

Part02

1-What is copy constructor?

- A copy constructor makes a new object with the same data as an existing object.
- Changes to the new object do not affect the original.

أي Object ينعمله Instantiate في الكود محتاج مرحلة مهمة جداً وهي ال Initialization. وهنا بييجي دور ال Constructor.

ال Constructor مش مجرد Method عادية: هو Special Block of Code بيشتغل تلقائياً بمجرد ما نستخدم ال Keyword new. بيميزه دائماً إن اسمه بيكون نفس اسم ال Class، ومفيش فيه أي Return Type (ولا حتى void).

في ال OOP، عندنا 3 أنواع أساسية لل Constructors بنحتاجهم في ال Architecture بتاعنا:

1- Default Constructor (No-Args)
وهو ال Constructor اللي ماياخدش أي Arguments. لو إنت معملتش Declare لأي Constructor في ال Class، ال Compiler بيعمل واحد Implicitly.

Use Case: بنستخدمه لما نكون محتاجين نكريت Object فاضي وندي لل Attributes بتاعته Default Values (زي 0 لل integers أو null لل Objects)، وبعدين نعملها Set باستخدام ال Setters لاحقاً.

2- Parameterized Constructor
وهو ال Constructor اللي بيستقبل Parameters عشان يعمل Initialize لل Object بـ Specific State من لحظة ال Creation.

Use Case: لو بنعمل Object من Class اسمه User، الأفضل نبعث ال userId وال username في ال Constructor مباشرة بدل ما نكريت Object فاضي وبعدين نملئ البيانات. ده بيفضن إن ال Object دائماً Valid ومفيش فيه Missing Data.

3- Copy Constructor
بيستقبل Object من نفس ال Class كـ Parameter عشان يعمل منه "Deep Copy" أو "Shallow Copy" (حسب ال Implementation).

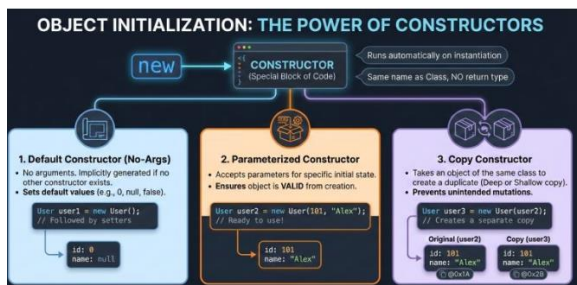
2-

Use Case: مفيد جداً لما تكون عاوز تكريرت Object جديد بنفس ال State بتاعة Object ثاني موجود فعلاً، من غير ما تعمل Reference لنفس ال Memory Location، وده بيحميك من مشكلة ال Unintended Mutations.

من الآخر: بمجرد ما بتعمل Define ال Parameterized Constructor، ال Compiler بيبتل يعمل Generate لل Default Constructor التلقائي، فلو الكود بتاعك (أو ال Framework اللي بتستخدمه) بيعتمد على ال No-Args Constructor، لازم تعمل Define بنفسك.

[Programming #OOP #SoftwareEngineering #CleanCode #Developers#](#)
[#Constructors #TechCommunity](#)

Show translation



3-What is Indexer, when used, as business mention cases u have to utilize it?

-An indexer allows objects to be accessed like arrays using indexes or keys.
It is used when a class represents a collection of data, improving readability and making the code more easy to understand in business like orders, products, and configurations.

4-Summarize keywords we have learnt last lecture?

Struct - Class -Attributes (Fields)-Methods-Constructor-Parameterized Constructor-Constructor Overloading-Encapsulation-Getter / Setter -Properties-Access Modifiers-

Part03

1-When to make the constructor Private? Which design pattern & use case?

- Make a constructor **private** to prevent direct object creation.
- Design Pattern is Singleton
- Ensure only one instance exists for shared resources (database connection).

2-Why in .NET 7.0 the constructor is not obligated to initialize every struct attribute (business-wise)?

-Not all attributes need to be initialized because default values exist. Business-wise, it reduces repetitive code, increases flexibility, and speeds up development.

3-Why allowing developer to override the default constructor from .NET 6 (business-wise)?

-Allows developers to provide custom initial values for structs, ensuring objects start in a valid state and improving API usability and maintainability.

