

1- What is copy constructor?

A **copy constructor** is a special type of constructor in object-oriented programming that is used to create a new object by copying the data from an existing object of the same class or struct. It typically takes one parameter a reference to another object of the same type and duplicates all relevant field or property values into the new instance.

The main purposes of a copy constructor are:

- To create an exact replica of an existing object.
- To control how the copying process is performed, especially when deep copying is needed for reference-type fields.
- To ensure that changes to the new object do not unintentionally affect the original object.

In C#, structs automatically have a copy constructor-like behavior because they are value types, but classes may require explicitly defining one when custom copying logic is needed.

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

An **indexer** in C# is a special property that allows objects to be accessed like arrays using an index. It is mainly used when a class or struct needs to provide indexed access to its internal data without exposing the actual implementation.

When used: Indexers are used when the object represents a collection, mapping, or structured data where retrieval and modification by index or key is needed.

Business use cases:

- Creating custom collections where items are accessed by position.
- Building data mappings (like a phone book) where values are retrieved by a key.
- Wrapping data sources (arrays, lists, database results) to provide clean access.
- Implementing structures like matrices or tables for quick data access.

3- Summarize keywords we have learnt last lecture

C# Keywords -> Always small char

1- Private

2- Private Protected

3- Protected

4- Internal

5- Internal Protected

6- Public