

**RELATÓRIO DISCENTE DE ACOMPANHAMENTO**

**Campus Polo Cavalhada – Porto Alegre**

**DESENVOLVIMENTO FULL STACK**

**Missão Prática | Nível 1 | Mundo 3**

**RPG0014 – Iniciando o caminho pelo Java**

**Turma: 9003**

**Semestre Letivo: 3º - 2024.1**

**GUILHERME BERNARDES BASTOS**

**Repositório:**

**CADASTROPOO**

**Objetivo da Prática**

Criar um sistema cadastral em Java, utilizando a programação orientada a objetos, herança, polimorfismo, persistência de objetos em arquivos binários, utilizar o controle de exceções e implementar uma interface cadastral em modo texto.

**1º Procedimento**

**CÓDIGOS**

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário

Descrição gerada automaticamente com confiança baixa

Uma imagem contendo Texto

Descrição gerada automaticamente

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Tabela

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Tabela

Descrição gerada automaticamente

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**RESULTADO**

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

**2º Procedimento**

**CÓDIGOS**

Tabela

Descrição gerada automaticamente

Interface gráfica do usuário

Descrição gerada automaticamente com confiança baixa

Uma imagem contendo Tabela

Descrição gerada automaticamente

Tabela

Descrição gerada automaticamente com confiança baixa

Interface gráfica do usuário

Descrição gerada automaticamente com confiança média

Uma imagem contendo Interface gráfica do usuário

Descrição gerada automaticamente

Tabela

Descrição gerada automaticamente com confiança baixa

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**RESULTADO**

**Tabela

Descrição gerada automaticamente**

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

**ANÁLISE E CONCLUSÃO**

**Procedimento 1**

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

Uma das vantagens mais significativas vejo que seria a reutilização do código em si, com a herança podemos criar classes que herdam métodos e atributos de outras, não precisando reescrever tudo novamente. Também permite alterar o comportamento de classes para adicionar novos métodos em uma subclasse. A manutenção do condigo, quando temos que alterar algo, com a herança, alteramos na classe base e assim todas as outras subclasses herdarão essas mudanças.

A principal desvantagem do uso de herança se dá na rigidez do código, no qual onde queremos alterar apenas um detalhe na classe, todas as outras subclasses serão alteradas e passiveis de erros. A complexidade de hierarquia pode se tornar difícil de entender e resultar em possíveis problemas.

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

Ela é necessária porque converte o código em uma sequência de bites em um arquivo para ser transmitido em rede.

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

Ela uso o paradigma funcional que permite operações de transformação e agregação de dados em coleções de uma forma mais declarativa.

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

O padrão mais comum usado é o DAO (Data Access Object).

**Procedimento 2**

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

São métodos ou variáveis que pertencem a própria classe ao invés de instâncias. O motivo do método main adotar esse modificador é que assim ele pode ser chamado diretamente para execução do programa sem a necessidade de depender de uma instância.

1. Para que serve a classe Scanner?

Serve para ler entradas no programa.

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

Ela impactou diretamente na manutenção do código, separando cada responsabilidade, reutilização de códigos e acesso a dados.