



1. Introducing Entity Framework Core

Database Access in Web Applications

- ■Most applications require storing and loading some kind of data.
- ■These data are typically stored in database.
- ■Working with databases can be a rather cumbersome process.
 - ■Manage connections to the database;
 - Translate data from application to a format the database can understand;
 - \blacksquare Handle a plethora of other subtle problems.

4

What is EF Core?

- ■EF Core is a library that provides an object-oriented way to access databases
- ■EF acts as an object-relational mapper (ORM), communicating with the database and mapping database responses to .NET classes and objects.

5

What is ORM?

- ■ORM is a set of technologies that allows developers to access RDBMS data from an object-oriented programming language.
- ■The idea behind ORM is to create a translation (mapping) mechanism that will allow developers to work with familiar object-oriented (OO) to manipulate tables, views, stored procedures, and more in a relational database.

6

Why Use ORM?

- An ORM tool represents data stored in database tables as objects.
- Expose the data using native .NET types, expose relations using simple .NET properties.
- ■Provide compile time checking.
- Developers do not have to use SQL, they use LINQ to query data.

7

The Result of ORM

- ■EF exposes the database as a set of objects:
 - ■DbContext: database abstraction
 - ■DbSet<TRowType>, collection properties of DbContext: tables abstraction
 - ■Entity, each object within the collection: a row in the corresponding table
 - ■Properties of the TRowType class: columns abstraction

8

