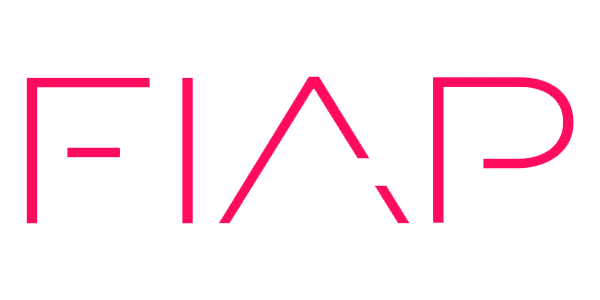
****

**# FIAP - Checkpoint 1: API de Pedidos**

**Disciplina: Arquitetura SOA e Web Services**

**Aluno 1: Enzo Vazquez Sartorelli RM94618  
Aluno 2: Nicolas Nogueira Boni RM551965**

**Aluno 3: Eduardo de Oliveira Nistal RM94524**

**Link GitHub: https://github.com/efsartorelli/fiap-checkpoint1**

**| Objetivo do Projeto:**

Esta API RESTful permite cadastrar, listar, atualizar e deletar pedidos de clientes.

Foi desenvolvida com Spring Boot 3.4.4, usando banco de dados H2 em memória, com estrutura MVC (Model-Service-Repository-Controller).

**| Tecnologias Utilizadas:**

- Java 17

- Spring Boot 3.4.4

- Spring Web

- Spring Data JPA

- H2 Database

- Lombok

- Gradle

- Postman

**| Estrutura de Pastas:**

src/main/java/br/com/fiap/checkpoint1

├ model

├ repository

├ service

├ controller

**| Endpoints da API:**

|  |  |  |
| --- | --- | --- |
| **Método** | **Endpoint** | **Descrição** |
| GET | /pedidos | Lista todos os pedidos |
| GET | /pedidos/{id} | Retorna pedido por ID |
| POST | /pedidos | Cria novo pedido |
| PUT | /pedidos/{id} | Atualiza pedido existente |
| DELETE | /pedidos/{id} | Remove pedido por ID |

**| Exemplo de Requisição POST:**

**-----------------------------------------**

{

"clienteNome": "Vinicius",

"valorTotal": 450.24

}

**------------------------------------------**

**| Validações Implementadas:**

- “clienteNome”: obrigatório (não pode estar em branco)

- “valorTotal”: não pode ser negativo

- Validações feitas com anotações: @NotBlank, @PositiveOrZero

- Se enviar dados inválidos, a API responde com erro 400

**| Configuração do Banco (H2):**

spring.h2.console.enabled=true

spring.h2.console.path=/h2-console

spring.datasource.url=jdbc:h2:mem:pedidosdb

spring.datasource.username=sa

spring.jpa.hibernate.ddl-auto=update

**Link para acessar:**

<http://localhost:8080/h2-console>

**| Como Executar:**

1. Clone o repositório:

git clone https://github.com/[seu-usuario]/fiap-checkpoint1;

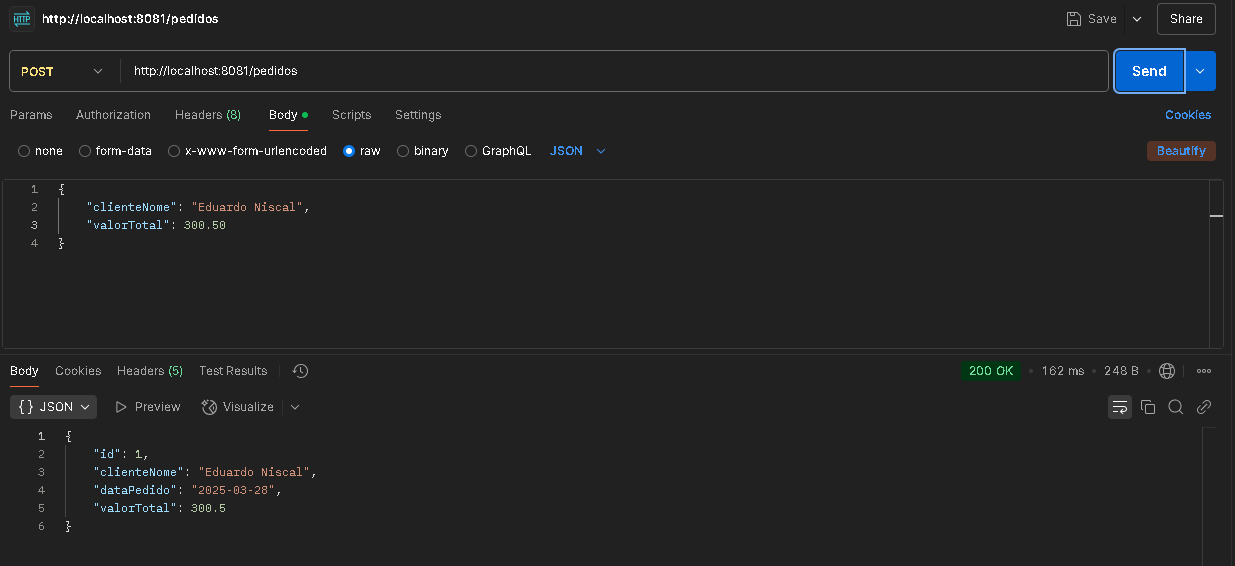
2. Abra o projeto na sua IDE (IntelliJ) e rode a classe Checkpoint1Application;

3. Abra o Postman e coloque na nova aba o link usado no postman e utilize o link ref ao H2 para entrar no H2 Console ciclando em connect (usando jdbc:h2:mem:pedidosdb na URL);

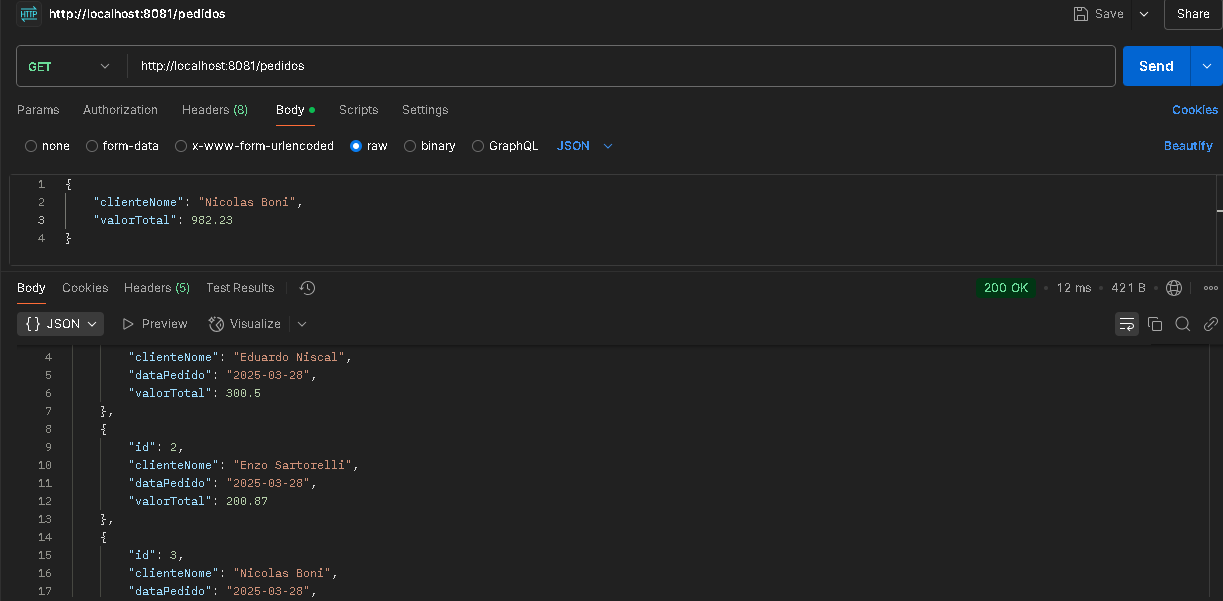
4. Teste os comandos usando o postman e coloque o comando “select \* from PEDIDO” no H2 para visualizar em tabela os pedidos.

**| Testes com Postman:**

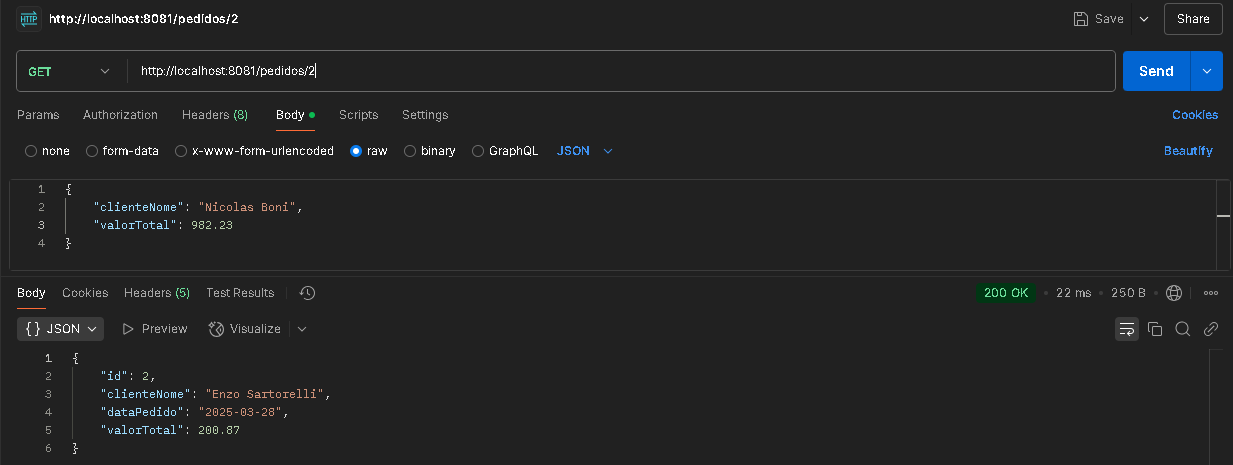
1. Criar novo pedido (POST)



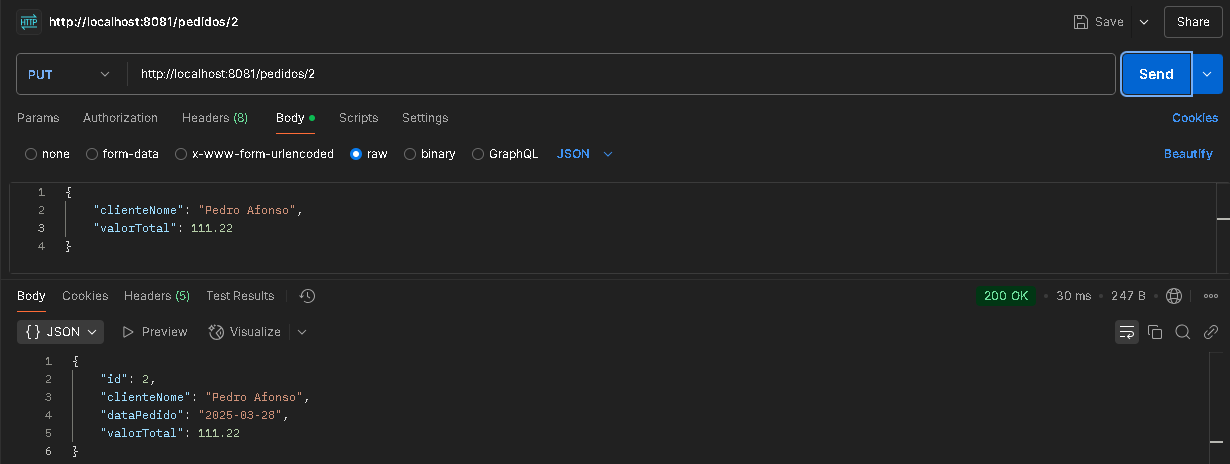
1. Listar todos os pedidos (GET)



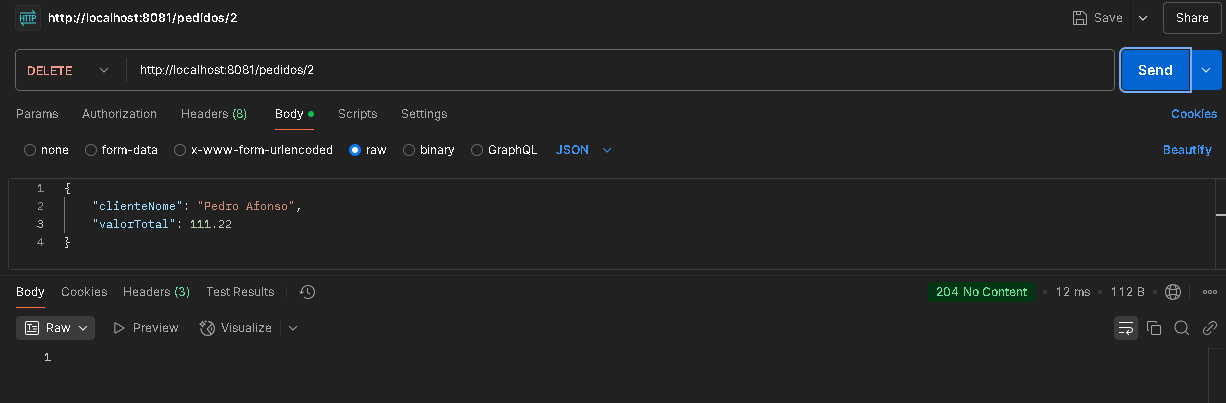
1. Buscar por ID (GET)



1. Atualizar pedido (PUT)



1. Deletar pedido (DELETE)



**| Exemplo de Table no H2 Console:**

