

Preparando o ambiente (Obrigatório)

Instalação do Visual Studio e SQL Server

Como pré-requisito de aplicações, você vai precisar do Visual Studio 2017 Community e do SQL Server Express. Para baixar os dois, é só visitar a [página do Visual Studio \(https://www.visualstudio.com/pt-br/vs\)](https://www.visualstudio.com/pt-br/vs) e obter o instalador do Visual Studio 2017. O SQL Server já vem embutido nele.

Download do projeto

Para rodar os exemplos dos vídeos em sua máquina, baixe o projeto inicial no link [aqui \(https://s3.amazonaws.com/caelum-online-public/635-entity-framework-core/01/projetos/Alura.Loja.zip\)](https://s3.amazonaws.com/caelum-online-public/635-entity-framework-core/01/projetos/Alura.Loja.zip).

Depois que você fizer o download do projeto, basta descompactá-lo numa pasta qualquer para abri-lo no Visual Studio. Aqui colocamos o projeto na pasta que o VS usa como padrão para guardar seus projetos (no Windows 10, fica em `c:\users\<nome do seu usuário>\Documents\Visual Studio 2017\Projects`).

Já no Visual Studio, vamos abrir a solução clicando em Arquivo, Abrir, Projeto\Solução, e escolhendo a pasta que descompactamos no passo anterior. A solução é do tipo Console Application, apenas para mostrar os conceitos e ferramentas que serão discutidas no curso.

Criação do banco de dados

Lembrando que esse curso é uma introdução ao Entity Framework Core e irá aplicar sua teoria em um banco de dados completamente novo. Se sua necessidade é aprender **como integrar o EF Core a um banco de dados já existente**, temos um outro curso dividido em duas partes voltado para esse propósito:

- [Entity Framework Core parte 1: Mapeando um banco pré-existente \(https://cursos.alura.com.br/course/entity-framework-core-banco-pre-existente-parte1\)](https://cursos.alura.com.br/course/entity-framework-core-banco-pre-existente-parte1)
- [Entity Framework Core parte 2: Mapeando um banco pré-existente \(https://cursos.alura.com.br/course/entity-framework-core-banco-pre-existente-parte2\)](https://cursos.alura.com.br/course/entity-framework-core-banco-pre-existente-parte2)

Outro passo necessário para rodar os exemplos é criar o banco de dados utilizado nessa solução. Abra o Pesquisador de Objetos do SQL Server no menu Exibir e clique na aba SQL Server para mostrar os servidores disponíveis em sua máquina. Abra o servidor `(localdb)`, clique com o botão direito em cima da aba *Bancos de Dados* e escolha a opção *Adicionar Novo Banco de Dados*.

Na janela de diálogo a seguir, defina como nome do banco `LojaDB` e aceite a sugestão do VS para a localização. Ao clicar em OK será criado o banco.

Criação da tabela Produtos

Agora que temos um banco, vamos criar a tabela de produtos. Primeiro, abra o arquivo `ddl-produtos.txt`, que está disponível no projeto que você baixou, e copie todo o seu conteúdo usando `Ctrl+C`. Abra o banco recém criado *LojaDB* e com o botão direito na aba *Tabelas*, escolha a opção *Adicionar Nova Tabela....* Na janela que será aberta, procure a aba *T-SQL*

e substitua todo o código nela pelo texto que você copiou na área de transferência. Clique no botão *Atualizar* e em seguida *Atualizar Banco de Dados*. Confira que a tabela *Produtos* está criada no banco de dados *LojaDB*.

Pronto! Agora você está preparado para acompanhar o curso.