

Iniciando O Caminho Pelo Java (CadastroPOO)

Lucca Ribeiro Polli Alves - 202208833811

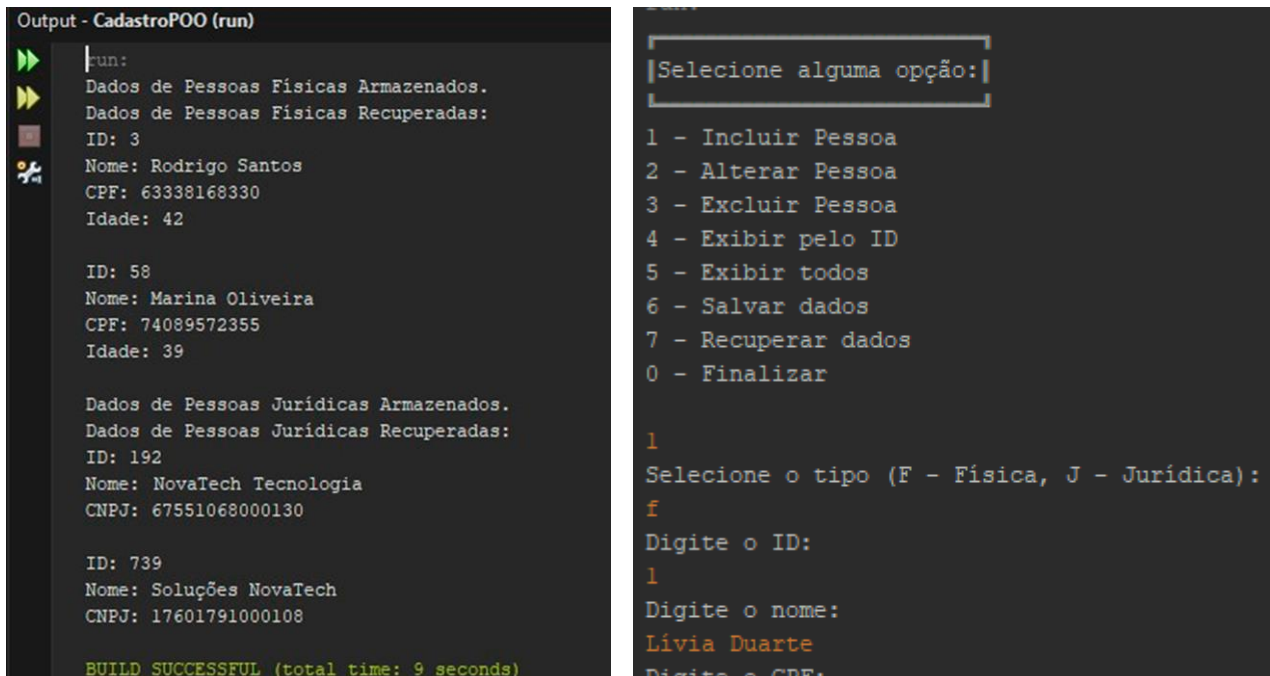
Campus: Nova América – Rio de Janeiro

Curso: Desenvolvimento full stack

Número da Turma: 9003

Semestre letivo: Mundo 3

Iniciando O Caminho Pelo Java (CadastroPOO)



```
Output - CadastroPOO (run)
run:
Dados de Pessoas Físicas Armazenados.
Dados de Pessoas Físicas Recuperadas:
ID: 3
Nome: Rodrigo Santos
CPF: 63338168330
Idade: 42

ID: 58
Nome: Marina Oliveira
CPF: 74089572355
Idade: 39

Dados de Pessoas Jurídicas Armazenados.
Dados de Pessoas Jurídicas Recuperadas:
ID: 192
Nome: NovaTech Tecnologia
CNPJ: 67551068000130

ID: 739
Nome: Soluções NovaTech
CNPJ: 17601791000108

BUILD SUCCESSFUL (total time: 9 seconds)

Selezione alguma opção:
1 - Incluir Pessoa
2 - Alterar Pessoa
3 - Excluir Pessoa
4 - Exibir pelo ID
5 - Exibir todos
6 - Salvar dados
7 - Recuperar dados
0 - Finalizar

1
Selezione o tipo (F - Física, J - Jurídica):
f
Digite o ID:
1
Digite o nome:
Livia Duarte
Digite o CPF:
```

Análise e Conclusão:

1º Procedimento | Criação das Entidades e Sistema de Persistência

1. Quais as vantagens e desvantagens do uso de herança?

- Vantagens incluem a reutilização de código, abstração e generalização, facilidade de manutenção e polimorfismo.
- Desvantagens englobam acoplamento, hierarquia complexa, fragilidade e aumento do acoplamento temporal.

2. Por que a interface Serializable é necessária ao efetuar persistência em arquivos binários?

- A interface Serializable é essencial para a persistência em arquivos binários, pois permite a conversão de objetos em bytes, preservando seu estado. Isso facilita a gravação e a recuperação de objetos, garantindo a persistência adequada.

3. Como o paradigma funcional é utilizado pela API stream no Java?

- A API Stream do Java emprega o paradigma funcional para manipular coleções de maneira concisa e eficiente. Utiliza pipelines de operações, compostas por operações intermediárias e terminais, utilizando expressões lambda e interfaces funcionais para definir operações de transformação e filtragem nos elementos da coleção.

4. Quando trabalhamos com Java, qual padrão de desenvolvimento é adotado na persistência de dados em arquivos?

- Ao desenvolver em Java, é comum adotar o padrão DAO (Data Access Object) para persistência de dados em arquivos. Esse padrão separa a lógica de acesso aos dados do restante do código, proporcionando modularidade, separação de preocupações e flexibilidade na manipulação de dados em arquivos.

2º Procedimento | Criação do Cadastro em Modo Texto

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

- Elementos estáticos referem-se a membros da classe compartilhados por todas as instâncias, como variáveis e métodos estáticos. O método "main" adota o modificador "static" porque é o ponto de entrada do programa, necessitando de acesso direto pela classe, sem a criação de instâncias, e sendo executado no contexto estático da classe.

2. Para que serve a classe Scanner?

- A classe Scanner em Java é utilizada para obter entrada do usuário, seja do teclado ou de arquivos. Ela oferece métodos para a leitura de diferentes tipos de dados, como inteiros, números de ponto flutuante e strings. Simplificando a interação com o usuário, a classe Scanner permite que programas recebam e processem informações inseridas durante a execução.

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

- A utilização de classes de repositório tem um impacto positivo na organização do código ao separar a camada de persistência da lógica de negócio. Isso resulta em um código mais limpo, coeso e reutilizável, concentrando as operações de acesso e manipulação de dados em um

único local. Essa abordagem facilita a substituição de mecanismos de armazenamento e promove boas práticas de programação, como o princípio da responsabilidade única, mantendo uma estrutura coesa e consistente.

Conclusão:

A abordagem de desenvolvimento em Java, empregando o padrão Ant e o NetBeans, é apropriada para criar um sistema de cadastro com persistência. A estruturação das entidades e gerenciadores é claramente definida, facilitando a manutenção do sistema. No entanto, a extensa utilização de herança para distinguir entre PessoaFisica e PessoaJuridica pode complexificar a evolução do sistema. Adicionalmente, a persistência em arquivos binários pode apresentar limitações em cenários mais complexos devido à falta de flexibilidade e escalabilidade. A falta de validações de entrada e tratamento robusto de exceções pode impactar a integridade do sistema.

