**Research on Code-First**

After researching **Entity Framework Code-First**, I learned that it is one of the main approaches used in the Entity Framework ORM (Object-Relational Mapping) for .NET applications. The Code-First approach allows developers to create a database directly from C# classes, instead of designing the database first.

In this method, I start by defining my model classes (for example, Student, Course, Teacher, etc.) in C#. Each class represents a table in the database, and each property becomes a column. Then, I created a class that inherits **DbContext**, which manages the database connection and gives access to each table through **DbSet**<T>.

Entity Framework automatically generates the database schema based on these classes when I run the program for the first time. If I make changes later, I can use **migrations** to update the database structure without losing existing data.

**Advantages I learned:**

* It allows developers to focus on the object model and business logic first.
* It’s easier to maintain and version-control compared to manual SQL scripts.
* It automatically creates and manages relationships between entities.
* It supports **Data Annotations** and **Fluent API** to customize table names, relationships, and constraints.

**Example:**

For example, a simple Student class with Id, Name, and **EnrollmentDate** can generate a full SQL Server database table automatically when the program runs.

**What I learned overall:**

Through my research and practice, I learned that Code-First is ideal for new projects where the database does not exist yet. It helps simplify development, keep code and data structure consistent, and reduces manual database work.