Migration Notes

PM> Install-Package Microsoft.EntityFrameworkCore.SqlServer

PM> Install-Package Microsoft.EntityFrameworkCore.Tools

PM> Install-Package Microsoft.EntityFrameworkCore.Sqlite

Create classes

Create DbContext derived class

See: <https://docs.microsoft.com/en-us/ef/core/cli/dbcontext-creation?tabs=dotnet-core-cli>

PM> add-Migration InitialCreate

Or

PM> Add-Migration InitialCreate -OutputDir Your\Directory -Namespace Your.Namespace

PM> Update-Database

Remove-Migration

If you get error:

add-Migrations : The term 'add-Migrations' is not recognized as the name of a cmdlet, function, script file, or operable program.

Check the spelling of the name, or if a path was included, verify that the path is correct and try again.

At line:1 char:1

Reason: Tools not installed (Microsoft.EntityFrameworkCore.Tools)

If you get:

Build started...

Build succeeded.

No DbContext was found in assembly 'SampleDataService2'. Ensure that you're using the correct assembly and that the type is neither abstract nor generic.

Add:

public class DataServiceDbContext : DbContext

{

public DbSet<Order> Orders { get; set; }

public DbSet<OrderItem> OrderDetails { get; set; }

}

If you get:

No database provider has been configured for this DbContext. A provider can be configured by overriding the 'DbContext.OnConfiguring' method or by using 'AddDbContext' on the application service provider. If 'AddDbContext' is used, then also ensure that your DbContext type accepts a DbContextOptions<TContext> object in its constructor and passes it to the base constructor for DbContext.

* Supply DbContextOptions<dirivedContext> options on contructor, pass to base
* Use Constructor with no parameters for design time usage: Migrations
* Override OnConfiguring to extablish provider database information.