**What is ASP.NET?**

**ASP.NET is an open source web framework, created by Microsoft, for building modern web apps and services with .NET.**

**ASP.NET is cross platform and runs on Windows, Linux, macOS, and Docker.**

**The .NET platform**

.NET is a developer platform made up of tools, programming languages, and libraries for building many different types of applications.

The base platform provides components that apply to all different types of apps. Additional frameworks, such as ASP.NET, extend .NET with components for building specific types of apps.

Here are some things included in the .NET platform:

* **The C#, F#, and Visual Basic programming languages**
* **Base libraries** for working with strings, dates, files/IO, and more
* **Editors and tools** for Windows, Linux, macOS, and Docker

[**Learn more about the .NET platform**](https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet)

**ASP.NET extends .NET**

ASP.NET extends the .NET platform with tools and libraries specifically for building web apps.

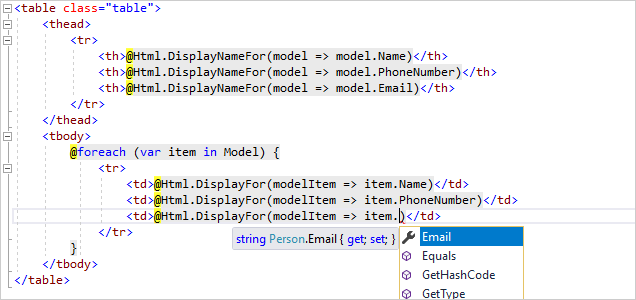
These are some things that ASP.NET adds to the .NET platform:

* **Base framework for processing web requests in C# or F#**
* **Web-page templating syntax**, known as Razor, for building dynamic web pages using C#
* **Libraries for common web patterns**, such as Model View Controller (MVC)
* **Authentication system** that includes libraries, a database, and template pages for handling logins, including multi-factor authentication and external authentication with Google, Twitter, and more.
* **Editor extensions** to provide syntax highlighting, code completion, and other functionality specifically for developing web pages

**Back-end code**

When using ASP.NET your back-end code, such as business logic and data access, is written using C#, F#, or Visual Basic.

Because ASP.NET extends .NET, you can use the large ecosystem of packages and libraries available to all .NET developers. You can also author your own libraries that are shared between any applications written on the .NET platform.



**Dynamic pages using C#, HTML, CSS, and JavaScript**

Razor provides a syntax for creating dynamic web pages using HTML and C#. Your C# code is evaluated on the server and the resulting HTML content is sent to the user.

Code that executes client-side is written in JavaScript. ASP.NET integrates with JavaScript frameworks and includes pre-configured templates for single page app (SPA) frameworks like React and Angular.

**100,000+**OSS contributions**3,700+**OSS company contributors

**Open-source**

Like the rest of .NET, ASP.NET is open source on GitHub. .NET has over 100,000 contributions and 3,700 companies have already contributed.

[**Learn more about .NET and open-source**](https://dotnet.microsoft.com/en-us/platform/open-source)

**Windows, Linux, macOS, and Docker**

ASP.NET apps can be developed and run on Windows, Linux, macOS, and Docker.

The Visual Studio family of products has tools for building .NET apps on any operating system. There are also command-line tools and extensions for many popular editors.

[**Learn about tools for .NET**](https://dotnet.microsoft.com/en-us/platform/tools)

**Pages, APIs, real-time, and microservices**

ASP.NET allows you to build many types of web applications, including [web pages](https://dotnet.microsoft.com/en-us/apps/aspnet/web-apps), [REST APIs](https://dotnet.microsoft.com/en-us/apps/aspnet/apis), [microservices](https://dotnet.microsoft.com/en-us/apps/aspnet/microservices), and [hubs that push real-time content to connected clients](https://dotnet.microsoft.com/en-us/apps/aspnet/signalr).

**What is ASP.NET Core?**

If you use ASP.NET, you'll soon come across the term ASP.NET Core.

ASP.NET Core is the open-source and cross-platform version of ASP.NET. You should use ASP.NET Core for all new applications. The tutorials on this site all use ASP.NET Core.

The Windows-only versions of ASP.NET, that existed before ASP.NET Core, is typically just referred to as ASP.NET. The majority of innovation occurs in ASP.NET Core, but other versions continue to receive minor updates and bug-fixes.

[**Learn more about ASP.NET Core**](https://dotnet.microsoft.com/en-us/learn/aspnet/what-is-aspnet-core)

***F# is a universal programming language for writing succinct, robust and performant code. F# allows you to write uncluttered, self-documenting code, where your focus remains on your problem domain, rather than the details of programming.Oct 13, 2022***

### **[What is F# | Microsoft Learn](https://learn.microsoft.com/en-us/dotnet/fsharp/what-is-fsharp" \l ":~:text=F%23%20is%20a%20universal%20programming,than%20the%20details%20of%20programming.)**

**Where is F# used programming?**

Among others, F# is used for **quantitative finance programming, energy trading and portfolio optimization, machine learning, business intelligence and social gaming on Facebook**. In the 2010s, F# has been positioned as an optimized alternative to C#.

What does asynchronous mean?

What does asynchronous mean? More specifically, asynchronous describes the relationship between two or more events/objects that do interact within the same system but do not occur at predetermined intervals and do not necessarily rely on each other's existence to function.

a·syn·chro·nous

/āˈsiNGkrənəs/

[Learn to pronounce](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=how+to+pronounce+asynchronous&stick=H4sIAAAAAAAAAOMIfcRoxy3w8sc9YSnTSWtOXmPU5-INKMrPK81LzkwsyczPE5LgYglJLcoVEpDi4-JJLK7MS84AyueXFluxKDGl5vEsYpXNyC9XKMlXKACL5yWnKiArAwDK--GRYwAAAA&pron_lang=en&pron_country=us&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQ3eEDegQIChAK)

See definitions in:

All

Technology

Telecommunications

Astronomy

*adjective*

1. 1.

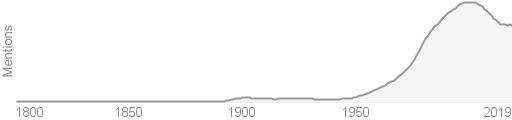
(of two or more objects or events) not existing or happening at the same time.

1. 2.

**COMPUTING•TELECOMMUNICATIONS**

of or requiring a form of computer control timing protocol in which a specific operation begins upon receipt of an indication (signal) that the preceding operation has been completed.



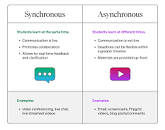
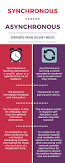
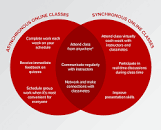
[](http://books.google.com/ngrams/graph?year_start=1800&year_end=2019&corpus=26&smoothing=7&case_insensitive=on&content=asynchronous)

### **[Images for asynchronous](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&source=univ&fir=Kd07mH71pwnKuM%252CtiyBEVAwis_swM%252C_%253B9XZjXg2kPASmiM%252ChBebYUCCO3xliM%252C_%253B-4zh70p2BmRXcM%252CYoVjSe0tSF6lvM%252C_%253B4-ZDwE2ykop-HM%252CPuwDGTXOiNWB7M%252C_%253BiT0NTejn-Q-UmM%252C1DK_-70ql8NAIM%252C_%253BmbN3Q7fOVeQN2M%252CJLZn0aHV_7kcCM%252C_%253BV_uK1jE6rThALM%252CJ4-nE-rQhUDuDM%252C_%253BfwzOGC5y9xRpcM%252C4v3_kjqhXq94YM%252C_%253BpQo4rMQjWdoBXM%252Cji0QbIfCs-WrZM%252C_%253BLPjRPUSJWQPhTM%252C4NLOEgQYfS5W9M%252C_&usg=AI4_-kROOcqBb4vRtpx5-pZPlZt8nbl7xg&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQjJkEegQIJRAC)**

### **Guided Search Filters**

Filter by feature

* [C:\Users\client\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\13DA158.tmp](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:aws+lambda:VH-1hBLyfjA%3D&usg=AI4_-kQwyNAJousOqanygRhl4BHucS3bFA&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAB6BAglEBE)
* [aws lambda](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:aws+lambda:VH-1hBLyfjA%3D&usg=AI4_-kQwyNAJousOqanygRhl4BHucS3bFA&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAB6BAglEBE)
* [lambda function](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:lambda+function:4XO0mCl7LdM%3D&usg=AI4_-kQK743T3bapMjUuJL_IS4UgRbXX4w&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAF6BAglEBU)
* [asynchronous development](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:asynchronous+development:97jkoIKVWkc%3D&usg=AI4_-kTNRvII_PSYADaq9M2GS42Cx0mnYw&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAJ6BAglEBk)
* [asynchronous programming](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:asynchronous+programming:FPdyLhjODH8%3D&usg=AI4_-kS7SPHVcms-AOuscOdhsp6Zlko6sQ&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAN6BAglEB0)
* [async programming](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:async+programming&usg=AI4_-kRm4Zx9yX4bzOMvQs3XhsTEQ5XfOg&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAR6BAglECE)
* [gifted students](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:gifted+students&usg=AI4_-kRKOlBNkIIeZq87HyNLJ0lFR61uag&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAV6BAglECQ)
* [synchronous communication](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:synchronous+communication:2lkjxKw4CqU%3D&usg=AI4_-kTlElOh-9fbXeKJrEfWYFFquTPFgA&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAZ6BAglECc)
* [giftedness](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:giftedness&usg=AI4_-kTBn4vgJnFC4edVmh48ewiUVHO4cw&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAd6BAglECs)
* [input output](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:input+output:2YAKpYTVpVs%3D&usg=AI4_-kSKVZkQFslKyiESyfLQ6MoK0iidDA&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAh6BAglEC4)
* [asynchronous invocation](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:asynchronous+invocation&usg=AI4_-kQ__UIJmARhnd1yMu9h8orqlPe8mA&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAl6BAglEDI)
* [async communication](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:async+communication:o4x9xIV_Bok%3D&usg=AI4_-kR9MuWi11N_LF3k2QYpiqV7TPACsw&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAp6BAglEDU)
* [e learning](https://www.google.com/search?rlz=1C1GCEU_enUS899US899&sxsrf=AJOqlzVDy4gTriTZ6zk83EHJR5BLVLSj0Q:1674076592158&q=asynchronous&tbm=isch&chips=q:asynchronous,online_chips:e+learning:4gZT5wKS6NE%3D&usg=AI4_-kTYFe6b0SbVMJy0C1Q9mB6toSolHw&sa=X&ved=2ahUKEwjXr-ikhdL8AhV7mWoFHeyTAZUQgIoDKAt6BAglEDk)





# Asynchronous

The term **asynchronous** refers to two or more objects or events **not** existing or happening at the same time (or multiple related things happening without waiting for the previous one to complete). In computing, the word "asynchronous" is used in two major contexts.

Networking and communications

Asynchronous communication is a method of exchanging messages between two or more parties in which each party receives and processes messages whenever it's convenient or possible to do so, rather than doing so immediately upon receipt. Additionally, messages may be sent without waiting for acknowledgement, with the understanding that if a problem occurs, the recipient will request corrections or otherwise handle the situation.

For humans, email is an asynchronous communication method; the sender sends an email and the recipient will read and reply to the message when it's convenient to do so, rather than doing so at once. And both sides can continue to send and receive messages whenever they wish, instead of having to schedule them around each other.

When software communicates asynchronously, a program may make a request for information from another piece of software (such as a server), and continue to do other things while waiting for a reply. For example, the [AJAX](https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX) (Asynchronous JavaScript and [XML](https://developer.mozilla.org/en-US/docs/Glossary/XML)) programming technique—now usually "Ajax", even though [JSON](https://developer.mozilla.org/en-US/docs/Glossary/JSON) is usually used rather than XML in modern applications—is a mechanism that requests relatively small amounts of data from the server using [HTTP](https://developer.mozilla.org/en-US/docs/Glossary/HTTP), with the result being returned when available rather than immediately.

Software design

Asynchronous software design expands upon the concept by building code that allows a program to ask that a task be performed alongside the original task (or tasks), without stopping to wait for the task to complete. When the secondary task is completed, the original task is notified using an agreed-upon mechanism so that it knows the work is done, and that the result, if any, is available.

There are a number of programming techniques for implementing asynchronous software. See the article [Asynchronous JavaScript](https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Asynchronous) for an introduction to them.

## [See also](https://developer.mozilla.org/en-US/docs/Glossary/Asynchronous#see_also)

* [Fetching data from the server](https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Client-side_web_APIs/Fetching_data) (Learning Area)
* [Synchronous](https://developer.mozilla.org/en-US/docs/Glossary/Synchronous)

### Found a content problem with this page?

* Edit the page [on Github](https://github.com/mdn/content/edit/main/files/en-us/glossary/asynchronous/index.md).
* Report the [content issue](https://github.com/mdn/content/issues/new?template=page-report.yml&mdn-url=https%3A%2F%2Fdeveloper.mozilla.org%2Fen-US%2Fdocs%2FGlossary%2FAsynchronous&metadata=%3C%21--+Do+not+make+changes+below+this+line+--%3E%0A%3Cdetails%3E%0A%3Csummary%3EPage+report+details%3C%2Fsummary%3E%0A%0A*+Folder%3A+%60en-us%2Fglossary%2Fasynchronous%60%0A*+MDN+URL%3A+https%3A%2F%2Fdeveloper.mozilla.org%2Fen-US%2Fdocs%2FGlossary%2FAsynchronous%0A*+GitHub+URL%3A+https%3A%2F%2Fgithub.com%2Fmdn%2Fcontent%2Fblob%2Fmain%2Ffiles%2Fen-us%2Fglossary%2Fasynchronous%2Findex.md%0A*+Last+commit%3A+https%3A%2F%2Fgithub.com%2Fmdn%2Fcontent%2Fcommit%2F9d53a300e0baae28dcad8c60080fa0152dd2b023%0A*+Document+last+modified%3A+2023-01-13T15%3A50%3A18.000Z%0A%0A%3C%2Fdetails%3E).
* View the source [on GitHub](https://github.com/mdn/content/blob/main/files/en-us/glossary/asynchronous/index.md?plain=1).

Want to get more involved? Learn [how to contribute](https://github.com/mdn/content/blob/main/CONTRIBUTING.md).

This page was last modified on Jan 13, 2023 by [MDN contributors](https://developer.mozilla.org/en-US/docs/Glossary/Asynchronous/contributors.txt).