`
Remove and Sort Usings	`Ctrl` + `R`, `Ctrl` + `G`
Extract Method	`Ctrl` + `R`, `Ctrl` + `M`

Visual Studio - Code Snippets

Type word in the editor, then hit `Tab`-key twice.

`Class` ...creates a class declaration
`ctor` ...a constructor for class
`prop` ...an auto-implemented property.
`propfull` ...a property with get and set
`propg` ...a read only property (private set)
`cw` ...Console.Writeline()
`do` ...do while loop
`for` ...for loop
`foreach` ...a foreach loop
`svm` ...static void main method declaration
`switch` ...a switch statement
`try` ...try block
`try` ...try block with finally clause
`while` ...while loop

Visual Studio - Popular Build Shortcuts

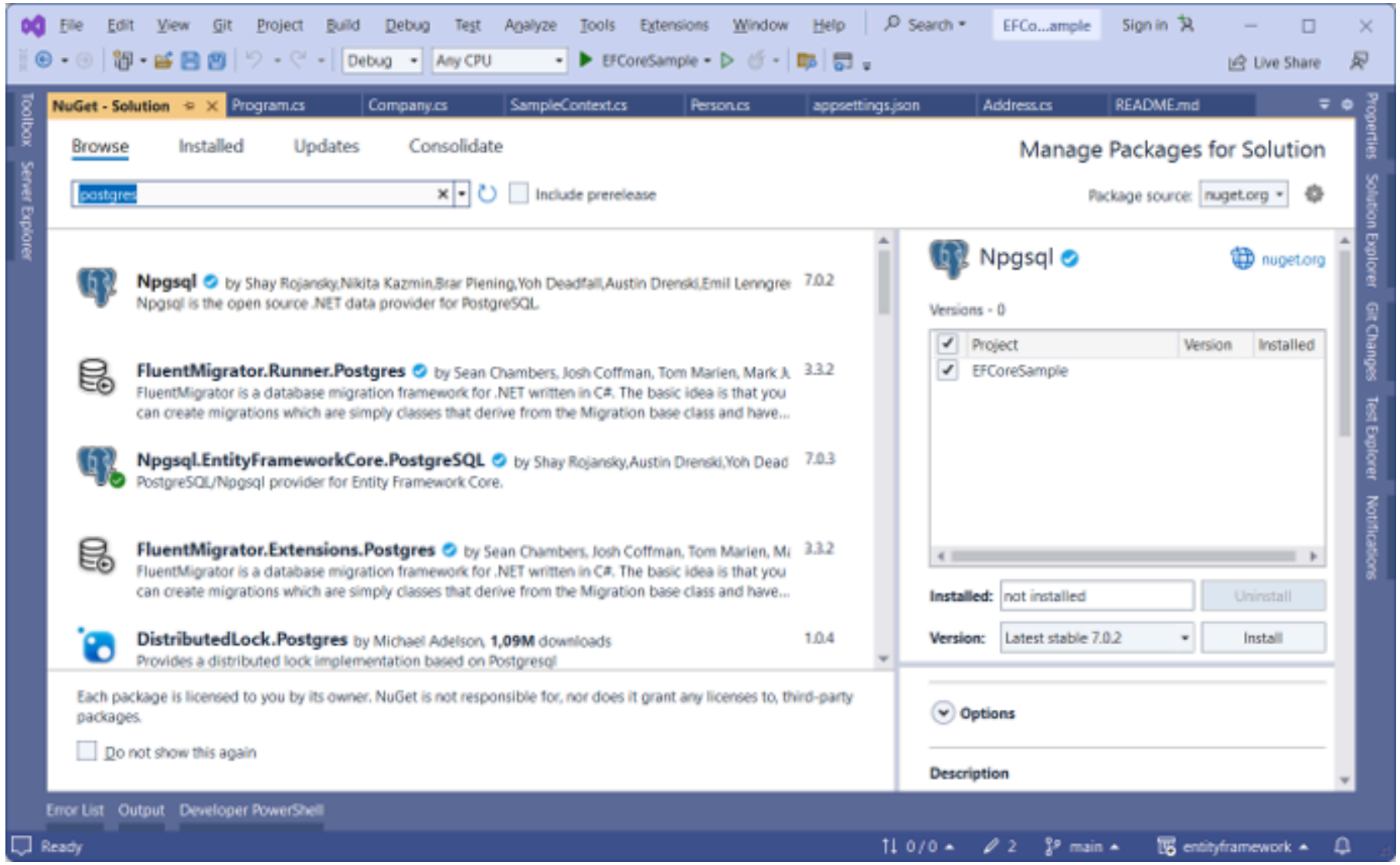
Action	Keys
Build solution	`Ctrl` + `Shift` + `B`
Cancel	`Ctrl` + `Break`
Compile	`Ctrl` + `F7`
Run code analysis on solution	`Alt` + `F11`

Visual Studio - Debugging and Testing

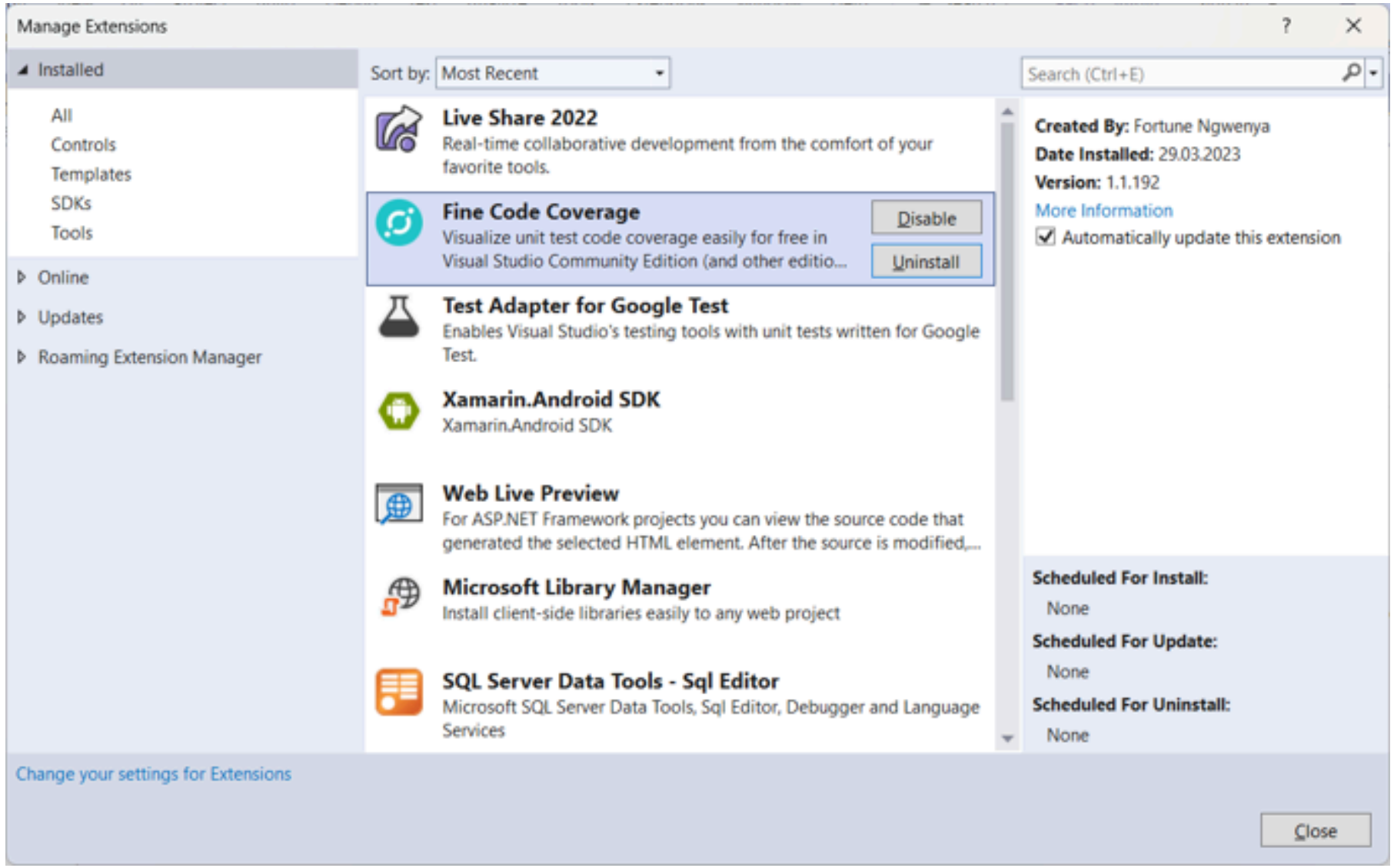
Action	Keys
Debug / Run / Stop	`F5` / `Ctrl` + `S` / `Alt` + `S`
Toggle Breakpoint	`F9`
Step Over	`F10`
Step Into	`F11`
Step Out	`Shift` + `F11`
Debug All Tests	`Ctrl` + `R`, `Ctrl` + `A`
Run All Tests	`Ctrl` + `R`, `A`

A list of further shortcuts: see https://visualstudio.microsoft.com/keyboard-shortcuts.pdf

Visual Studio - NuGet Packages



Visual Studio - Extensions



Visual Studio Code

SWEN1 | BIF3

VSCode - Pre-Requisites

- Install .Net SDK (8.0 or later) from https://dotnet.microsoft.com/download
- Verify installation in a terminal window: `dotnet --version`
- Install VS Code from https://code.visualstudio.com
- Start VSCode and install the following Extension: "C# Dev Kit" (by Microsoft)

VSCode - Create a new C# project

- In VS Code, choose `File > Open Folder…` and create/select a new folder like `HelloWorld`.
- Open the integrated terminal: `Ctrl+ (backtick)` or `View > Terminal`.
- Run the following command to create a console app: `dotnet new console -n HelloWorld`
  Remarks: To see what other program types can be created use the following command:
  `dotnet new list`
  Some of the further possibilities are:

Templatename	Shortname	Language	Tags
Class Library	classlib	[C#],F#,VB	Common/Library
Console-App	console	[C#],F#,VB	Common/Console
NUnit 3 Test Item	nunit-test	[C#],F#,VB	Test/NUnit
NUnit 3 Test Project	nunit	[C#],F#,VB	Test/NUnit
Solutionfile	sln,solution		Solution
„dotNet gitignore“-File	gitignore,.gitignore		Config

VSCode - Build and Run your application

- Open the "Program.cs" file, click on the `Play` button to build and run the application
- or Press `F5` or click `Run > Start Debugging`. VS Code will launch the debugger and stop at breakpoints if you set any.

You may also do all this using CLI commands in the Terminal:

- Change into the directory where the .csproj file is stored: `cd HelloWorld`
- To Build the project: `dotnet build`
- To Run the program: `dotnet run`

VSCode - Launchfile

/.vscode/settings.json:

```
{
    // Use IntelliSense to learn about possible attributes.
    // Hover to view descriptions of existing attributes.
    // For more information, visit: https://go.microsoft.com/fwlink/?linkid=830387
    "version": "0.2.0",
    "configurations": [
        {
            "name": "C#: HelloWorld Debug",
            "type": "dotnet",
            "request": "launch",
            "projectPath": "${workspaceFolder}/HelloWorld/HelloWorld.csproj"
        }

    ]
}
```

VSCode - Update NuGet Packages

To update all NuGet packages in your project, follow these steps:

- Open a terminal in your project directory (where the .csproj file is located).
- Run the following command to update all packages to their latest versions:
  `dotnet outdated`
- If you don't have the dotnet-outdated tool installed, install it first:
  `dotnet tool install --global dotnet-outdated-tool`
- Update the packages:
  `dotnet outdated --upgrade`
- Restore the updated packages:
  `dotnet restore`