

History of C# Language

C# is the most popular language in world. C# was developed by Microsoft within its .NET framework initiative and later approved as a standard by ECMA (ECMA-334) C# programming language is a general-purpose, OOPS based programming language. Its development team is led by Anders Hejlsberg.

When you go back and look, C# version 1.0 looked a lot like Java, it sought to be a "simple, modern, general-purpose object-oriented language." At the time, looking like Java meant it achieved those early design goals.

But if you look back on C# 1.0 now, you'd find yourself a little dizzy. It lacked the built-in async capabilities and some of the slick functionality around generics you take for granted. As a matter of fact, it lacked generics altogether. And LINQ? Not available yet. Those additions would take some years to come out.

C# version 1.0 looked stripped of features, compared to today. You'd find yourself writing some verbose code. But yet, you have to start somewhere. C# version 1.0 was a viable alternative to Java on the Windows platform.

The major features of each version:

C# version 1.0 (Released 2002 with .NET framework 1.0 and Visual Studio 2002)

- Classes
- Structs
- Interfaces
- Events
- Properties
- Delegates
- Expressions
- Statements
- Attributes

C# version 2.0 (Released 2005 with .NET framework 2.0 and Visual Studio 2005)

- Generics
- Partial types
- Anonymous Methods
- Iterators
- Nullable types
- Getter/setter separate accessibility

- Method group conversions (delegates)
- Static classes
- Delegate inference

C# version 3.0 (Released 2008 with .NET framework 2.0 and Visual Studio 2008)

- Implicitly typed local variables
- Object and collection initializers
- Auto-Implemented properties
- Anonymous types
- Extension Methods
- Query expressions
- Lambda expression
- Expression trees
- Partial methods

C# version 4.0 (Released 2010 with .NET framework 4.0 and Visual Studio 2010)

- Dynamic binding
- Named and optional arguments

C# version 5.0 (Released 2012 with .NET framework 4.5 and Visual Studio 2012)

- Asynchronous methods
- Caller info attributes

C# version 6.0 (Released 2015 with .NET framework 4.6 and Visual Studio 2015)

- Compiler-as-a-service
- Import of static type members into namespace
- Exception filters
- Await in catch/finally blocks
- Auto property initializers
- Default values for getter-only properties
- Expression-bodied members

- Null propagator
- String interpolation
- nameof operator
- Dictionary initializer

C# version 7.0 (Released 2017 with .NET framework 4.6.2 and Visual Studio 2017)

- Out variables
- Pattern matching
- Tuples
- Deconstruction
- Local functions
- Digit separators
- Binary literals
- Ref returns and locals
- Generalized async return types
- Expression bodied constructors and finalizers
- Expression bodied getters and setters
- Throw can also be used as expression

Minor versions

C# version 7.1 (August 2017)

- Async main

- Default literal expressions
- Inferred tuple element names

C# version 7.2 (November 2017)

- Reference semantics with value types
- Non-trailing named arguments
- Leading underscores in numeric literals
- `private protected` access modifier

C# version 7.3 (May 2018)

- Accessing fixed fields without pinning
- Reassigning `ref` local variables
- Using initializers on `stackalloc` arrays
- Using `fixed` statements with any type that supports a pattern
- Using additional generic constraints