

Código Refatorado

Classe Abstrata Pessoa

```
@Data
@AllArgsConstructor
@NoArgsConstructor
public abstract class Pessoa {

    private String nome;
    private String telefone;
    private String email;
    private LocalDate dataDeNascimento;

    // Método abstrato para validar informações específicas
    public abstract void validarInformacoes();
}
```

Classe PessoaFisica

```
@Data
@EqualsAndHashCode(callSuper=true)
@AllArgsConstructor
@NoArgsConstructor
public class PessoaFisica extends Pessoa {

    private String cpf;

    @Override
    public void validarInformacoes() {
        // Validação específica para CPF
        if (cpf == null || cpf.isEmpty()) {
            throw new IllegalArgumentException("CPF é obrigatório.");
        }
    }
}
```

Classe PessoaJuridica

```
@Data
@EqualsAndHashCode(callSuper=true)
@AllArgsConstructor
@NoArgsConstructor
public class PessoaJuridica extends Pessoa {

    private String cnpj;
    private String razaoSocial;

    @Override
    public void validarInformacoes() {
        // Validação específica para CNPJ
        if (cnpj == null || cnpj.isEmpty()) {
```

```

        throw new IllegalArgumentException("CNPJ é obrigatório.");
    }
}

```

Controller Refatorado

```

@RestController
@RequestMapping("/pessoa")
public class PessoaController {

    @Autowired
    private PessoaCore pessoaCore;

    // Endpoint para criar uma nova pessoa física
    @PostMapping("/pessoa-fisica/criar")
    public PessoaFisica criarPessoaFisica(@RequestBody PessoaFisica
pessoaFisica) {
        pessoaFisica.validarInformacoes();
        return pessoaCore.salvarPessoaFisica(pessoaFisica);
    }

    // Endpoint para criar uma nova pessoa jurídica
    @PostMapping("/pessoa-juridica/criar")
    public PessoaJuridica criarPessoaJuridica(@RequestBody PessoaJuridica
pessoaJuridica) {
        pessoaJuridica.validarInformacoes();
        return pessoaCore.salvarPessoaJuridica(pessoaJuridica);
    }
}

```

Serviço Core Refatorado

```

@Service
public class PessoaCore {

    // Método para salvar pessoa física
    public PessoaFisica salvarPessoaFisica(PessoaFisica pessoaFisica) {
        // Lógica para salvar pessoa física
        return pessoaFisica;
    }

    // Método para salvar pessoa jurídica
    public PessoaJuridica salvarPessoaJuridica(PessoaJuridica pessoaJuridica)
{
        // Lógica para salvar pessoa jurídica
        return pessoaJuridica;
    }
}

```