



EAD - Polo Santa Luiza – Vitória – ES

DESENVOLVIMENTO FULL STACK

RELATÓRIO DE PRÁTICA

MUNDO 3 – NÍVEL 1 – MISSÃO PRÁTICA

RPG0014 - Iniciando o caminho pelo Java

CARLOS ALTOMARE CATÃO

Semestre Letivo: 2025/1 - 3º Período

Data: 2025/04

PROCEDIMENTO 2

Objetivos da Prática:

1. Utilizar herança e polimorfismo na definição de entidades.
2. Utilizar persistência de objetos em arquivos binários.
3. Implementar uma interface cadastral em modo texto.
4. Utilizar o controle de exceções da plataforma Java.
5. No final do projeto, o aluno terá implementado um sistema cadastral em Java, utilizando os recursos da programação orientada a objetos e a persistência em arquivos binários.

Objetivos do Procedimento 2: Criação do Cadastro em Modo Texto

O objetivo do segundo procedimento é desenvolver uma aplicação Java de linha de comando capaz de realizar o **cadastro, alteração, exclusão e listagem** de pessoas físicas e jurídicas. O projeto deverá aplicar os princípios de programação orientada a objetos (POO), manipulação de arquivos binários para persistência de dados.

Códigos:

Os códigos relativos a esta prática se encontram em reservatório no GitHub e podem ser acessados pelo link: https://github.com/CarlosCatao/Mundo_3_Nivel_1_Missao_Pratica/tree/main/Procedimento_2/CadastroPOO.

Resultados:

Os resultados da execução dos códigos se encontram no arquivo PDF OUTPUT.pdf armazenado em repositório GitHub e pode ser acessado pelo link: https://github.com/CarlosCatao/Mundo_3_Nivel_1_Missao_Pratica/blob/main/Procedimento_2/OUTPUT.pdf.

Análise e Conclusão:

O que são elementos estáticos e qual o motivo para o método *main* adotar esse modificador?

Elementos estáticos pertencem à **classe**, e não a um objeto específico. O método *main* é o ponto de entrada da aplicação e é declarado como *static* porque ele precisa ser executado sem a criação de uma instância da classe principal.

Para que serve a classe *Scanner*?

A classe *Scanner* é usada para ler entradas do usuário, como textos ou números, diretamente do teclado (via *System.in*).

Como o uso de classes de repositório impactou na organização do código?

O uso de repositórios promoveu uma organização do código baseada na separação de responsabilidades, tornando o código mais limpo e modular. Cada tipo

de entidade (*Pessoa Física ou Jurídica*) tem seu próprio controle de dados, o que facilita a manutenção, a leitura e a escalabilidade do sistema.