🎯 CONTEXTO DA PRÁTICA

🏢 Cenário Real: DeliveryTech em Crescimento

A startup **DeliveryTech** está evoluindo rapidamente! Após estabelecer a base do sistema com **Spring Boot** (Aula 3) e modelar as entidades **JPA** (Aula 4), chegou o momento crucial de implementar a **camada de acesso a dados e regras de negócio**.

✅ Situação Atual

- ✅ Ambiente configurado com **JDK 21**
- ✅ Projeto Spring Boot funcionando
- ✅ Entidades JPA modeladas
- ❌ Falta: Persistir dados no banco
- ❌ Falta: Implementar operações CRUD
- ❌ Falta: Regras de negócio organizadas

🚨 Problema Real Enfrentado

O CTO da DeliveryTech te chamou para uma reunião urgente:

💬 "**Pessoal, temos um problema! Os investidores querem ver o sistema funcionando com dados reais na próxima semana. Precisamos sair do 'Hello World' e implementar funcionalidades que realmente persistam informações no banco de dados. Nossos concorrentes já estão operando e não podemos ficar para trás!**"

🧩 Desafios Identificados

1. 👥 Clientes precisam ser cadastrados e consultados
2. 🍽️ Restaurantes devem ser gerenciados no sistema
3. 🛒 Produtos precisam ser organizados por restaurante
4. 📦 Pedidos devem ser criados e rastreados
5. 🧠 Regras de negócio precisam estar bem organizadas

🎮 Sua Missão como Desenvolvedor

Implementar a camada de persistência e a camada de serviços do sistema, criando uma base sólida para as funcionalidades de negócio.

📋 ATIVIDADES A SEREM DESENVOLVIDAS

⚙️ ATIVIDADE 1: CONFIGURAÇÃO DOS REPOSITORIES

⏱️ **Tempo estimado:** 25 minutos
🎯 **Objetivo:** Criar interfaces Repository com Spring Data JPA

✅ **Tarefas**

- Criar ClienteRepository:
    - Estender JpaRepository
    - Implementar buscas por e-mail e status ativo
- Criar RestauranteRepository:
    - Buscar por nome, categoria, ativos, ordenação por avaliação
- Criar ProdutoRepository:
    - Buscar produtos por restaurante, por categoria e disponibilidade
- Criar PedidoRepository:
    - Buscar pedidos por cliente, filtrar por status/data, relatórios

📦 **Entregáveis**

- Interfaces Repository criadas
- Métodos de consulta implementados
- Anotações @Query quando necessário
- Testes básicos funcionando

⚙️ **ATIVIDADE 2: IMPLEMENTAÇÃO DOS SERVICES**

⏱️ **Tempo estimado:** 35 minutos
🎯 **Objetivo:** Criar a camada de serviços com regras de negócio organizadas

✅ **Tarefas**

- ClienteService: Cadastro, validação de e-mail, busca, atualização, inativação
- RestauranteService: Gestão, validações, controle de status ativo/inativo
- ProdutoService: Cadastro por restaurante, disponibilidade, validação de preço
- PedidoService: Criação, cálculo de valores, mudança de status, validações

📦 **Entregáveis**

- Classes Service implementadas
- Injeção de dependência configurada
- Regras de negócio implementadas
- Tratamento básico de exceções

⚙️ **ATIVIDADE 3: CONTROLLERS REST**

⏱️ **Tempo estimado:** 30 minutos
🎯 **Objetivo:** Criar endpoints REST para exposição das funcionalidades

✅ **Tarefas**

- ClienteController:
    - POST /clientes, GET /clientes, GET /clientes/{id}, PUT /clientes/{id}, DELETE /clientes/{id}
- RestauranteController:
    - CRUD completo + buscar por categoria
- ProdutoController:
    - CRUD + buscar produtos por restaurante
- PedidoController:
    - Criar pedido, consultar por cliente, atualizar status

📦 **Entregáveis**

- Controllers REST implementados
- Endpoints funcionando
- Validações de entrada aplicadas
- Respostas HTTP adequadas

⚙️ **ATIVIDADE 4: TESTES E VALIDAÇÃO**

⏱️ **Tempo estimado:** 20 minutos
🎯 **Objetivo:** Testar todas as funcionalidades implementadas

✅ **Tarefas**

- Testes via **Postman** / **Insomnia**
- Verificação no console do banco **H2**
- Testes das regras de negócio (e-mail único, cálculos, status)

📦 **Entregáveis**

- Collection de testes Postman/Insomnia
- Screenshots dos testes
- Dados de exemplo no H2
- Funcionalidades validadas com sucesso

⚙️ **ATIVIDADE 5: DOCUMENTAÇÃO E FINALIZAÇÃO**

⏱️ **Tempo estimado:** 10 minutos
🎯 **Objetivo:** Documentar e organizar o projeto final

✅ **Tarefas**

- Atualizar README.md com:
    - Instruções de teste
    - Exemplos de uso
- Organizar commits e subir no GitHub
- Preparar aplicação para demonstração

📦 **Entregáveis**

- README.md atualizado
- Repositório GitHub organizado
- Aplicação demonstrável
- Documentação completa

📦 **ENTREGÁVEIS FINAIS**

✅ **O que deve ser entregue:**

1. **Repositório GitHub Atualizado**
    - Nome: delivery-api-[seunome]
    - Branch: main
    - Commits organizados por funcionalidade
    -
2. **Documentação Completa**
    - README.md com instruções, endpoints e exemplos
3.

📁 Estrutura de Pastas Esperada

```
delivery-api/
├── src/
│   ├── main/
│   │   ├── java/
│   │   │   └── com/deliverytech/delivery/
│   │   │       ├── controller/
│   │   │       │   ├── ClienteController.java
│   │   │       │   ├── RestauranteController.java
│   │   │       │   ├── ProdutoController.java
│   │   │       │   └── PedidoController.java
│   │   │       ├── service/
│   │   │       │   ├── ClienteService.java
│   │   │       │   ├── RestauranteService.java
│   │   │       │   ├── ProdutoService.java
│   │   │       │   └── PedidoService.java
│   │   │       ├── repository/
│   │   │       │   ├── ClienteRepository.java
│   │   │       │   ├── RestauranteRepository.java
│   │   │       │   ├── ProdutoRepository.java
│   │   │       │   └── PedidoRepository.java
│   │   │       ├── entity/
│   │   │       │   └── [entidades da aula anterior]
│   │   │       └── DeliveryApiApplication.java
│   │   └── resources/
│   │       ├── application.properties
│   │       └── data.sql
├── postman/
│   └── DeliveryAPI.postman_collection.json
├── README.md
└── pom.xml
```

🚀 DICAS IMPORTANTES

💡 Dicas de Implementação

- Comece pelos **Repositories**
- Mantenha **Services** com a lógica de negócio
- Teste frequentemente usando **H2 Console**
- Organize **commits descritivos** para cada funcionalidade

🔧 Recursos Disponíveis

- Console H2: http://localhost:8080/h2-console
- 📘 Documentação do Spring Data JPA
- 📎 Exemplos no GitHub do professor
- 📞 Suporte durante a aula

🧱 *"Grandes sistemas são construídos uma camada por vez. Hoje vocês estão construindo a fundação da persistência!"*