**What is a record in C#?**

* A record is a special C# type introduced in **C# 9.0**
* It's designed for **immutable** data (data that doesn't change after creation)
* Often used for representing simple, data-focused structures

Example:

public record DepartamentDto(string Name, int CoursesCount);

With this:

✔ You get built-in ToString(), Equals(), and GetHashCode()  
✔ It's great for returning data without worrying about changes

**✅ How does record relate to DTO Concept?**

**DTO (Data Transfer Object)** is:

* A lightweight class to move data between systems/layers
* It hides internal logic, only exposes needed fields
* DTO is often immutable — that matches perfectly with record

Using record for DTOs:

✔ Keeps your DTO immutable by default  
✔ Reduces boilerplate code  
✔ Makes code cleaner for API responses

**✅ Should you always use record for DTOs?**

**Not always**, but:

✔ Use record when DTO is simple and immutable  
✔ If the DTO needs to be mutable (e.g., for forms or updates), use a class

**✅ Summary:**

* record fits perfectly for read-only DTOs
* Helps with clean, safe, and simple API models
* Works well in modern .NET APIs