

What's New in .NET 6 APIs

Jason Bock

Personal Info

- <http://www.jasonbock.net>
- <https://www.twitter.com/jasonbock>
- <https://www.github.com/jasonbock>
- <https://www.youtube.com/c/JasonBock>
- jason.r.bock@outlook.com

Downloads

<https://github.com/JasonBock/WhatsNewInNET6APIs>

<https://github.com/JasonBock/Presentations>

Overview

- Focus
- .NET 6 Features
- Call To Action

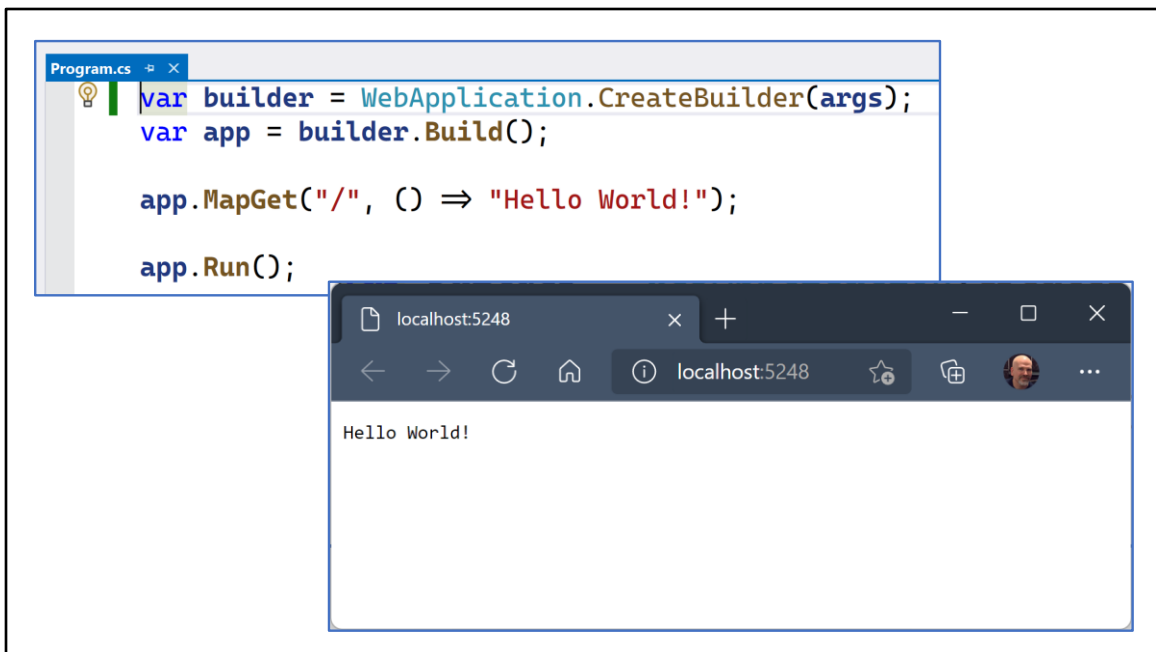
Remember...

<https://github.com/JasonBock/WhatsNewInNET6APIs>

<https://github.com/JasonBock/Presentations>

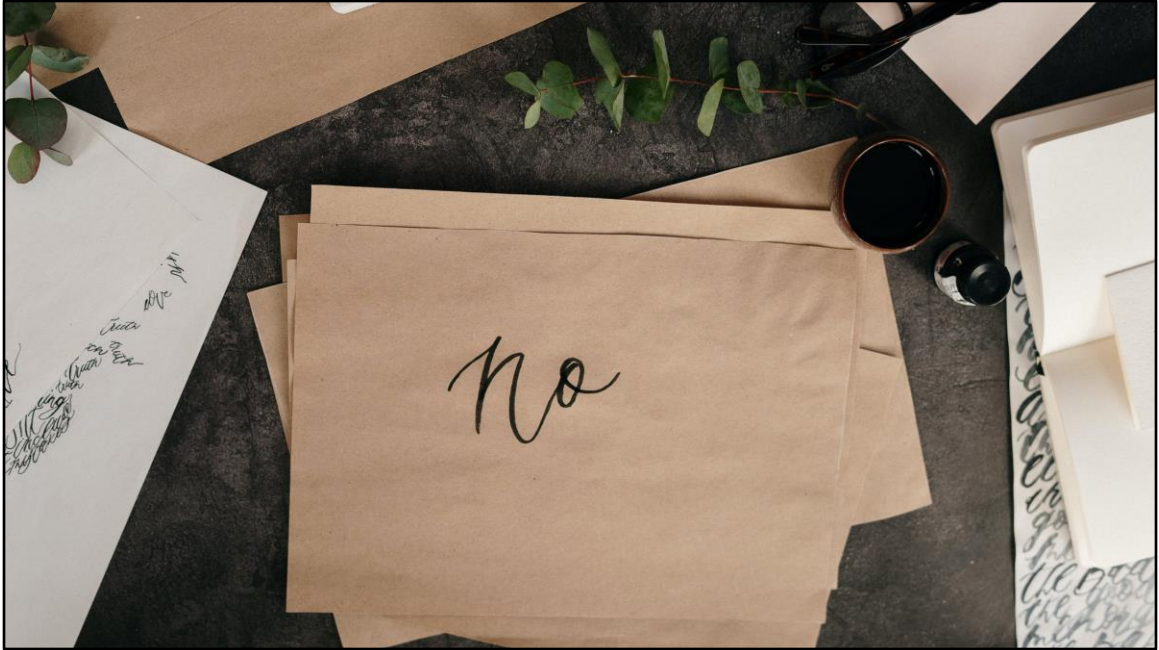
Focus

What's New in .NET 6 APIs



Let's level-set what we're going to be talking about. When developers hear "API", they may think that we're talking about "REST APIs". Or, if they use .NET, they may have heard of "Minimal APIs" in .NET 6. Either way, they might think I'm going to talk about one....or both.

<https://docs.microsoft.com/en-us/aspnet/core/fundamentals/minimal-apis>



That's not it. Those are definitely worth discussing, but that's not the focus of this talk.

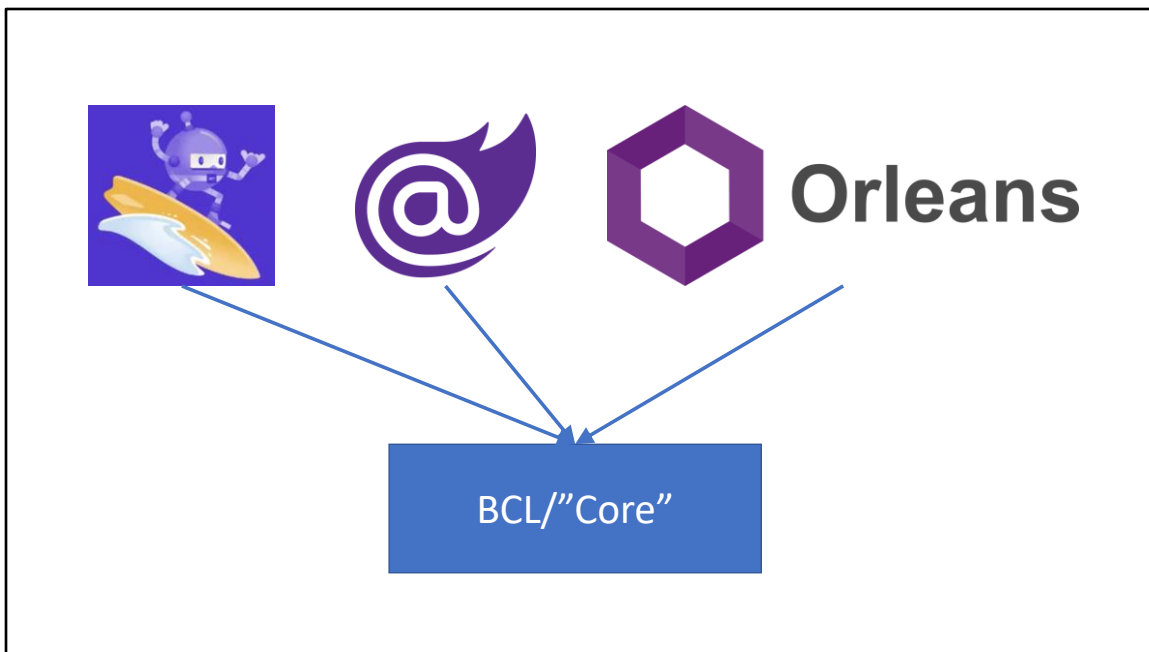
<https://www.pexels.com/photo/brown-paper-with-handwritten-text-3826681/>

An application programming interface (API) is a connection between computers or between computer programs. It is a type of software interface, offering a service to other pieces of software. A document or standard that describes how to build or use such a connection or interface is called an API specification. A computer system that meets this standard is said to implement or expose an API. The term API may refer either to the specification or to the implementation.

This is a somewhat formal definition of an API.

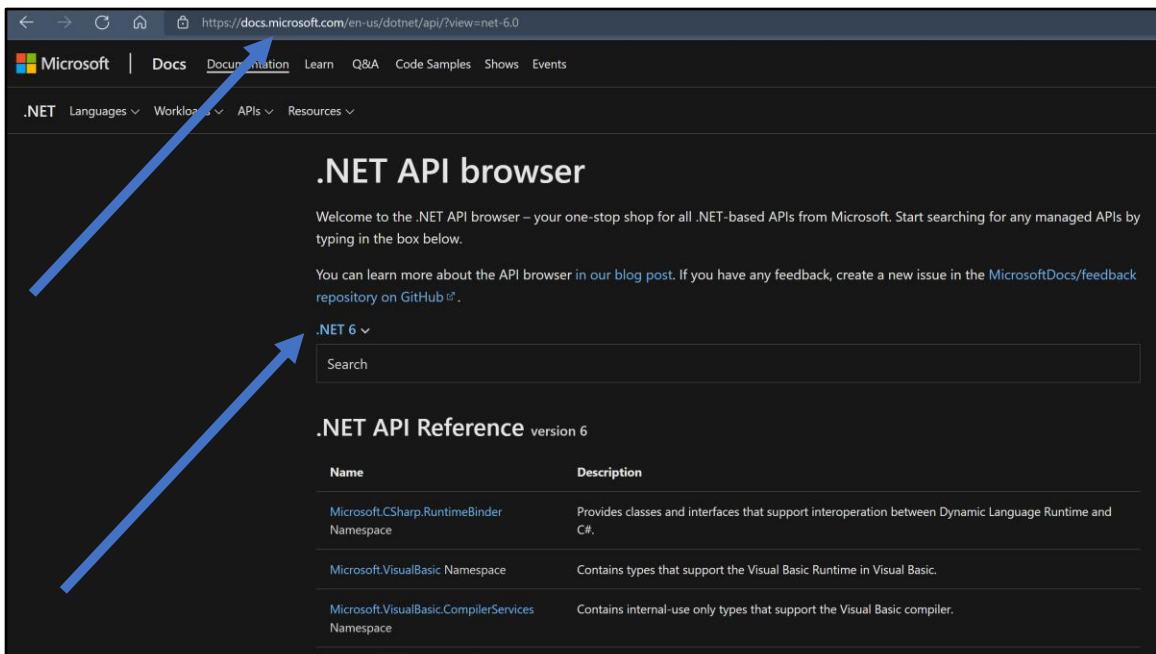
<https://en.wikipedia.org/wiki/API>

https://unsplash.com/photos/4Zaq5xY5M_c



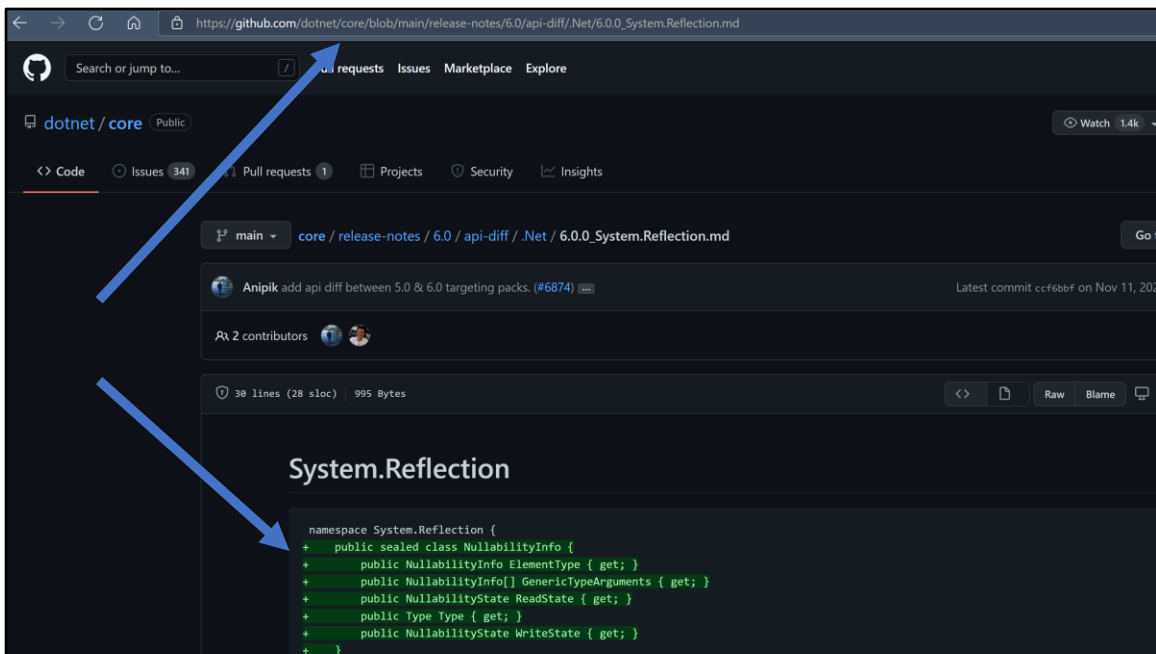
There are a bunch of frameworks, like MAUI, Blazor, Orleans, etc. that have their own APIs. What I'm referring to in this talk is the "Base Class Library". These are the core types.

What are the "core" types? The "Base Class Library" definition was always kind of nebulous. .NET Standard provided a "standard", and now .NET 5 and beyond provides that "standard".



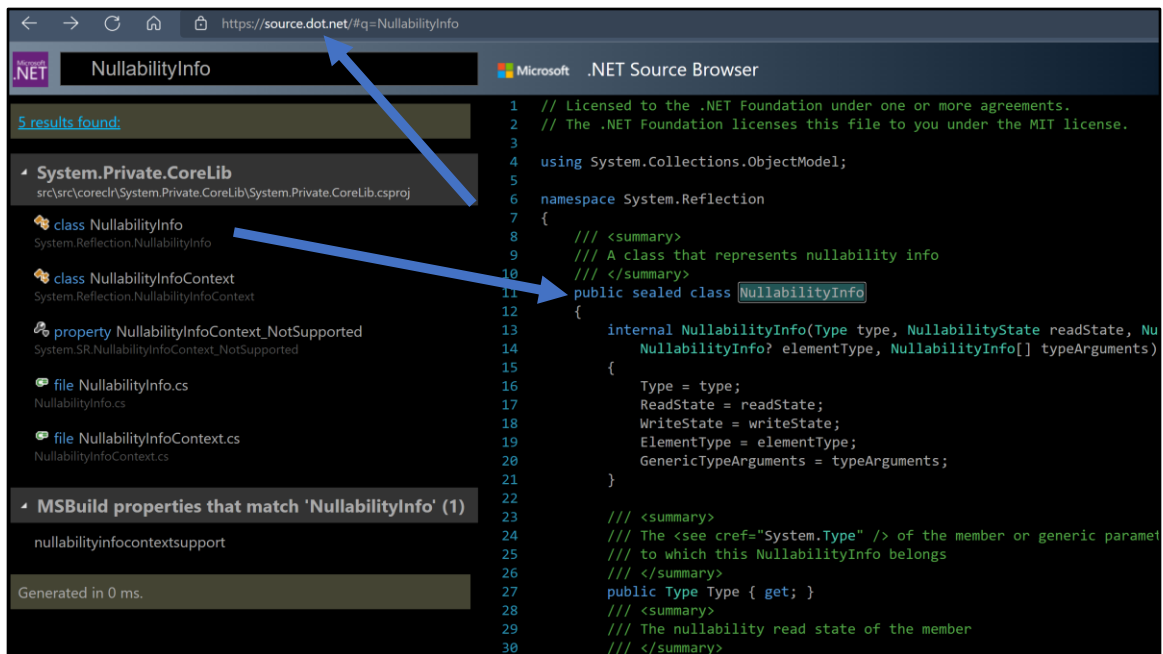
So what defines what's in each version? You can use the API browser, and filter it to ".NET 6"

https://docs.microsoft.com/en-us/dotnet/api/?view=net-6.0



Another place is to go to the dotnet/core repository, and look at the diff files produced by Microsoft. They're under the "release-notes" folder. It's not the greatest way, but you find everything that's changed.

<https://github.com/dotnet/core/blob/main/release-notes>
<https://docs.microsoft.com/en-us/dotnet/api/?view=net-6.0>



And if you're curious to see how it's defined, you can use the Source Index browser!

<https://source.dot.net/>



One thing to keep in mind before I do the demos. Some of these additions won't matter for you, where some of them will. The relevance of the feature is highly dependent on what you do in your code. So, don't be surprised if only a couple grab your attention. Then again, you never know where you may need some of these.

https://unsplash.com/photos/X4zx5Vc_LZU

Performance Improvements in .NET 6



Stephen

August 17th, 2021

Four years ago, around the time .NET Core 2.0 was being released, I wrote [Performance Improvements in .NET Core](#) to highlight the quantity and quality of performance improvements finding their way into .NET. With its very positive reception, I did so again a year later with [Performance Improvements in .NET Core 2.1](#), and an annual tradition was born. Then came [Performance Improvements in .NET Core 3.0](#), followed by [Performance Improvements in .NET 5](#). Which brings us to today.

The [dotnet/runtime](#) repository is the home of .NET's runtimes, runtime hosts, and core libraries. Since its main branch forked a year or so ago to be for .NET 6, there have been over 6500 merged PRs (pull

But even if none of these grab your attention, you should still consider upgrading to .NET 6. Since .NET Core, there has been a lot of engineering effort put into making the runtime and core types better from a perf perspective. The latest article from Taub illustrates the continuation of this journey. Simply upgrading to the new version has a big potential of improving the performance of your applications.

<https://devblogs.microsoft.com/dotnet/performance-improvements-in-net-6/>

Demo: .NET 6 Features

What's New in .NET 6 APIs

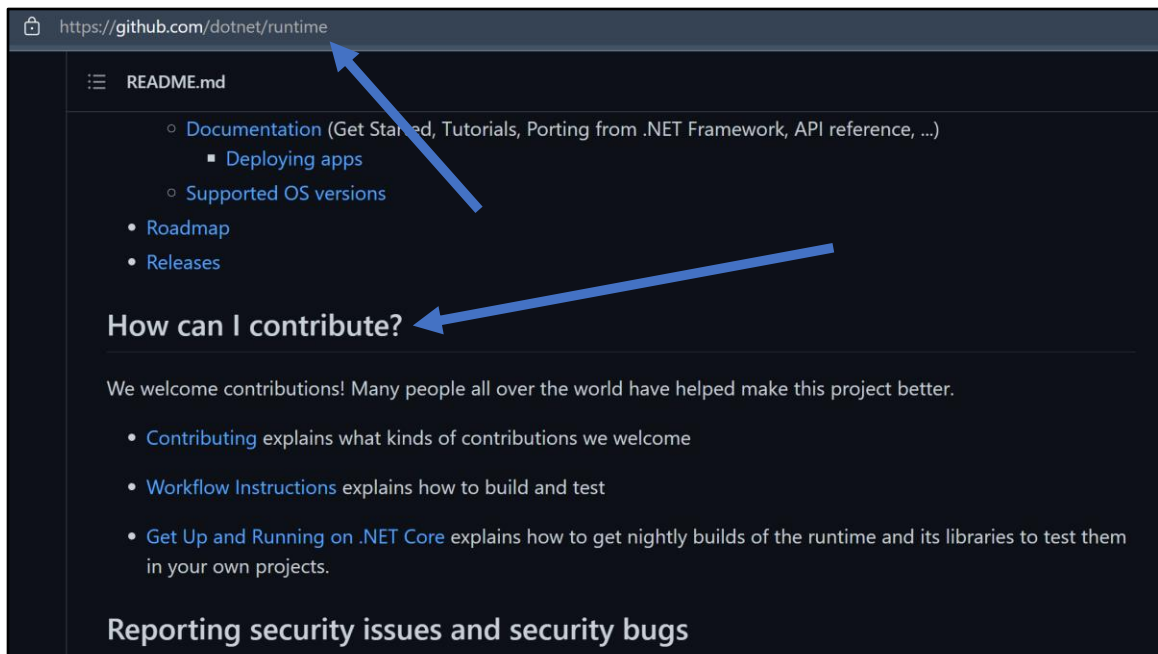
Call To Action

What's New in .NET 6 APIs



Dive in! .NET 6 has a bunch of new APIs that you may want to use, and the entire .NET API surface is large. You may find some new types or methods that you can use for your own applications.

<https://www.pexels.com/photo/man-wearing-blue-shorts-about-to-dive-on-body-of-water-1098365/>



And if you want to contribute, either by suggesting new features, or by doing the work, visit the .NET runtime repository.

<https://github.com/dotnet/runtime>

What's New in .NET 6 APIs

Jason Bock

Remember...

- <https://github.com/JasonBock/WhatsNewInNET6APIs>
- <https://github.com/JasonBock/Presentations>
- References in the notes on this slide

References

* Sources

- * [Official Start Page](<https://dotnet.microsoft.com/>)
- * [.NET Runtime](<https://github.com/dotnet/runtime>)
- * [Source Index](<https://source.dot.net/>)
- * [.NET API Browser](<https://docs.microsoft.com/en-us/dotnet/api/>)

- * [.NET Release

Notes](<https://github.com/dotnet/core/tree/main/release-notes>)

* References

* [.NET's Cryptographic One-Shots](<https://vcsjones.dev/one-shot-crypto/>)

* [.NET 6 Networking Improvements](<https://devblogs.microsoft.com/dotnet/dotnet-6-networking-improvements/>)

* [What's new in .NET 6](<https://david.gardiner.net.au/2021/11/dotnet-6.html>)

- * [Announcing .NET 6 — The Fastest .NET

Yet](<https://devblogs.microsoft.com/dotnet/announcing-net-6/>)

- * [Breaking changes in .NET 6](<https://docs.microsoft.com/en-us/dotnet/core/compatibility/6.0>)
- * [A big thread of new APIs in the .NET 6](<https://twitter.com/okyrylchuk/status/1455657409645617152>)
- * [Favorite improvements in .NET 6](<https://www.poppastring.com/blog/favorite-improvements-in-net-6>)
- * [New .NET 6 APIs driven by the developer community](<https://devblogs.microsoft.com/dotnet/new-dotnet-6-apis-driven-by-the-developer-community/>)
- * [Performance Improvements in .NET 6](<https://devblogs.microsoft.com/dotnet/performance-improvements-in-net-6/>)
- * [System.Reflection.NullabilityInfo mention](https://twitter.com/eric_sink/status/1429237223979995137)
- * [File IO improvements in .NET 6](<https://devblogs.microsoft.com/dotnet/file-io-improvements-in-dotnet-6/>)
- * [String Interpolation in C# 10 and .NET 6](<https://devblogs.microsoft.com/dotnet/string-interpolation-in-c-10-and-net-6/>)
- * [.NET 6 features](https://github.com/okyrylchuk/dotnet6_features)
- * [Generic Math](<https://devblogs.microsoft.com/dotnet/preview-features-in-net-6-generic-math/>)
- * .NET 6
 - * [LINQ Improvements](<https://www.infoq.com/news/2021/04/Net6-Linq/>)
 - * [Date and Time Structures](<https://www.infoq.com/news/2021/04/Net6-Date-Time/>)
 - *
 - [Threading](<https://www.infoq.com/news/2021/08/net6-Threading/>)
 - * [Async Improvements](<https://www.infoq.com/news/2021/04/Net6-Async/>)
 - * [Collections Improvements](<https://www.infoq.com/news/2021/06/Net6-Collections/>)