**APLICANDO ENTITY FRAMEWORK A APLICAÇÃO**

**1 –** Adicionar uma referência ao Entity Framework à aplicação.

- Caminho para abrir o package manager console:

“TOOLS” -> “NUGET PACKAGE MANAGER” -> “PACKAGE MANAGER CONSOLE”

- Mandando instalar o pacote. Comando para instalar o pacote:

Install-Package entityframework

- No momento em que for inserido o comando o VS vai na internet e baixa a DLL e fazer uma referência para o meu projeto automaticamente.

**2 –** Utilizando as EF dentro do Contexto (Classe que representa o banco de dados)

- Adicione a Using do EF a sua classe para poder utilizar sua biblioteca de classes: “using System.Data.Entity; ”

- Faça com que o Contexto herde de uma classe no EF que vai mapear para mim tudo que eu preciso para poder trabalhar com meu BD. A classe é a: “DbContext”

- Daí eu posso começar a mapear as minhas classes para o meu BD. Ou seja apartir da minha classe o EF gerará uma tabela para meu BD.

- Para isto as propriedades da minha classe terão que ser do tipo “DbSet<NomeDaClasse>”. Ou seja, é ela que vai criar a tabela no BD e mapear ela para minha propriedade da classe em que estiver trabalhando.

- Crie uma Connection String no AppConfig para que o EF saiba o BD que queremos trabalhar.

- O nome do banco precisa ser o mesmo do contexto.

- Crie uma instância da classe Contexto. Chame a propriedade criada do tipo EF e mande Add o produto que queremos criar no BD:

“DBProduto banco = new DBProduto();

banco.EFProduto.Add(produto01); ”

- Logo isto precisamos commitar isto no BD:

“banco.SaveChanges(); ”.

**3 –** Instalando o Migration para quando houver alteração no Contexto refletir no BD

- Habilitando o Migration comando: “Enable-Migrations”. Ao habilitar o Migration ele cria um repositório “Migrations” onde existe um ponto de restauração, uma classe com o ponto inicial para criação das tabelas até aquele ponto.

- Para isto é preciso executar o comando: “add-Migrations ‘Nome\_dado\_a\_migracao’ ”.

Assim ele altera somente a classes de negócio que reflete o BD.

- Para que a alteração seja feita no BD é preciso executar o comando:

“Update-Database -Verbose”

- Para gerar um script das alterações feitas no Migration comando:

“Update-Database -Script -SourceMigration:”Ponto Inicial” -TargetMigration:" ”Ponto Inicial”

"Ponto Final”

“Update-Database -Script -SourceMigration:$InitialDatabase -TargetMigration:"12062016-Adicionado-Cateria-Contexto"”

- Para atualizar o banco de dados comando: “Update-Database -Force”

4 – Data Notaition

O EF segue algumas convenções e uma dela é, para que o EF crie chaves primarias ele utiliza da primeira propriedade que esteja com o nome de “ID” ou nome da classe seguido de id “NomeDaClasseID”.

Se não quiser seguir esta convecção será preciso informar para o EF qual a propriedade que será usada como chave primaria:

[Key]

Public int Codigo { get; set;}