

.Net Core.

- **What is .NET Core**

In short- ".NET Core is a software development framework which is used to create multiple types of application."

- Relationship with .NET Framework

✓ .NET Core and .Net Framework both are completely different frameworks.

✓ .NET Core is a completely new written framework which does not have any dependency or relationship with .Net framework.

✓ .NET Core is the newest and latest framework than the .Net framework.

- What can we do with .NET Core

✓ .NET Core framework is used to create multiple types of application

✓ There are multiple frameworks which are written on top of .NET Core to create different types of applications.

✓ Examples -

- ✓ Testing (Unit test)

- ✓ Console

- ✓ Mobile

- ✓ Web (MVC, API, Angular, React etc.)

- ✓ Windows (WPF, Win forms)

- ✓ Class library

- **Characteristics/Features of .NET Core**

Open Source - .Net core is open source which is developed and maintained by Microsoft and developers all around the world on GitHub. (Link - <https://github.com/dotnet/core>)

Cross Platform - .NET works on Windows, MacOS, Linux operating systems with the same behavior including x64, x86 and ARM.

Command line tool support - .NET Core fully supports command line tools which are useful in the complete cycle of development (Create new project, add package, build, run) etc.

Deployment - Application which are developed with .NET Core can be deployed on IIS, Linux, Docker etc.

It can be used with .NET Framework, Xamarin and Mono via .Net Standard.

Free of cost. .Net core is available as free of cost with MIT license. So you can use it for private and commercial purpose

- .Net core supports following programming language -
C#
F#
Visual Basic

- **CLI (Command line interface)**

✓ CLI stands for "Command Line Interface"

✓ CLI is an interface which is used to interact with the a program (or software)

✓ Nowadays each programming language has its own CLI For example - Angular, .NET, PHP, Package manager etc..

✓ As per Wikipedia -

"A command-line interface or command language interpreter (CLI), also known as command-line user interface, console user interface and character user interface (CUI), is a means of interacting with a computer program where the user (or client) issues commands to the program in the form of successive lines of text (command lines). A program which handles the interface is called a command language interpreter or shell (computing)."

- What is possible with .NET CLI?

✓ .NET CLI helps us to perform almost all the tasks which are required in order to work with a .NET Core application

✓ .NET CLI works with the command and these commands are applicable on all types of application of .NET Core.

✓ .NET CLI is a cross platform tool for developing .NET applications.

Examples -

✓ Create new project (Console, Web, Windows, Mobile app, Test etc.)

✓ Add/Update file, package, reference etc.

✓ Clean, build and run the application

✓ Test the application

✓ Publish the application

- Setting Up .NET CLI

✓ Install .NET Core SDK on your system.

✓ Choose any CLI available on your system.

Like Command prompt, PowerShell, Terminal etc....

✓ Now we need a driver for .NET Core

✓ And the driver in **dotnet**

✓ Type dotnet on CLI tool and press enter. Now if you don't see any error then you are good to go.

- NET CLI driver

✓ To run any command or application from .Net Core CLI we need a driver.

✓ **dotnet** is the driver for .Net Core CLI.

✓ It has two responsibilities - -

1. Running a app (Dependent on .Net Core framework)
2. Execute a command

- **Command, Argument, Option in CLI**

✓ A **command** also known as verb is used to perform an action. Example - dotnet build

✓ **Arguments** are passed to command invoked.

Example - - dotnet publish myApp.csproj

✓ **Option** - We can also pass some optional things to the command.

Example - dotnet new console --name myProjectName