

.NET Core versus .NET Framework

Source: [.NET Core vs .NET Framework - What's The Difference? - InterviewBit](#)

Basis	.NET Core	.NET Framework
Platform or Framework	When we talk about .NET Core it is defined as the platform on which frameworks like ASP.NET Core and the Universal Windows Platform rely and extend the .NET Core platform's functionalities.	.Net Framework is a full-fledged development framework. The framework provides all the basic requirements for the development of applications such as UI, DB connectivity, services, APIs, etc.
Open-Source	.NET Core is an open-source platform.	The .Net Framework includes certain open-source components.
Cross-Platform	It is based on the concept of "create once, run anywhere." Because it is cross-platform, it is compatible with a variety of operating systems, including Windows, Linux, and Mac OS.	.NET Framework is compatible with Windows OS (operating system) only
Application models	The Application Model of .Net Core includes ASP.NET and windows universal apps.	The Application Model of the .NET Framework includes WinForms, ASP.NET, and WPF.
Installation	.Net Core is cross-platform, hence it needs to be installed independently.	.NET Framework has a single packaged installation and runtime environment for windows.
Microservices support	.NET Core has support for microservices. , NET Core allows a mix of technologies that can be minimized for each microservice.	When we talk about the .NET Framework it does not allow for the construction and deployment of microservices in multiple languages.
REST services support	.NET Core has no support for WCF(Windows Communication Foundation) services. You would always need to create a REST API.	When it comes to WCF (Windows Communication Foundation) services, the .NET Framework is a fantastic choice. It also works with RESTful services.
Performance and Scalability	.NET core provides high scalability and performance compared to .NET Framework because of its architecture.	.NET Framework is less scalable and provides low performance compared to .NET Core.

Security	Feature such as Code Access Security is not present in .NET core so .NET Framework does have the edge in that case.	.NET Framework has this feature called code access security.
Focus on devices	.NET Core focuses to develop apps in a variety of domains like gaming, mobile, IoT, AI, etc.	.NET Framework is limited to Windows OS.
Compatibility	Mobile.NET Core is compatible with various operating systems-Windows, Linux, and Mac OS.	On the other .NET Framework is only compatible with Windows OS.
Mobile Development	Mobile apps are becoming more important for businesses. .NET Core has some support for mobile apps. It's compatible with Xamarin and other open-source platforms for mobile applications.	On the other hand, the .NET Framework does not support their development at all, and that is a problem.
CLI Tools	For all platforms,.NET Core provides a very lightweight CLI(Command Line Interface). There's always the option of switching to an IDE.	.NET Framework is too heavy for CLI. some developers prefer working on CLI rather than on IDE.
Deployment Model	When a new version of.NET Core is installed, it is updated on one computer at a time, resulting in new directories/folders being created in the existing program without affecting it. As a result,.NET Core provides a solid and adaptable deployment model.	IDE In the case of .NET Framework, when the updated version is released it is first deployed on the internet information service only.
Packaging and shipping	.NET Core is shipped as a collection of Nuggets packages.	All the libraries of the .NET Framework are packed and shipped together.