Visual Programming

Dr. Muhammad Tufail

Assistant Professor

GPGC Nowshera

About this Course

- This course covers the core concepts of software development in C# (Dot net Framework 3.5).
- This will give students the idea about Graphical User Interface with different important controls like use of textbox for input data and GUI connectivity to database.
- How to save records in file
- How to create reports from historical/saved data.

Course Outline:

- **Dot Net Framework: C# Basic Introduction:**
- **Conditional Statements:**
- **Loops: , Array structure:,**
- **Object Oriented Concepts:**
- **Generics: Windows Form Basic: Textbox / Buttons: Checkbox / Radio Buttons: Combo box / Picture Box:**
- **Dialog Result: Menus and containers: Data Grid View:**
- **Other Controls:** List box, progress bar, timer etc.
- **N-Layer Architecture:** Introduction of N- Layer architecture
- **File Stream:** Read / write file, parsing a file
- **Reporting:** Use of crystal report

Books Recommended:

- Programming in Visual C# 2008 by Bradley–Millspaugh, McGraw Hill Publishers.
- Mastering Database Programming using C#, Sybex Publishers.
- C# How to Program: Deitel and Deitel.

Dot Net Framework

- The .Net framework is a software development platform developed by Microsoft.
- It is used to create applications, which would run on the Windows.
- The first version of the .Net framework was released in the year 2002.
- The version was called .Net framework 1.0. The .Net framework has come a long way since then, and the current version is 5.0

Dot Net Framework

- The .Net framework can be used to create both
  - - **Form-based** and **Web-based** applications.
- It supports running websites, services, desktop apps, and more on Windows.
- **.NET Core** is a cross-platform implementation for running websites, services, and console apps on Windows, Linux, and macOS.
- Xamarin/Mono is a .NET implementation for running apps on all the major mobile operating systems, including iOS and Android.

Dot Net Framework

- The framework also supports various programming languages such as Visual Basic and C#.
- Developers can choose and select the language to develop the required application.

Architecture of .NET Framework

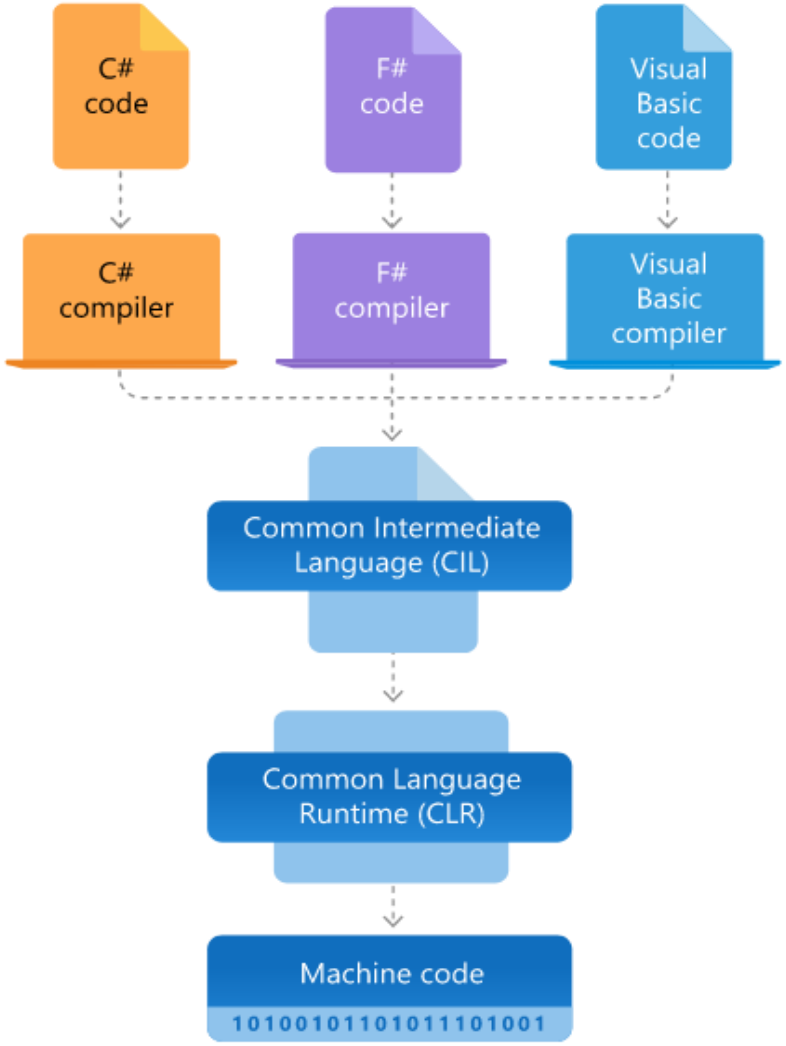
- The two major components of .NET Framework are
  - The Common Language Runtime and
  - The .NET Framework Class Library.

- The **Common Language Runtime (CLR)** is the execution engine that handles running applications. It provides services like thread management, garbage collection, type-safety, exception handling, etc.

- The **Class Library** provides a set of APIs (Application Programming Interface) and common functionality. The Class Library includes APIs for reading and writing files, connecting to databases, drawing, and more.

Architecture of .NET Framework

- .NET applications are written in the C#, F#, or Visual Basic programming language.
- The output of the compiler is not executable code. Instead, it is a file that contains a special type of pseudocode called Microsoft Intermediate Language (MSIL).
- Compiled code is stored in assemblies—files with a .dll or .exe file extension.
- Microsoft Intermediate Language is turned into executable code using a JIT compiler. “JIT” stands for “Just-In-Time.”
- The JIT compiler converts MSIL into native code on demand as each part of your program is needed



Managed vs. Unmanaged Code

- A program written in C#.net is called managed code.
- Managed code is executed under the control of the Common Language Runtime as just described.
- The opposite of managed code is unmanaged code.
- Unmanaged code does not execute under the Common Language Runtime.
- Thus, all Windows programs prior to the creation of the .NET Framework use unmanaged code.
- It is possible for managed code and unmanaged code to work together, so the fact that C# generates managed code does not restrict its ability to operate in conjunction with preexisting programs.

Thanks