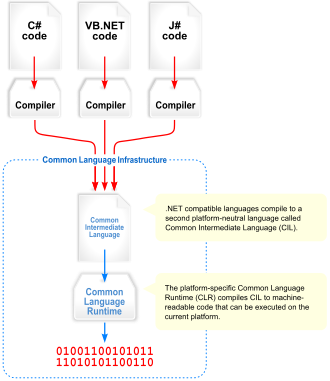
.Net

.NET (denumit anterior .NET Core) este o platformă software gratuită și open-source pentru sistemele de operare Windows, Linux și macOS. Este un succesor al .NET Framework oferind mai multă varietate în ceea ce privește compatibilitate cu mai multe platforme. Proiectul este dezvoltat în principal de către angajații Microsoft prin intermediul .NET Foundation și este lansat sub licența MIT.

.NET are patru mari utilizări: aplicații web ASP.NET Core; aplicații cu linie de comandă; biblioteci; și aplicațiile Universal Windows Platform. Înainte de .NET Core 3.0, nu implementa Windows Forms sau Windows Presentation Foundation (WPF), care redau interfață grafică standard pentru software-ul desktop pe Windows. După apariția .NET Core 3, platforma acceptă tehnologii desktop Windows Forms, WPF și Universal Windows Platform (UWP).

. NET acceptă utilizarea pachetelor NuGet. Spre deosebire de .NET Framework, care este deservit folosind Windows Update, .NET se bazează pe managerul său de pachete pentru a primi actualizări. Începând cu decembrie 2020, totuși, actualizările .NET au început să fie livrate și prin Windows Update.

Cele două componente principale ale .NET sunt CoreCLR și CoreFX.

.NET acceptă pe deplin scrierea în limbajele C# și F# (și C++/CLI începând cu versiunea 3.1; activ însă numai pe Windows) și acceptă Visual Basic .NET (pentru versiunea 15.5 în .NET Core 5 și unele versiuni vechi acceptate în vechiul .NET Core).

Infrastructura utilizată de .NET ([.NET - Wikipedia](https://en.wikipedia.org/wiki/.NET))

C# (C sharp)

C# este un limbaj pentru programare profesională. C# (pronunțat C sharp) este un limbaj de programare conceput pentru a construi o gamă largă de aplicații de întreprindere care rulează pe .NET Framework sau .Net (în trecut .Net Core). Scopul creatorilor C# este de a oferi un limbaj simplu, sigur, modern, orientat pe obiecte, de înaltă performanță, robust și durabil pentru dezvoltarea .NET. De asemenea, permite dezvoltatorilor să construiască soluții pentru cea mai largă gamă de clienți, inclusiv aplicații Web, aplicații bazate pe Microsoft Windows Forms.

C# a fost proiectat de Anders Hejlsberg de la Microsoft în 2000 și a fost ulterior aprobat ca standard internațional de Ecma (ECMA-334) în 2002 și ISO (ISO/IEC 23270) în 2003. Microsoft a introdus C# împreună cu .NET Framework și Visual Studio fără a dezvălui modul lor de lucru. La acea vreme, Microsoft nu avea produse open-source. Patru ani mai târziu, în 2004, a început un proiect gratuit și open-source numit Mono, care oferă un compilator multiplatformă și un mediu de rulare pentru limbajul de programare C#. Un deceniu mai târziu, Microsoft a lansat Visual Studio Code (editor de cod), Roslyn (compilator) și platforma unificată .NET (cadru software), toate care acceptă C# și sunt gratuite, open-source și multiplatformă. Mono s-a alăturat de asemenea celor de la Microsoft, dar nu a fost fuzionat în .NET.

Dezvoltatorii Visual C# își pot folosi abilitățile și cunoștințele existente în C, C++, Java pentru a avea succes în mediul de dezvoltare Microsoft .NET. Atât de multe dezvoltări C, Java și C++ se vor muta în C# pentru a profita de caracteristicile .NET. În cooperare cu .NET CLR (Common Language Runtime), oferă un limbaj de utilizare pentru software-ul orientat pe componente, fără a forța programatorii să-și abandoneze cunoștințele existente în codul C, C++ sau COM.

Sintaxa de bază a limbajului C# este similară cu cea a altor limbaje în stil C, cum ar fi C, C++ și Java, în special:

* Punctele și virgulă sunt folosite pentru a indica sfârșitul unei declarații.
* Parantezele sunt folosite pentru a grupa declarațiile. Instrucțiunile sunt grupate în mod obișnuit în metode (funcții), metodele în clase și clasele în spații de nume.
* Variabilele sunt atribuite folosind un semn egal, dar comparate folosind două semne egal consecutive.
* Parantezele pătrate sunt folosite cu matrice, atât pentru a le declara, cât și pentru a obține o valoare la un index dat într-unul dintre ele.

Microsoft a folosit pentru prima dată numele C# în 1988 pentru o variantă a limbajului C concepută pentru compilare incrementală.[36] Acel proiect nu a fost finalizat, dar numele rămâne.

Denumirea „C sharp” a fost inspirată de notația muzicală prin care un simbol ascuțit indică faptul că nota scrisă ar trebui să fie făcută cu un semiton mai mare în înălțime. Acesta este similar cu numele limbajului C++, unde „++” indică faptul că o variabilă ar trebui să fie incrementată cu 1 după ce a fost evaluată. Simbolul diez seamănă, de asemenea, cu o legătură de patru simboluri „+” (într-o grilă de două câte două), implicând în continuare că limbajul este un increment de C++.

Din cauza limitărilor tehnice ale afișajului (fonturi standard, browsere etc.) și a faptului că simbolul specific muzical nu este prezent pe majoritatea formelor de tastatură, semnul numeric diez (#) a fost ales pentru a aproxima simbolul original din numele scris al limbajului de programare. Această convenție este reflectată în specificația limbajului ECMA-334 C#.

Sufixul „sharp” a fost folosit de o serie de alte limbaje .NET care sunt variante ale limbiajelor existente, inclusiv J# (un limbaj .NET conceput și de Microsoft, care este derivat din Java 1.1), A# (de la Ada) și limbaj de programare funcțional F#. Implementarea inițială a Eiffel pentru .NET a fost numită Eiffel#, un nume retras deoarece limbajul Eiffel este acum suportat. Sufixul a fost folosit și pentru biblioteci, cum ar fi Gtk# (un wrapper .NET pentru GTK și alte biblioteci GNOME) și Cocoa# (un wrapper pentru Cocoa).

Din 2021, cea mai recentă versiune a limbajului este C# 10.0, care a fost lansată în 2021 în .NET 6.0.

Angular

Angular este o platformă de dezvoltare, construită pe TypeScript. Ca platformă, Angular include:

* Un cadru bazat pe componente pentru construirea de aplicații web scalabile
* O colecție de biblioteci bine integrate care acoperă o mare varietate de caracteristici, inclusiv rutare, gestionare a formularelor, comunicare client-server și multe altele
* suite de instrumente pentru dezvoltatori care ajută la dezvoltarea, construirea, testarea și actualizarea codului

Angular oferă o platformă care se poate scala de la proiecte cu un singur dezvoltator la aplicații la nivel de întreprindere. Angular este conceput pentru a face actualizarea cât mai simplu posibilă,. Ecosistemul Angular este format dintr-un grup divers de peste 1,7 milioane de dezvoltatori, autori de biblioteci și creatori de conținut.

Angular este o rescrie completă făcută de aceeași echipă care a construit AngularJS. Inițial, rescrierea lui AngularJS a fost numită „Angular 2”, dar acest lucru a dus la confuzie în rândul dezvoltatorilor. Pentru a clarifica, echipa a anunțat că ar trebui folosiți termeni separați pentru fiecare cadru cu „AngularJS” referindu-se la versiunile 1.X și „Angular” fără ca „JS” să se refere la versiunile 2 și mai sus.

Angular 2.0 a fost anunțat la conferința ng-Europe 22–23 octombrie 2014. Schimbările drastice din versiunea 2.0 au creat controverse considerabile în rândul dezvoltatorilor. Pe 30 aprilie 2015, dezvoltatorii Angular au anunțat că Angular 2 a trecut de la Alpha la Developer Preview. Angular 2 a fost mutat în versiunea beta în decembrie 2015, iar prima versiune candidată a fost publicată în mai 2016. Versiunea finală a fost lansată pe 14 septembrie 2016.

Începând cu versiunea 9, echipa Angular a mutat toate aplicațiile noi pentru a utiliza compilatorul și runtime-ul Ivy. Ei lucrează cu Ivy pentru a îmbunătăți dimensiunile pachetelor de ieșire și vitezele de dezvoltare. Fiecare dintre versiuni este compatibilă cu versiunea anterioară. Echipa de dezvoltare Angular s-a angajat să facă upgrade-uri de două ori pe an.

Toate versiunile majore sunt acceptate timp de 18 luni. Acesta constă în 6 luni de asistență activă, timp în care sunt lansate actualizări și patch-uri programate în mod regulat. Această perioadă este apoi urmată de 12 luni de asistență pe termen lung (LTS), timp în care sunt lansate numai corecțiile critice și corecțiile de securitate.

Cea mai recentă versiune a limbajului este Angular V13, activă din 4 noiembrie 2021.

Bibliografie

* *Arif, Hammad; Qureshi, Habib (2020). Adopting .NET 5: Understand modern architectures, migration best practices, and the new features in .NET 5. Packt Publishing.*[*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-1800560567*](https://en.wikipedia.org/wiki/Special:BookSources/978-1800560567)*.*
* *Metzgar, Dustin (2018). .NET Core in Action. Manning Publications.*[*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-1617294273*](https://en.wikipedia.org/wiki/Special:BookSources/978-1617294273)*.*
* *Price, Mark J. (2020). C# 9 and .NET 5 – Modern Cross-Platform Development. Packt Publishing.*[*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-1800568105*](https://en.wikipedia.org/wiki/Special:BookSources/978-1800568105)*.*
* .Net - <https://en.wikipedia.org/wiki/.NET>, Accesat 14 Dec 2021.
* C sharp - <https://en.wikipedia.org/wiki/C_Sharp_(programming_language)>, Accesat 14 Dec 2021.
* What is .NET? - <https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet>. Accesat 14 Dec 2021.
* C# 9.0 on the record. *.NET Blog*. November 10, 2020, Accesat 14 Dec 2021.
* Site-ul oficial Angular - <https://angular.io/>, Accesat 14 Dec 2021.
* Angular - <https://en.wikipedia.org/wiki/Angular_(web_framework)>, Accesat 14 Dec 2021
* AngularJS vs Angular 2+ - <https://www.sitepoint.com/angularjs-vs-angular/>, Accesat 14 Dec 2021
* AngularJS vs Angular - <https://gorrion.io/blog/angularjs-vs-angular/>