In modern application development, the data model often evolves as new features are added. This means the underlying database schema must also change to stay in sync with the application’s requirements. Entity Framework Core (EF Core) Migrations provide a structured and incremental way to update the database schema as your application's data model changes, all while preserving existing data.

### ****What is a Migration in EF Core?****

* A **migration** is a way to keep your **C# models and your database schema in sync**.
* Instead of manually writing SQL CREATE TABLE scripts, EF Core:
* Tracks changes you make to your **C# model classes**
* Automatically generates the corresponding **SQL scripts**

Lets you apply those scripts to a **real database** (SQL Server in this case)

### ****Why Use Migrations?****

* **Version control** for your database schema
* Ensures consistency between development, testing, and production
* Great for teams: everyone can share and apply the same schema changes

### ****EF Core CLI Role****

* The **EF Core CLI (Command Line Interface)** gives you tools to:
* **Generate** migration files (dotnet ef migrations add)
* **Apply** migrations to a database (dotnet ef database update)
* **Inspect** schema status (dotnet ef migrations list)