Agora é com você! Faça o mesmo procedimento que eu fiz na aula, porém, para a funcionalidade de **cadastro de pacientes**.

VER OPINIÃO DO INSTRUTOR

**Opinião do instrutor**

Você precisará criar a entidade Paciente:

@Getter

@EqualsAndHashCode(of = "id")

@NoArgsConstructor

@AllArgsConstructor

@Entity(name = "Paciente")

@Table(name = "pacientes")

**public** **class** **Paciente** {

@Id @GeneratedValue(strategy = GenerationType.IDENTITY)

**private** Long id;

**private** String nome;

**private** String email;

**private** String cpf;

**private** String telefone;

@Embedded

**private** Endereco endereco;

**public** Paciente(DadosCadastroPaciente dados) {

**this**.nome = dados.nome();

**this**.email = dados.email();

**this**.telefone = dados.telefone();

**this**.cpf = dados.cpf();

**this**.endereco = new Endereco(dados.endereco());

}

}

COPIAR CÓDIGO

Na sequência, precisará criar um repository:

**public** **interface** **PacienteRepository** **extends** **JpaRepository**<**Paciente**, **Long**> {

}

COPIAR CÓDIGO

Depois, precisará alterar as classes Controller e DTO:

@RestController

@RequestMapping("pacientes")

public class PacienteController {

@Autowired

private PacienteRepository repository;

@PostMapping

@Transactional

public void cadastrar(@RequestBody @Valid DadosCadastroPaciente dados) {

**repository**.save(new Paciente(dados));

}

}

COPIAR CÓDIGO

**public** **record** **DadosCadastroPaciente**(

@NotBlank String nome,

@NotBlank @Email String email,

@NotBlank String telefone,

@NotBlank @Pattern(regexp = "\\d{3}\\.?\\d{3}\\.?\\d{3}\\-?\\d{2}") String cpf,

@NotNull @Valid DadosEndereco endereco

) {

}

COPIAR CÓDIGO

E, por fim, vai precisar criar uma *migration* (**Atenção!** Lembre-se de parar o projeto antes de criar a migration!):

create **table** pacientes(

id bigint not null auto\_increment,

nome varchar(100) not null,

email varchar(100) not null unique,

cpf varchar(14) not null unique,

telefone varchar(20) not null,

logradouro varchar(100) not null,

bairro varchar(100) not null,

cep varchar(9) not null,

complemento varchar(100),

numero varchar(20),

uf char(2) not null,

cidade varchar(100) not null,

primary key(id)

);