Intensivão Java Spring

Treinamento gratuito

- Crie um projeto para seu currículo

- Descubra o caminho para se tornar um desenvolvedor back end profissional

Aula 3

https://devsuperior.com.br

DEVSUPERIOR

Dr. Nelio Alves

Anteriormente

Aula 1:

- Conceitos
  - Sistemas web e recursos
  - Cliente/servidor, HTTP, JSON
  - Padrão Rest para API web
- Estruturação de projeto Spring Rest
- Entidades e ORM
- Database seeding
- Padrão camadas
- Controller, service, repository
- Padrão DTO

Aula 2:

- Relacionamentos N-N
- Classe de associação, embedded id
- Consultas SQL no Spring Data JPA
- Projections

Avisos

1. Perdeu algum episódio ou material de apoio?

Inscreva-se para receber no seu email:

https://devsuperior.com.br

ATENÇÃO: os conteúdos ficarão disponíveis somente até domingo. Então organize-se, e bora pra cima!

2. Tem alguma dúvida?

Envie uma mensagem pra gente no email que chegou pra você no ato da sua inscrição.

CALENDÁRIO

Os conteúdos ficarão temporariamente disponíveis no nosso canal de eventos. Ative o lembrete:

https://www.youtube.com/@DevsuperiorJavaSpring

Dia / horário	Conteúdo
Segunda-feira 20h30	Episódio 1: Projeto estruturado
Terça-feira 20h30	Episódio 2: Domínio, consultas
Quarta-feira 20h30	Episódio 3: Deploy CI/CD, CORS
Quinta-feira 20h30	Episódio 4: Endpoint especial
Sexta-feira 20h30	Episódio 5: Resumão e reforço do aprendizado

Projetos de portfólio

Github é um excelente material de currículo para os desenvolvedores

Dicas

1. Tenha projetos no seu Github e cite os principais no seu currículo.

2. O histórico é um indicativo de que você fez o projeto.

3. Não precisa decorar. Mas se for preciso, você deve saber fazer projetos similares (mesmo consultando).

4. No começo é natural reproduzir passo a passo. Porém, seu objetivo deve ser aprender a ser capaz de fazer por sua conta (mesmo consultando).

5. Crie um bom README no seu projeto. Por exemplo:

https://github.com/devsuperior/sds1-wmazoni

https://youtu.be/jIa8R69pKh8

Perfis de projeto

1. Perfil de desenvolvimento e testes:

- **test**
- Banco de dados H2

2. Perfil de homologação / staging:

- **dev**
- Banco de dados Postgres de homologação

3. Perfil de produção:

- **prod**
- Banco de dados Postgres de produção

Passos homologação

ATENÇÃO: OPCIONAL NO TREINAMENTO!!
Se você tiver encontrado dificuldades em instalar o Docker, ou mesmo o Postgres e pgAdmin direto no seu computador, pode apenas assistir essa parte como conhecimento, para entender como seria o processo de validação no Postgresql.

Preparação do ambiente

Docker
ou
Postgresql + pgAdmin ou DBeaver

Homologação local

1. Criar perfis de projeto
* system.properties

2. Gerar script da base de dados
* apagar arquivo gerado

3. Criar base de dados de homologação

4. Rodar app no modo dev e validar

Passos deploy CI/CD

ATENÇÃO: OPCIONAL NO TREINAMENTO!!

Os serviços de hospedagem de back end com banco de dados não estão mais oferecendo planos gratuitos. Assim, você pode apenas assistir essa parte a aula para conhecimento, para entender como funciona o processo de implantação. Neste momento, foque em deixar seu projeto caprichado no Github, com um bom Readme, conforme mostramos na aula.

Pré-requisitos

- Conta no Railway
- Conta no Github com mais de 90 dias
- Projeto Spring Boot salvo no seu Github
- Script SQL para criação e seed da base de dados
- Aplicativo de gestão de banco instalado (pgAdmin ou DBeaver)

Passos Railway

1. Prover um servidor de banco de dados

2. Criar a base de dados e seed

3. Criar uma aplicação Railway vinculada a um repositório Github

4. Configurar variáveis de ambiente

```
APP_PROFILE
DB_URL (Formato: jdbc:postgresql://host:porta/nomedabase)
DB_USERNAME
DB_PASSWORD
CORS_ORIGINS
```

5. Configurar o domínio público para a aplicação

6. Testar app no Postman

7. Testar a esteira de CI/CD

Revisão

O que você aprendeu:

- Dicas de currículo e portfólio
- Perfis de projeto
- Ambiente local com Docker Compose
- Processo de homologação local
- Processo de deploy com CI/CD
- Configuração de CORS