

The screenshot displays the Visual Studio Code (VS Code) interface with a C# program open and running in debug mode. The interface is divided into several panels:

- Top Panel:** Contains the menu bar (File, Edit, Selection, View, Go, Run, ...) and a search bar with the text "hello".
- Left Panel:** Contains the Explorer, Search, Source Control, and Run and Debug icons. The Run and Debug icon is currently selected.
- Variables Panel:** Labeled "VARIABLES", it is currently empty.
- Watch Panel:** Labeled "WATCH", it is currently empty.
- Call Stack Panel:** Labeled "CALL STACK", it is currently empty.
- Breakpoints Panel:** Labeled "BREAKPOINTS", it shows "All Exceptions" and "User-Unhandled..." as selected.
- Editor:** Displays the source code of "Program.cs". The code is as follows:

```
1 using System;
2 class Program {
3     static void Main(string[] args)
4     {
5         Console.WriteLine("Abdallah Hany");
6         Console.WriteLine("Hello DEPI!");
7     }
8 }
9
10
11
```
- Debug Console:** Labeled "DEBUG CONSOLE", it shows the output of the program:

```
Abdallah Hany
Hello DEPI!
The program '[3268] hello.exe' has exited with code 0 (0x0).
```
- Right Panel:** Contains the "Welcome to Copilot" message and buttons for "Build workspace" and "Show project config".

The status bar at the bottom shows the current file is "C#: hello (hello)", the project is "Projects: hello", and the debug target is "Debug Any CPU". The status bar also displays the current line and column (Ln 3, Col 18), the number of spaces (Spaces: 4), the encoding (UTF-8 with BOM), the line ending (CRLF), the language (C#), and the formatter (Prettier).

Report

.Net Versions

- .Net Fw 1.0 2002 → works only on “Windows” , closed source, slow.
- .Net Fw 4.8 2019
- .Net core 1.0 2016 → cross platform ,open source ,for web only.
- .Net core 3.1 2019
- .Net 5 2020 → cross platform , open source ,high performance ,supports all types (Web, Mobile,..)
- .Net 6 2021
- .Net 7 2022
- .Net 8 2023
- .Net 9 2024

Namespace

- Is a way to organise the code in the project .
- Prevents conflictions.

.Net core

- Is a cross platform with a high performance .

Solution

- The container that contains 1 or more project.
- It Facilitates the management of big projects.

what is overhead in run time(jitting) ?

The extra time the system spends during program execution to compile IL (Intermediate Language) into machine code.

what is done to reduce it?

- Compiles IL to native code before the application runs.
- Initially uses a fast, less-optimized compilation
- Monitors usage and recompiles hot methods with full optimization
- Already JIT-compiled methods are cached, avoiding recompilation.

Linked in Article

- إدارة تلقائيه للذاكرة بفضل Garbage Collector وواضح من اسمه انه يمسح اي حاجه مش مستخدمة ✓
- أسهل في التعامل بفضل المكتبات الجاهزه ✓

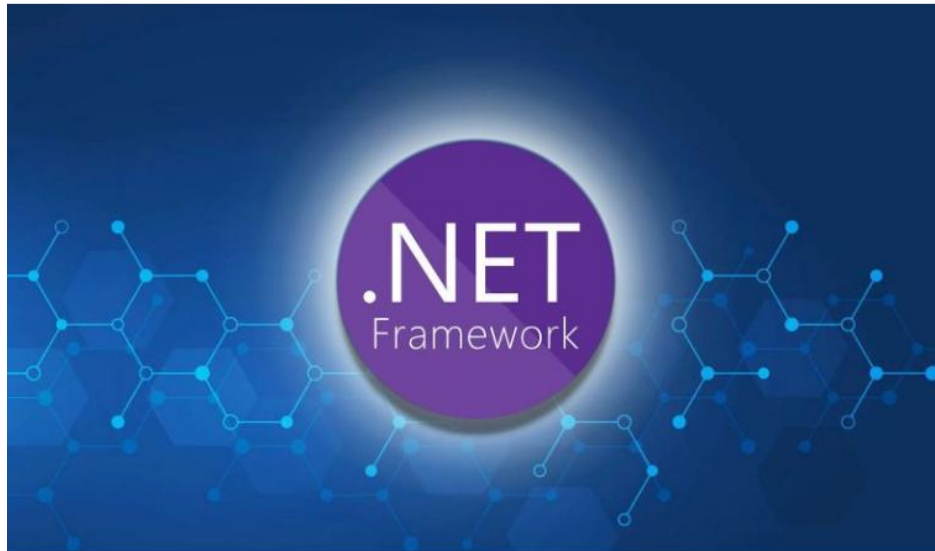
كده نستنتج إن العيب الوحيد لل .Net. أنه بطيئ شويه في البدايه بسبب ال JIT وده عيب ممكن التعايش معاه نظرا لمميزاته الفائقة عن ال Frameworks اللي قبله.

[DotnetFramework#](#)

[Dotnet#](#)

[#C#](#)

[Programming#](#)



Abdallah Hany • You

Aspiring Full-Stack .NET Developer | Computer & Informatics Student | Team ...
now • 📍

ليه عملنا framework .Net ؟

ليه C++/C مش كفايه؟

ببساطة كده لما كنا بنعمل برنامج ب C++ ونعمل compilation كان بياخد وقت لحد ما يُترجم لملف جاهز للتشغيل .exe. لما تدوس تشغيل الملف بيشتغل على طول مفيش compilation أثناء التشغيل وإنما قبل التشغيل ودي بنسميها AOT أو Native Compilation وده طبعاً بيديك أداء عالي وسريع ⚡

لكن 😞

البرنامج مبيشتغلش إلا على نظام تشغيل واحد يعني لو عملته على Windows مبيشتغلش على Mac إلا لو ترجمته تاني.

حاجه كمان أنت بتتعامل مع الذاكرة manually المبرمج هو المسؤول عن تحرير الذاكرة يعني خطأ بسيط ممكن البرنامج ينهار. زود على كده أن الأمان أقل أي حد يقدر يوصل لأي مكان في ال memory.

عشان كده كان لازم تظهر .Net Framework.

بيشتغل ازاي ؟

بنكتب كود بلغة C# بيترجم ل Intermediate Language (IL) لكن ده مش كود جاهز للتشغيل وإنما كود وسيط ، لما تشغل البرنامج بيشتغل داخل بيئة اسمها Common Language Runtime (CLR) بيشتغل ال JIT (Just In Time Compiler) بيترجم الكود ل machine code ويحفظ النسخة المترجمة 'أكيد مش كل مره هيترجم تاني' يعني من الآخر كده بيعمل Compile بعد التشغيل.

مميزات ال .Net Framework :

• كان بيشتغل مع نظام تشغيل واحد لكن بعد التطوير الي .Netcore و .Net5-9. اشتغل على

كل أنظمة التشغيل ✓

• أمان عالي، بيمنع الوصول للذاكرة ما عدا جزء معين ✓