Entity Framework Core 2.x

Building web app with ASP.NET Core

Lương Trần Hy Hiến

hyhien@gmail.com – 0989.366.990

Nội dung

.NET Core

Master II NHẤT NGHỆ

- Giới thiệu Entity Framework
- EF Core API (Create, Query, Saving)
- Tham khảo

HIENLTH

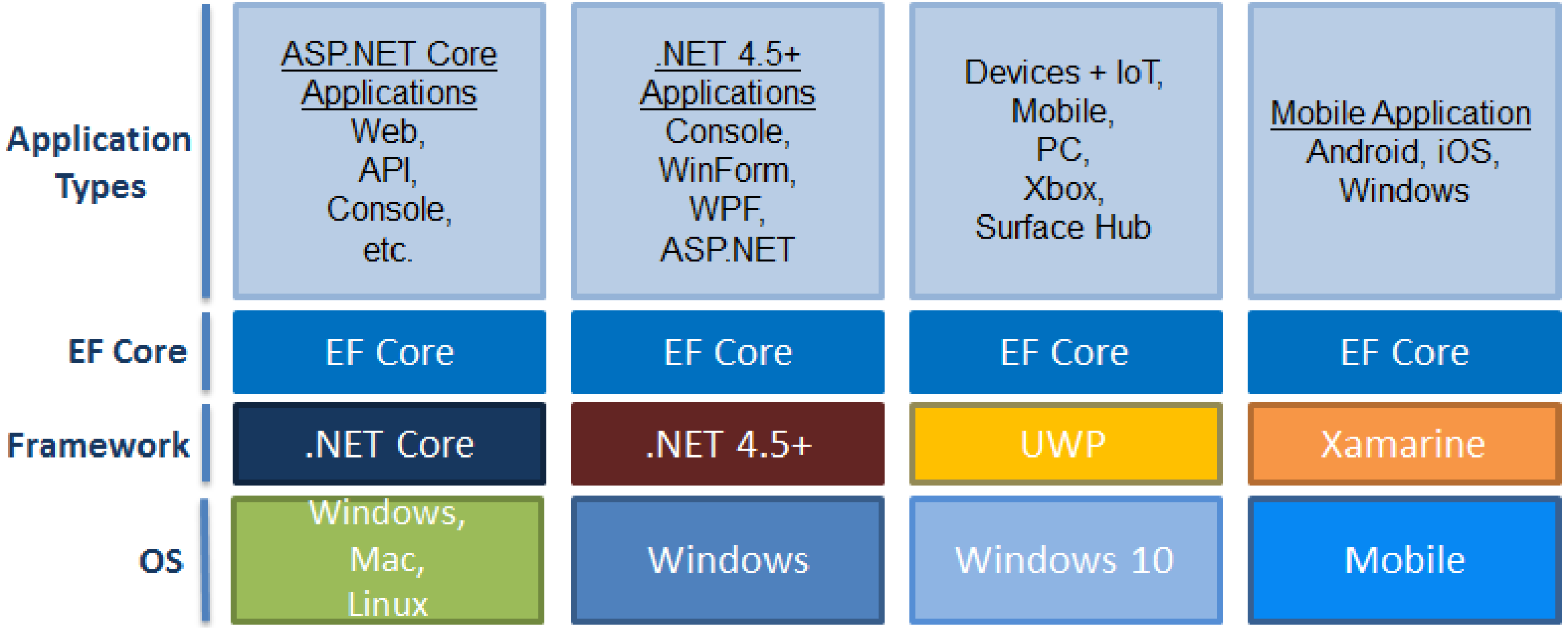
Entity Framework (EF) is an object/relational mapper (O/RM) for .NET

Entity Framework Core
is a lightweight, extensible,
and cross-platform version
of Entity Framework

About EF 2.0

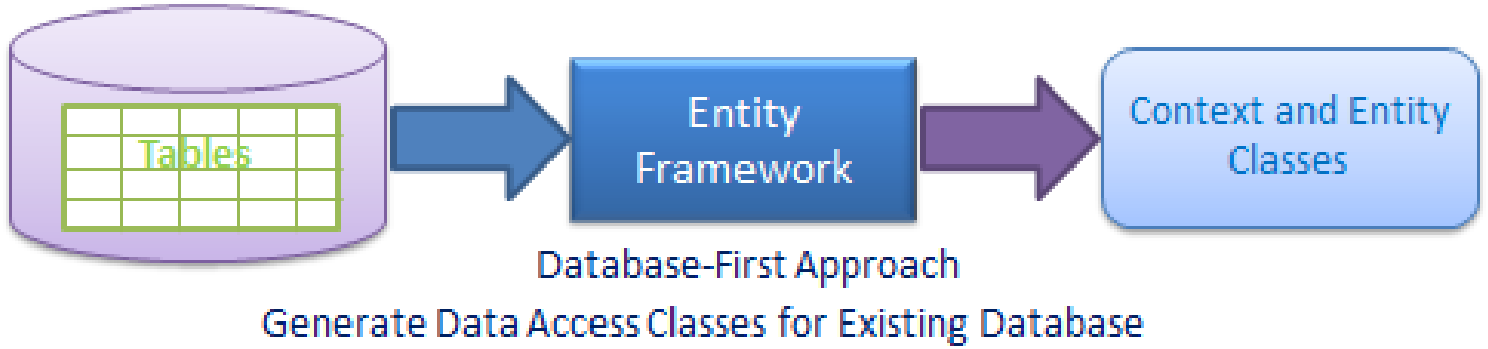
- Entity Framework is an **ORM** (Object Relational Mapper) **framework**.
- Entity Framework is an **open source** framework by Microsoft
- The EF enables developers to work with data in the form with data in the form of domain
- EF Core version history

EF Core Version	Release Date
EF Core 2.0	August 2017
EF Core 1.1	November 2016
EF Core 1.0	June 2016

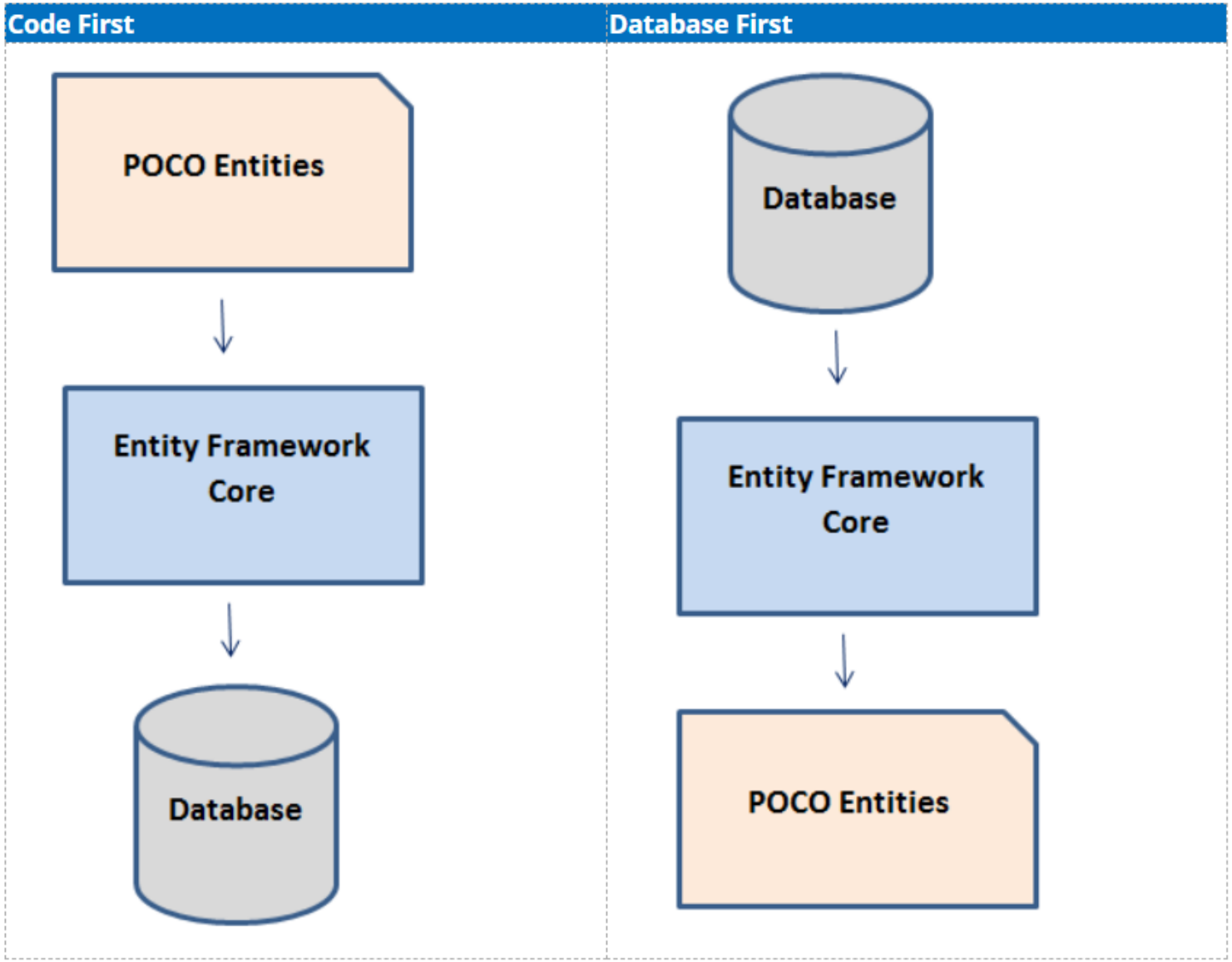


EF Core Development Approaches

- EF Core supports two development approaches
  - 1) Code-First
  - 2) Database-First.



Code First/DB First



	Database First	Code First
Code Generation	Create the database schema using your preferred tools (e.g SQL Server Management Studio) and import that database and schema using a script to scaffold everything for you.	Write classes with properties you want to keep track of and then define relationships between other classes.
Syncing Changes	Create a migration that represents the changes and runs the script against the database.	You create the **migrations** when you need them, and the code will use those files (and a corresponding table in the target database) to keep the schema changes in sync with what the code expects. In other words, you can manage your database schema without ever writing any SQL.
Type of project best suited for	It makes more sense to use database-first for existing project since you don't need to make big changes to scale the database. You could also use this approach for new projects when a database is developed separately by DBA's and owned by another team.	Code-first makes more sense if you're developing a new application from scratch as you can grow and evolve your model as you develop.

EF Core vs EF 6

- EF Core support some features same as EF6
  1. DbContext & DbSet
  2. Data Model
  3. Querying using Linq-to-Entities
  4. Change Tracking
  5. SaveChanges
  6. Migrations

More details : https://docs.microsoft.com/en-us/ef/efcore-and-ef6/features

EF Core vs EF 6

- EF Core have new features which are not supported in EF 6
  1. Easy relationship configuration
  2. Batch INSERT, UPDATE, and DELETE operations
  3. In-memory provider for testing
  4. Support for IoC (Inversion of Control)
  5. Unique constraints
  6. Shadow properties
  7. Alternate keys
  8. Global query filter
  9. Field mapping
  10. DbContext pooling
  11. Better patterns for handling disconnected entity graphs

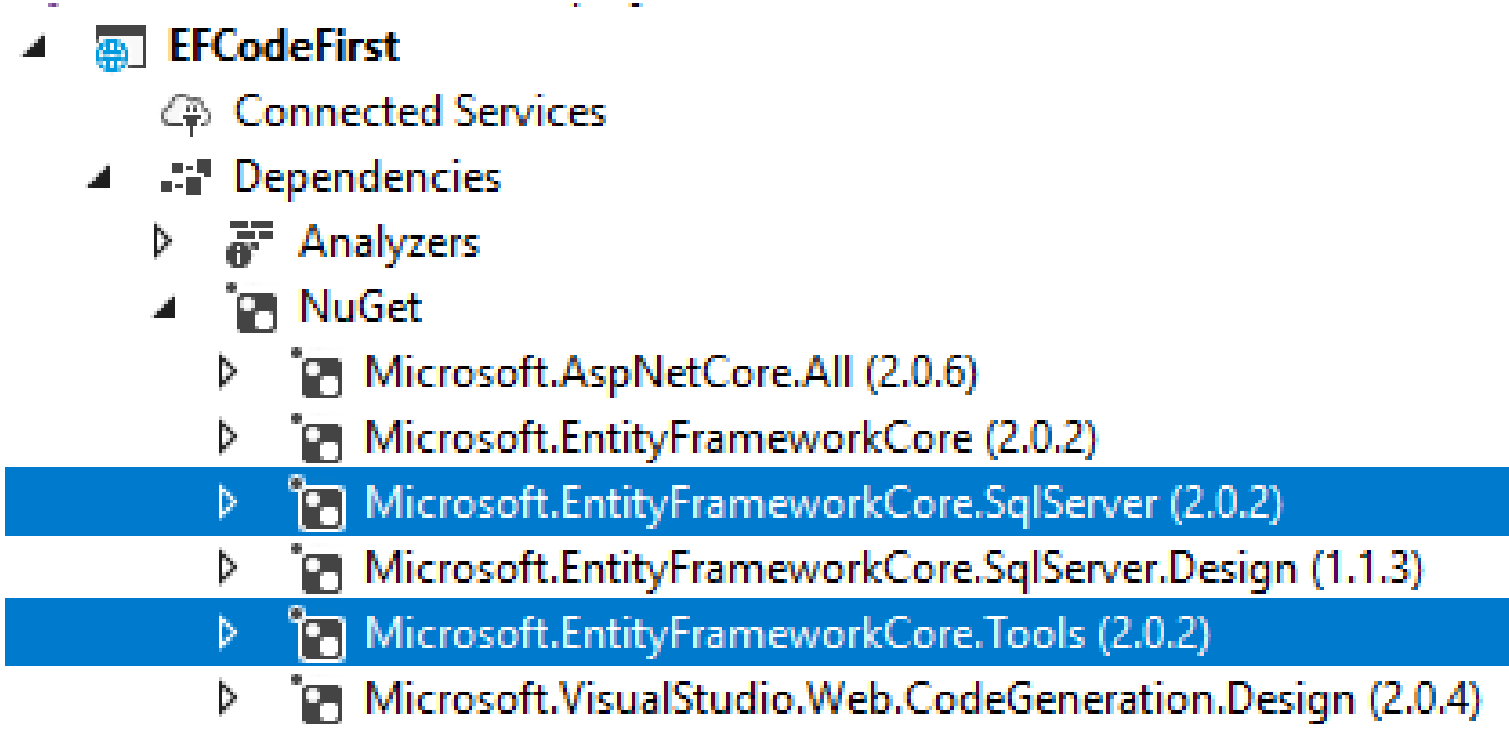
More details: https://docs.microsoft.com/en-us/ef/efcore-and-ef6/features

EF Core Database Providers

Database	NuGet Package
SQL Server	Microsoft.EntityFrameworkCore.SqlServer
MySQL	MySql.Data.EntityFrameworkCore
PostgreSQL	Npgsql.EntityFrameworkCore.PostgreSQL
SQLite	Microsoft.EntityFrameworkCore.SQLite
SQL Compact	EntityFrameworkCore.SqlServerCompact40
In-memory	Microsoft.EntityFrameworkCore.InMemory

Install NuGet packages

- EF Core DB provider
  - Microsoft.EntityFrameworkCore.SqlServer
- EF Core tools
  - Microsoft.EntityFrameworkCore.Tools



EF Core NuGet Packages

- Talk to SQL Server
- Microsoft.EntityFrameworkCore.SqlServer

- Database Updates (“Migrations”)
- Microsoft.EntityFrameworkCore.Design

- Reverse Engineer a SQL Server Database
- Microsoft.EntityFrameworkCore.SqlServer.Design

EF Core API

- Creating Model in EF Core
- Querying Data in EF Core
- Saving Data in EF Core

EF Core: **Create Model**

- Code First
  - Model Class → DbContext → Migration (PM Console Command)
- Database First
  - Database → Migration → Model Class

Import Existing Database to EF Core

- PM> Scaffold-DbContext “connection_string" Microsoft.EntityFrameworkCore.SqlServer

- Ví dụ:
- Scaffold-DbContext "**Server=.; Database=EFCoreDBFirst-QLBH;Integrated Security=True;**" Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models

Migration

- Migration is a way to keep the database schema in sync with the EF Core model by preserving data.

PM Command	Usage
add-migration <migration name>	Creates a migration by adding a migration snapshot.
remove-migration	Removes the last migration snapshot.
update-database	Updates the database schema based on the last migration snapshot.
script-migration	Generates a SQL script using all the migration snapshots.

MyDbContext : DbContext

- public virtual DbSet<Machine> Machine { get; set; }
- ...
- protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

```
        {
            if (!optionsBuilder.IsConfigured)
            {

optionsBuilder.UseSqlServer(@"Server=.;Database=BegEFCore2;Integrated Security=SSPI;");
            }
        }
```

MyDbContext : DbContext

- protected override void OnModelCreating(ModelBuilder modelBuilder)

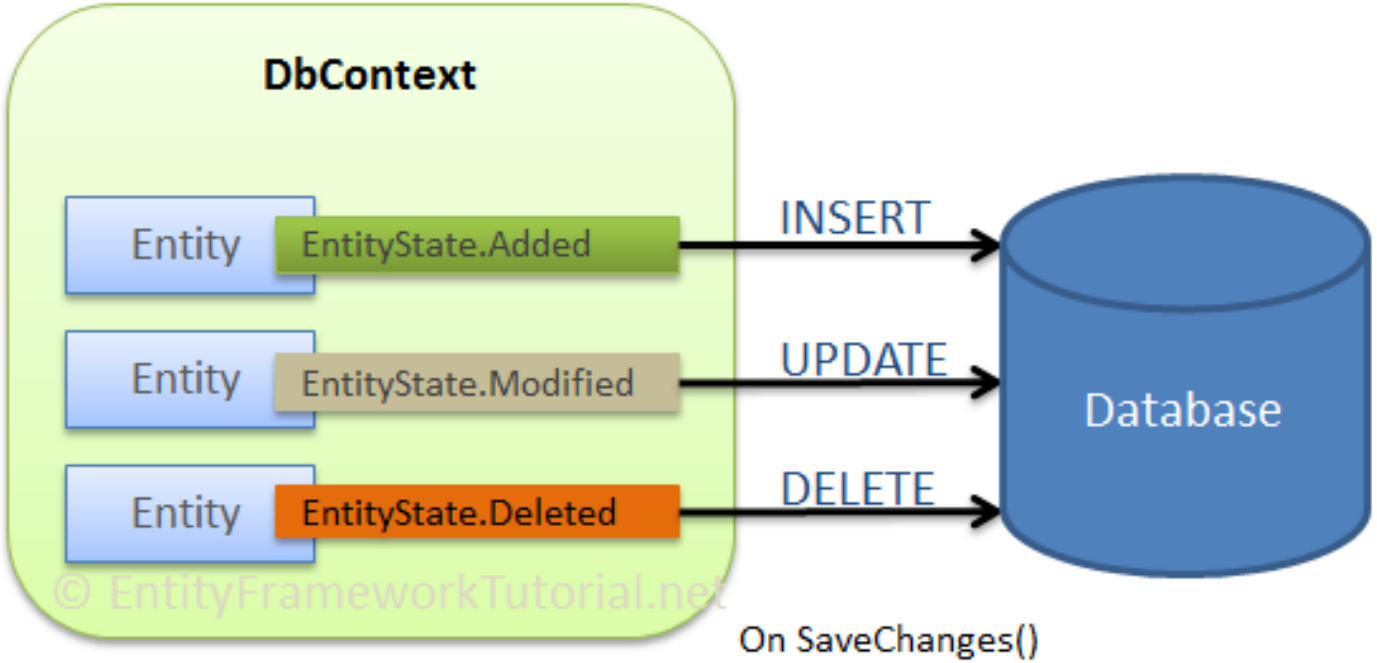
```
{
        modelBuilder.Entity<HangHoa>(entity =>
        {
            entity.HasKey(e => e.MaHh);

            entity.Property(e => e.MaHh).HasColumnName("MaHH");

            entity.Property(e => e.TenHh)
                .IsRequired()
                .HasColumnName("TenHH")
                .HasMaxLength(50);

            entity.HasOne(d => d.MaLoaiNavigation)
                .WithMany(p => p.HangHoa)
                .HasForeignKey(d => d.MaLoai);
        });
}
```

Mapping between :
C# data type to SQL Server column data type

C# Data Type	Mapping to SQL Server Data Type
int	int
string	nvarchar(Max)
decimal	decimal(18,2)
float	real
byte[]	varbinary(Max)
datetime	datetime
bool	bit
byte	tinyint
short	smallint
long	bigint
double	float
char, object	No mapping
sbyte	No mapping (throws exception)

EF Core: **Saving Data**

- context.SaveChanges()



EF Core: **Query Data**

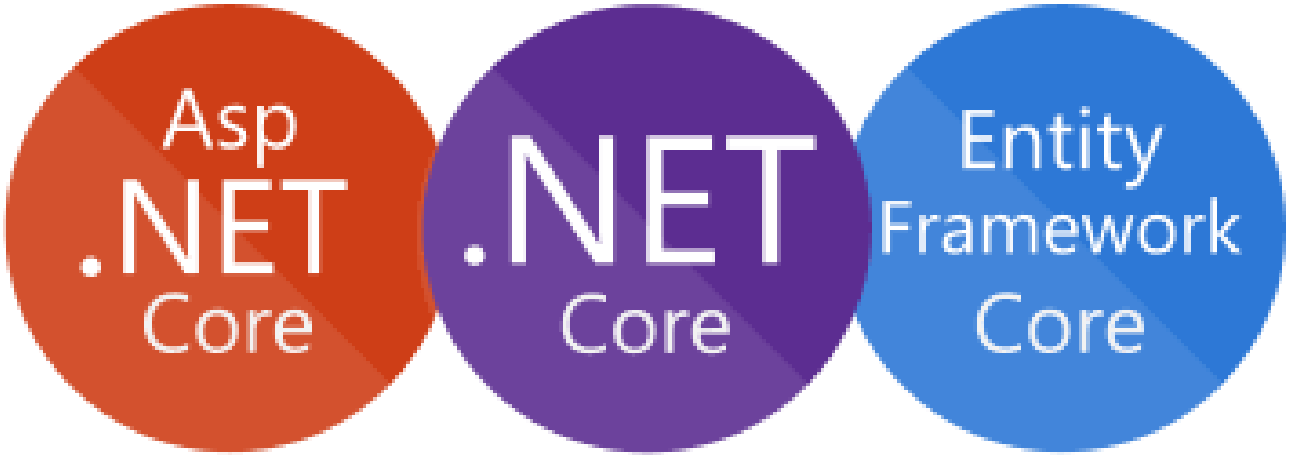
- Using LINQ to Entities
- LINQ Extension Methods
  - First(), FirstOrDefault()
  - Single(), SingleOrDefault()
  - ToList()
  - Count(), Min(), Max(), Sum(), Average()
  - Last(), LastOrDefault()

References

- ASP .NET: http://www.asp.net
- .NET Core: https://www.microsoft.com/net
- .NET Web Dev Blog: https://blogs.msdn.microsoft.com/webdev

- Tutorials: https://docs.microsoft.com/en-us/aspnet/core
- docs.efproject.net
- github.com/aspnet/EntityFrameworkCore
- http://www.entityframeworktutorial.net/efcore

Q&A



Chương trình ASP.NET Core 2.0

- http://nhatnghe.com/chuongtrinhhoc/aspcore
- 108 giờ, 4 triệu
  - Chiều T7, sáng CN: 4 giờ/buổi
  - Tối 357/246: 3 giờ/buổi
- Nội dung
  - Ôn tập C# cơ bản
  - HTML5/CSS3/jQuery/BootStrap cơ bản (FrontEnd development)
  - ASP.NET Core MVC/Web App
  - Entity Framework Core
  - Web API

THANKS FOR COMING!

Resources for seminar

- https://github.com/hienlth/aspcore