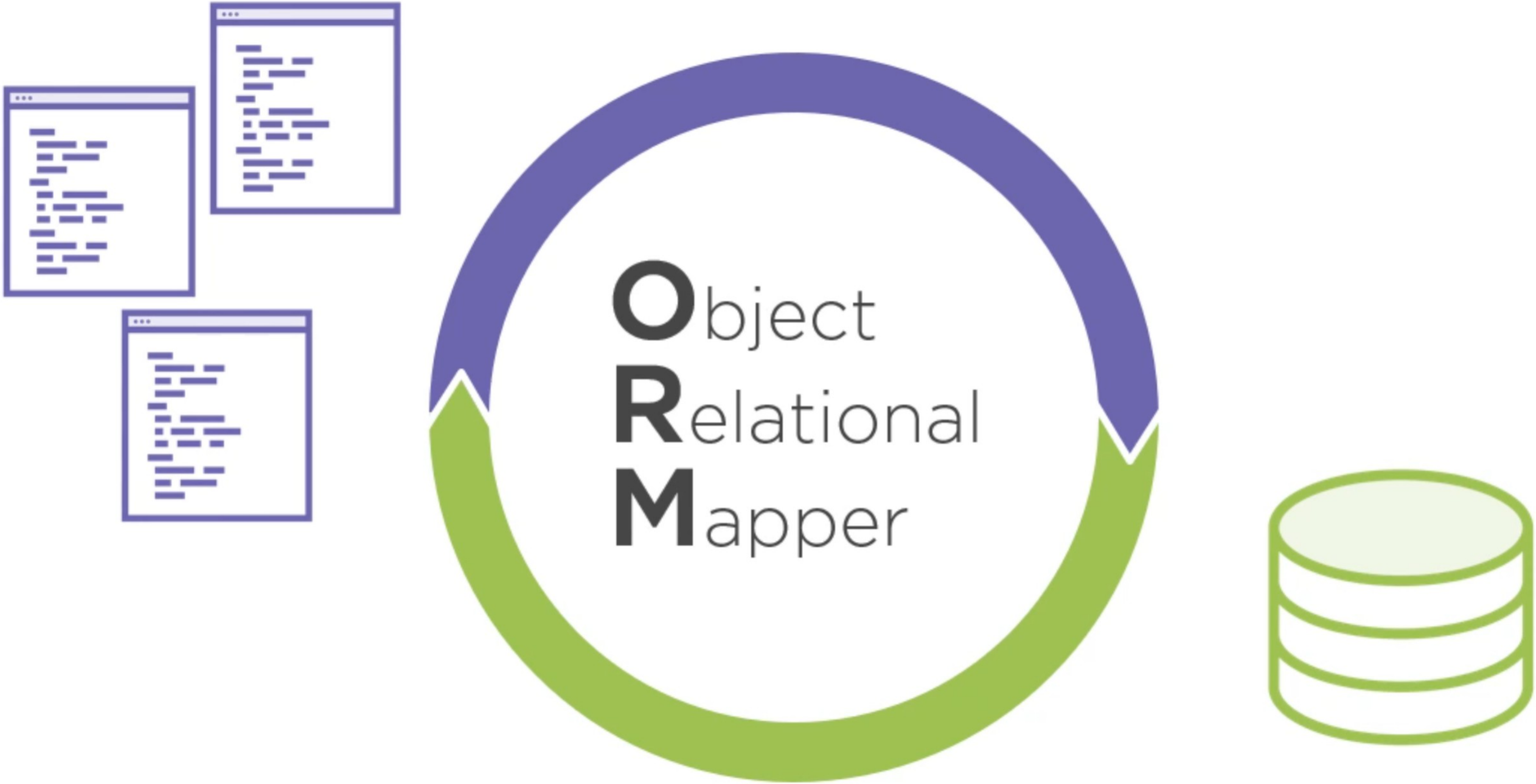
Entity Framework Core

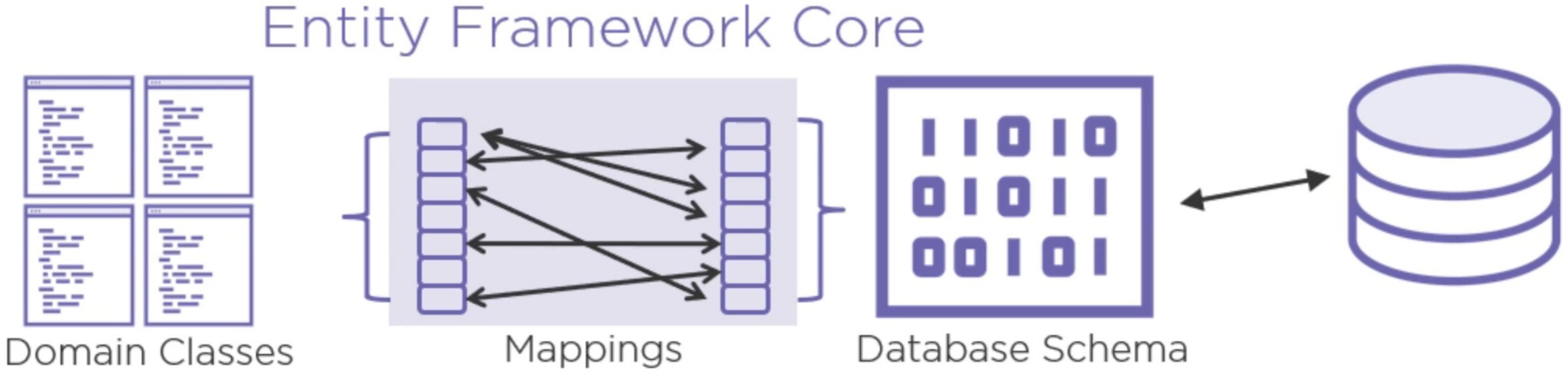
Code First

Lukáš Kubíček

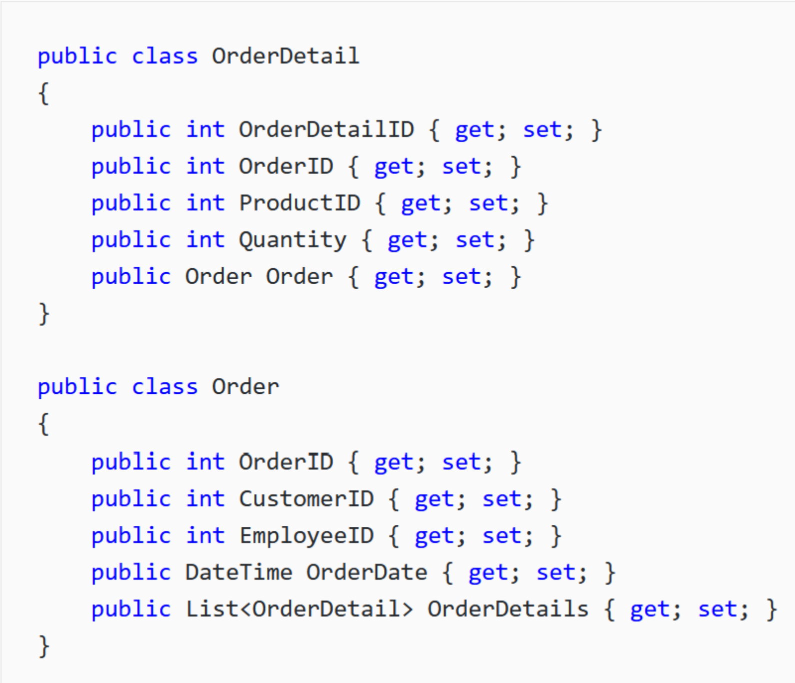
EF Core Is an ORM



Object
Relational
Mapper


Entity Framework Core
Domain Classes
Mappings
11010
01011
00101
Database Schema

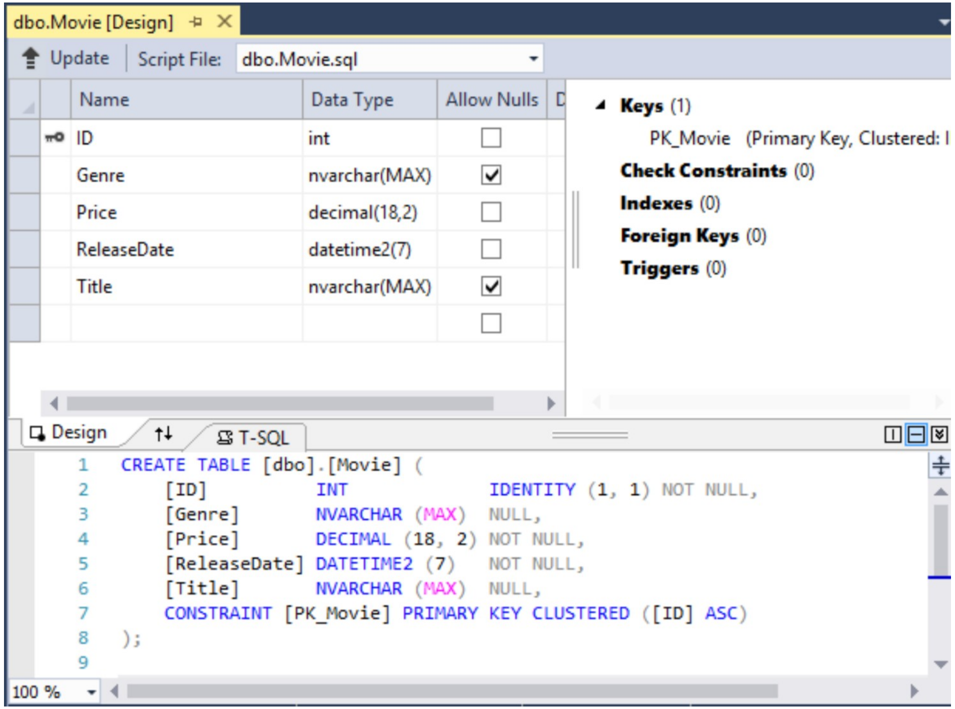
Obousměrné mapování mezi Classes/OOP ↔ SQL DB



```
public class OrderDetail
{
    public int OrderDetailID { get; set; }
    public int OrderID { get; set; }
    public int ProductID { get; set; }
    public int Quantity { get; set; }
    public Order Order { get; set; }
}

public class Order
{
    public int OrderID { get; set; }
    public int CustomerID { get; set; }
    public int EmployeeID { get; set; }
    public DateTime OrderDate { get; set; }
    public List<OrderDetail> OrderDetails { get; set; }
}
```

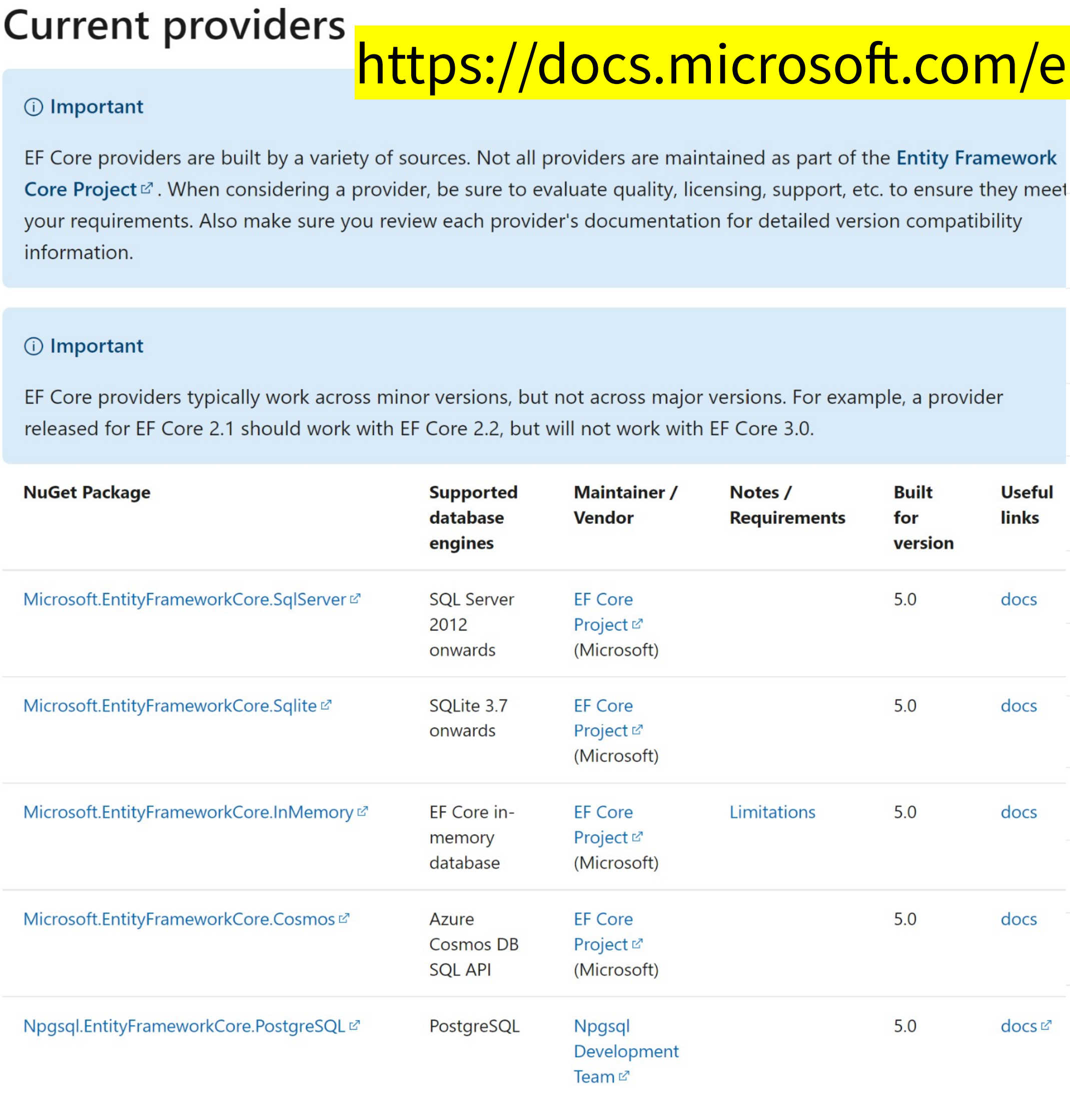
↔



	Name	Data Type	Allow Nulls	D
🔑	ID	int	□	
	Genre	nvarchar(MAX)	☑	
	Price	decimal(18,2)	□	
	ReleaseDate	datetime2(7)	□	
	Title	nvarchar(MAX)	☑	
			□	

Keys (1)
PK_Movie (Primary Key, Clustered: I
Check Constraints (0)
Indexes (0)
Foreign Keys (0)
Triggers (0)

```
CREATE TABLE [dbo].[Movie] (
    [ID]          INT            IDENTITY (1, 1) NOT NULL,
    [Genre]       NVARCHAR (MAX) NULL,
    [Price]       DECIMAL (18, 2) NOT NULL,
    [ReleaseDate] DATETIME2 (7)  NOT NULL,
    [Title]       NVARCHAR (MAX) NULL,
    CONSTRAINT [PK_Movie] PRIMARY KEY CLUSTERED ([ID] ASC)
);
```



Current providers

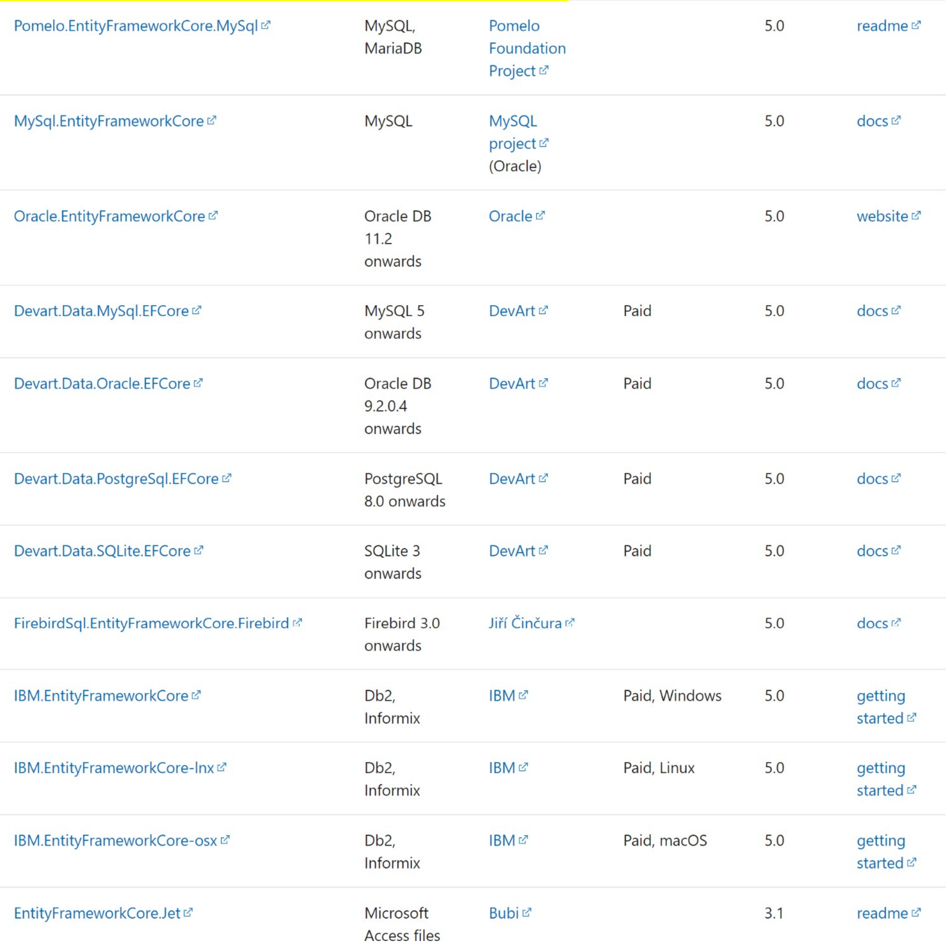
https://docs.microsoft.com/en-us/ef/core/providers/

ⓘ Important

EF Core providers are built by a variety of sources. Not all providers are maintained as part of the **Entity Framework Core Project**. When considering a provider, be sure to evaluate quality, licensing, support, etc. to ensure they meet your requirements. Also make sure you review each provider's documentation for detailed version compatibility information.

ⓘ Important

EF Core providers typically work across minor versions, but not across major versions. For example, a provider released for EF Core 2.1 should work with EF Core 2.2, but will not work with EF Core 3.0.



NuGet Package	Supported database engines	Maintainer / Vendor	Notes / Requirements	Built for version	Useful links
Microsoft.EntityFrameworkCore.SqlServer	SQL Server 2012 onwards	EF Core Project (Microsoft)		5.0	docs
Microsoft.EntityFrameworkCore.Sqlite	SQLite 3.7 onwards	EF Core Project (Microsoft)		5.0	docs
Microsoft.EntityFrameworkCore.InMemory	EF Core in-memory database	EF Core Project (Microsoft)	Limitations	5.0	docs
Microsoft.EntityFrameworkCore.Cosmos	Azure Cosmos DB SQL API	EF Core Project (Microsoft)		5.0	docs
Npgsql.EntityFrameworkCore.PostgreSQL	PostgreSQL	Npgsql Development Team		5.0	docs
Pomelo.EntityFrameworkCore.MySql	MySQL, MariaDB	Pomelo Foundation Project		5.0	readme
MySql.EntityFrameworkCore	MySQL	MySQL project (Oracle)		5.0	docs
Oracle.EntityFrameworkCore	Oracle DB 11.2 onwards	Oracle		5.0	website
Devart.Data.MySql.EFCore	MySQL 5 onwards	DevArt	Paid	5.0	docs
Devart.Data.Oracle.EFCore	Oracle DB 9.2.0.4 onwards	DevArt	Paid	5.0	docs
Devart.Data.PostgreSql.EFCore	PostgreSQL 8.0 onwards	DevArt	Paid	5.0	docs
Devart.Data.SQLite.EFCore	SQLite 3 onwards	DevArt	Paid	5.0	docs
FirebirdSql.EntityFrameworkCore.Firebird	Firebird 3.0 onwards	Jiří Činčura		5.0	docs
IBM.EntityFrameworkCore	Db2, Informix	IBM	Paid, Windows	5.0	getting started
IBM.EntityFrameworkCore-lnx	Db2, Informix	IBM	Paid, Linux	5.0	getting started
IBM.EntityFrameworkCore-osx	Db2, Informix	IBM	Paid, macOS	5.0	getting started
EntityFrameworkCore.Jet	Microsoft Access files	Bubi		3.1	readme

https://docs.microsoft.com/en-us/ef/core/

https://www.codemag.com/Article/2010042/EF-Core-5-Building-on-the-Foundation

https://www.learnentityframeworkcore5.com/

Resources

Entity Framework Core on GitHub github.com/dotnet/efcore

EF Core Documentation docs.microsoft.com/ef

Install dotnet Core SDK dot.net or dotnet.microsoft.com/download

Plan for Entity Framework Core 5.0
https://docs.microsoft.com/en-us/ef/core/what-is-new/ef-core-5.0/plan

EF Core 5.0: Building on the Foundation codemag.com/Article/20100412

EF Core 3.0: A Foundation for the Future codemag.com/Article/1911062

EF Core providers docs.microsoft.com/ef/core/providers

EF Core Migrations in Team Environments
docs.microsoft.com/ef/core/managing-schemas/migrations/teams

Basic Workflow

Define model

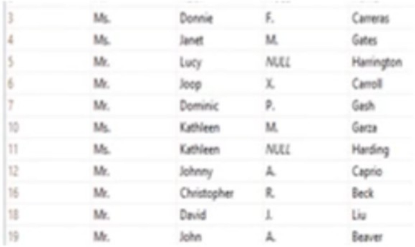
Express & execute query

(from p in people select p)
.ToList()

EF determines
& executes SQL

SELECT * from people

EF transforms results
into your types



3	Ms.	Donnie	F.	Carreras
4	Ms.	Janet	M.	Gates
5	Mr.	Lucy	NULL	Harrington
6	Mr.	Joop	X.	Carroll
7	Mr.	Dominic	P.	Gash
10	Ms.	Kathleen	M.	Garza
11	Ms.	Kathleen	NULL	Harding
12	Mr.	Johnny	A.	Caprio
16	Mr.	Christopher	R.	Beck
18	Mr.	David	J.	Liu
19	Mr.	John	A.	Beaver

User modifies data

Code triggers save

DbContext.SaveChanges

EF determines & executes SQL

UPDATE people
SET Firstname='Julie'
WHERE id=3

Code First

DB First

Model First

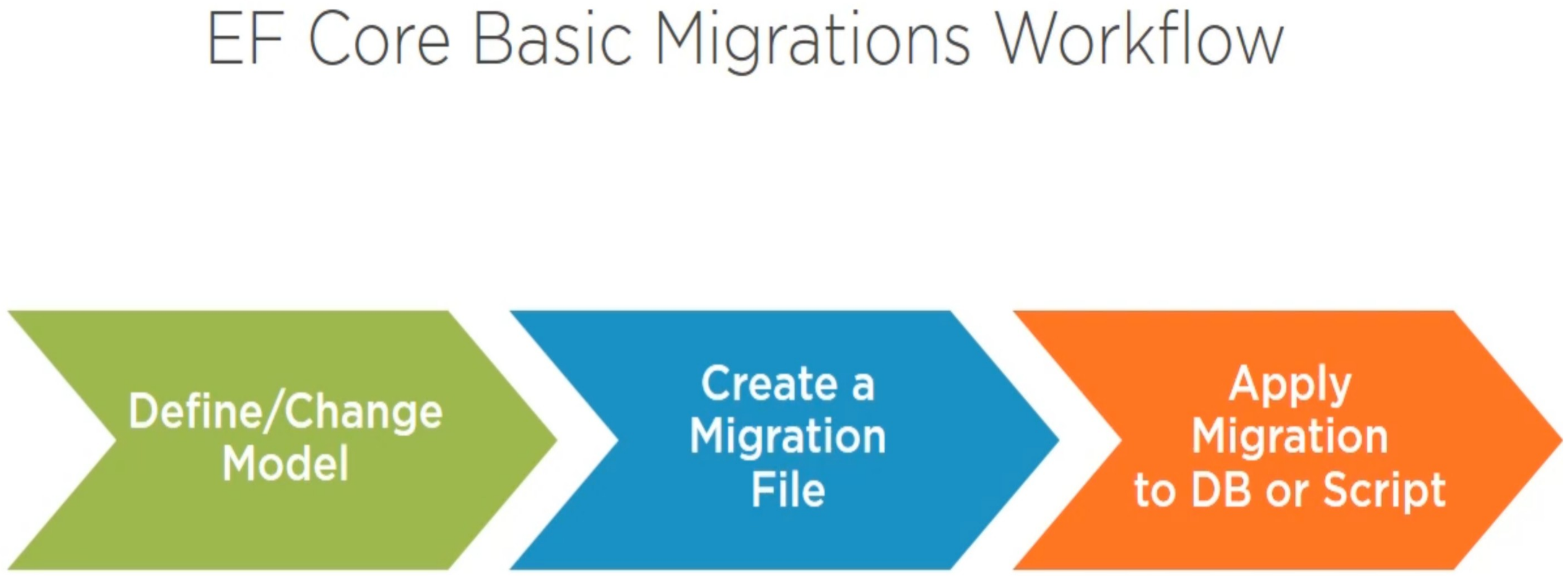
Model –

Domain Classes (Person,…)

+

DbContext

Migrace – změna modelu



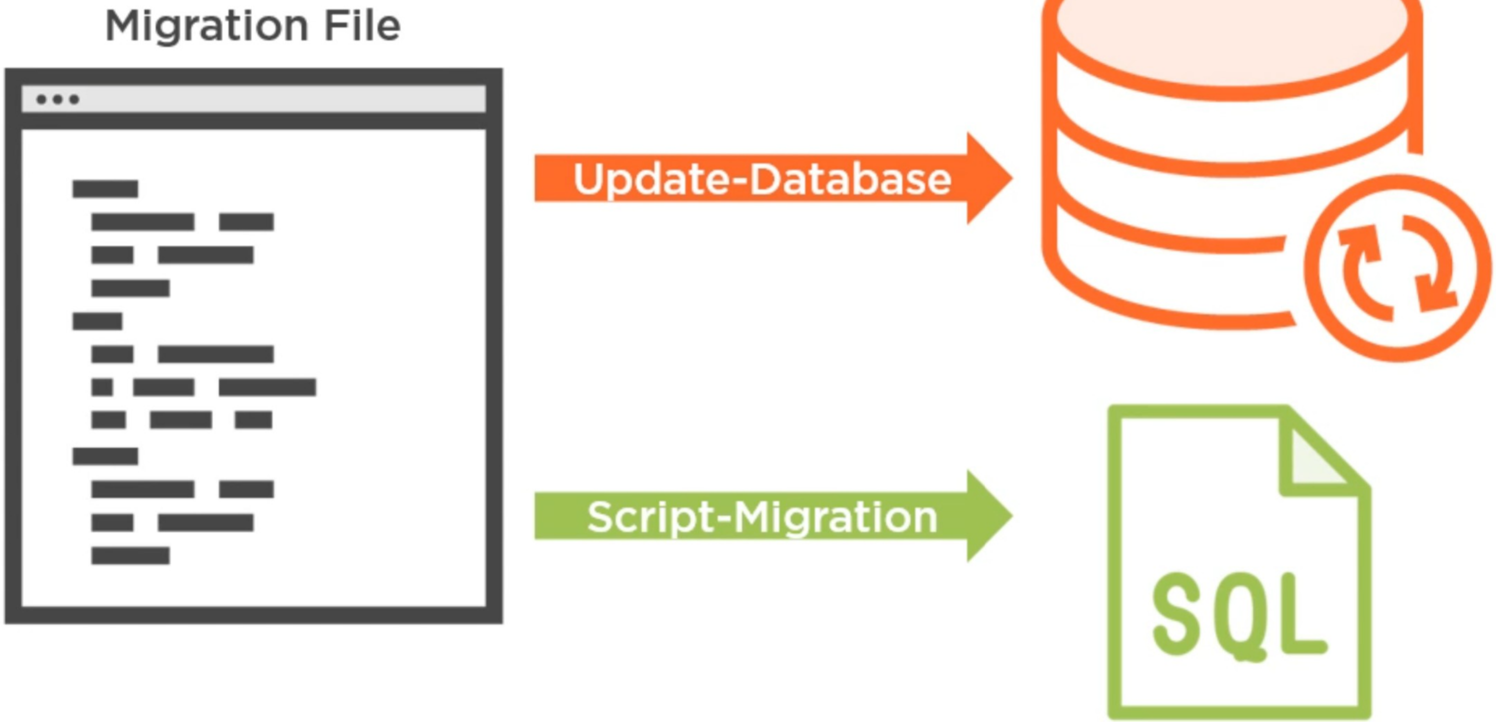
EF Core Basic Migrations Workflow

Define/Change Model

Create a Migration File

Apply Migration to DB or Script

Applying Migrations



How EF Core Determines Mappings to DB

Conventions

Default assumptions

```
property name=column name
```

Override with Fluent Mappings

Apply in DbContext using Fluent API

```
modelBuilder.Entity<Quotes>()
 .Property(q => q.Text)
 .HasColumnName("Line");
```

Override with Data Annotations

Apply in entity

```
[Column("Line")]
public string Text{get;set;}
```