

Gravando Produtos no Banco de dados

Transcrição

Tendo os objetos — os livros —, começaremos a importá-los como produtos do nosso banco de dados, varrendo-os e inserindo-os um a um com o laço `foreach`. Assim, para cada `livro` contido na coleção `livros`, acessaremos o que seria a tabela do banco de dados, que no *Entity Framework* chamamos de `DbSet`, e está dentro de `contexto`.

Passaremos o nome da classe que representa uma tabela do banco, no caso, a tabela `Produto`, e então chamaremos o método para adicionar uma nova instância de `Produto` da nossa lista de produtos, passando o código, nome e preço do livro como argumentos.

Entretanto isto ainda não é o suficiente, pois quando adicionamos tudo isso ao `DbSet` de `Produto`, o que fazemos é adicionar informações em memória, as quais ainda não são de fato enviadas ao banco de dados, exigindo que isto seja feito todas as vezes em que um `produto` for adicionado ao `DbSet`.

Para que isto ocorra de uma vez só, chamaremos o método `SaveChanges()` a partir de `contexto` e, aí sim, quando tivermos todos os produtos carregados em memória, prontos para serem adicionados ao banco de dados, fazemos a gravação no banco de dados de uma única vez. Nesta linha, também acrescentaremos um *breakpoint*, e então rodaremos a aplicação novamente.

```
public void InicializaDB()
{
    contexto.Database.EnsureCreated();

    var json = File.ReadAllText("livros.json");
    var livros = JsonConvert.DeserializeObject<List<Livro>>(json);

    foreach (var livro in livros)
    {
        contexto.Set<Produto>().Add(new Produto(livro.Codigo, livro.Nome, livro.Preco));
    }

    contexto.SaveChanges();
}
```

Recapitulando:

- o serviço será subido;
- quando o arquivo de produtos for lido, cada livro será inserido ao `DbSet`;
- eles serão enfim gravados no banco de dados, com `SaveChanges()`.

Atualizaremos o SQL Server Object Explorer, e em localdb, temos "Databases > CasadoCodigo > Tables". Clicaremos com o lado direito do mouse em "dbo.Produto", selecionaremos "View Data" e visualizarmos os dados. Então, teremos a carga dos produtos a partir do nosso arquivo `livros.json`.