

API de Vendas e Pedidos

Passo a passo lógico:

Características do modelo Produto:

```
string Id  
string Nome  
string Descricao  
decimal Preco  
int Estoque  
string Categotia
```

Armazenamento temporário na memória:

Os produtos são armazenados temporariamente em **Data → ProdutoData**

```
List<Produto> produtos  
Junto de métodos Getter/Setters e Remove
```

Requisições HTTP:

Listar todos os registros. (ROTA_GET)

Buscar registro específico pelo ID. (ROTA_GET)

Adicionar novos registros. (ROTA_POST)

Excluir registros. (ROTA_DELETE)

GET / → Retorna uma mensagem inicial.

GET /api/produtos → Lista todos os produtos.

GET /api/produtos/{id} → Retorna um produto específico.

POST /api/produtos → Adiciona um novo produto.

DELETE /api/produtos/{id} → Remove um produto pelo Id.

As estruturas e comandos utilizados:

Rotas definidas com o MapGet, MapPost e MapDelete, típicas de uma **Minimal API**.

Injeção de dependência para manipular os dados de produto.

Guid utilizado para gerar automaticamente o **Id** dos produtos.

Isso elimina a necessidade de informar o **Id** manualmente ao usar a rota POST.

List<Produto> é usada como repositório temporário, garantindo simplicidade e leveza ao projeto.

Justificativas que permitam a compreensão do método adotado.

A API foi projetada com o objetivo de ser **simples, didática e funcional**, ideal para aprendizado e testes de integração.

A utilização de uma **lista em memória** permite foco na lógica de rotas e controle de dados sem a complexidade de um banco.

O padrão de **Minimal API** oferece clareza e rapidez no desenvolvimento, facilitando manutenção e compreensão do código.

Application URL: <http://localhost:5065>