

Portfólio - Roteiro de aula prática

Linguagem Orientada a Objetos

Carlos Eduardo Sanches Sangregorio

RA: 3511167502

Portfólio- Roteiro de aula prática

Linguagem Orientada a Objetos

- Fixar os conceitos de programação orientada a objetos
- Empregar práticas de instanciação de classes em Java
- Utilizar métodos e atributos em Java
- Utilizar pacotes da linguagem Java para implementação de software com paradigma de orientação a objetos
- A IDE utilizada foi o NetBeans que é um software de desenvolvimento gratuito e de código aberto que pode ser baixado por esse link:
<https://netbeans.apache.org/front/main/download/nb21/>

SUMÁRIO

1. Atividade Proposta.....	04
2. Desenvolvimento.....	05
3. Execução do Código.....	09
4. Conclusão.....	10

1. Atividade Proposta:

Desenvolver uma pequena aplicação de gerenciamento bancário.

- A aplicação deve possibilitar ao usuário informar seu nome, sobrenome e CPF.
- Além disso, a aplicação deverá possibilitar ao usuário consultar saldo, realizar depósitos e saques.
- Esses procedimentos devem se repetir até que o usuário escolha encerrar o uso da aplicação.

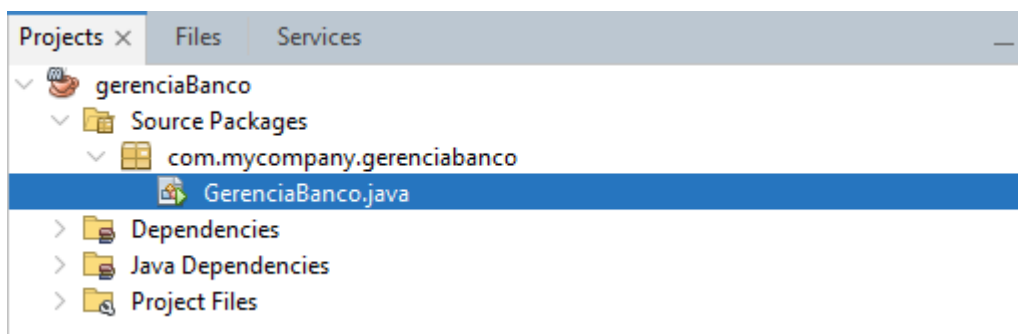
2. Desenvolvimento:

A primeira coisa a ser feita foi baixar o NetBeans e configurar o ambiente de desenvolvimento, que consistiu em baixar o JDK 16(Java Development Kit) pelo site oficial que é este:

<https://www.oracle.com/java/technologies/javase/jdk16-archive-downloads.html>. Depois disso, só foi necessário seguir o passo a passo e fazer o download do JDK normalmente. Para saber se foi tudo baixado devidamente, entre no CMD ou Prompt de comando do seu windows pela barra de pesquisa e digite no CMD “java -version”(sem as aspas), se aparecer como está na imagem a seguir quer dizer que foi instalado corretamente.

```
C:\Users\ >java -version
java version "16.0.2" 2021-07-20
Java(TM) SE Runtime Environment (build 16.0.2+7-67)
Java HotSpot(TM) 64-Bit Server VM (build 16.0.2+7-67, mixed mode, sharing)
```

Após ter certeza que tudo foi instalado corretamente foi a hora de abrir o NetBeans para começar o desenvolvimento do código, a criação do arquivo foi descrito no próprio roteiro então se você fez corretamente estará parecido com isso:



Depois de criar o arquivo com o nome “GerenciaBanco” comecei com o desenvolvimento do código.

Comecei colocando o comando “import java.util.Scanner;” para que quando o código for executado ele entenda os comandos que serão feitos pelo Cliente através do teclado. Após isso comecei a escrever a classe “Main” que está descrita como “public class GerenciaBanco” ela vai englobar a maior parte deste código, logo após uma parte importante que não se deve ser esquecida é a criação de um objeto da classe “Scanner” pois só aí será identificada a entrada do Cliente.

```
import java.util.Scanner;

public class GerenciaBanco {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
```

Após isso escrevi as linhas que vão armazenar as informações pessoais dos clientes em variáveis.

```
        System.out.println("Bem-vindo ao Gerenciamento Bancário!");
        System.out.print("Digite seu nome: ");
        String nome = scanner.nextLine();
        System.out.print("Digite seu sobrenome: ");
        String sobrenome = scanner.nextLine();
        System.out.print("Digite seu CPF: ");
        String cpf = scanner.nextLine();

        Cliente cliente = new Cliente(nome, sobrenome, cpf);
```

Depois de armazenar os dados em variáveis comecei a configurar o menu de ações do sistema, como: consultar saldo, realizar depósito, realizar saque ou simplesmente o encerramento.

```
public static void exibirMenu() {
    System.out.println("\n=== Menu ===");
    System.out.println("1. Consultar saldo");
    System.out.println("2. Realizar depósito");
    System.out.println("3. Realizar saque");
    System.out.println("4. Encerrar o programa");
    System.out.print("Escolha uma opção: ");
}
```

Após isso fiz a estrutura de decisão para que o Cliente pudesse escolher após ver o menu o que ele gostaria de fazer no programa.

```
int opcao;
do {
    exibirMenu();
    opcao = scanner.nextInt();

    switch (opcao) {
        case 1:
            System.out.println("Seu saldo atual é: R$" + cliente.consultarSaldo());
            break;
        case 2:
            System.out.print("Digite o valor do depósito: R$");
            double valorDeposito = scanner.nextDouble();
            cliente.deposito(valorDeposito);
            break;
        case 3:
            System.out.print("Digite o valor do saque: R$");
            double valorSaque = scanner.nextDouble();
            cliente.saque(valorSaque);
            break;
        case 4:
            System.out.println("Encerrando o programa...");
            break;
        default:
            System.out.println("Opção inválida. Por favor, escolha uma opção válida.");
    }
} while (opcao != 4);

scanner.close();
}
```

Apesar de parecer pronto com tudo em seu devido lugar, ainda falta a classe “Cliente” e as operações para quando o próprio Cliente quiser fazer um depósito ou até mesmo o saque e então eu escrevi a classe “Cliente” que é onde ficarão as variáveis que armazenam os dados pessoais e também escrevi as operações correspondentes às ações do menu.

```
class Cliente {
    private String nome;
    private String sobrenome;
    private String cpf;
    private double saldo;

    public Cliente(String nome, String sobrenome, String cpf) {
        this.nome = nome;
        this.sobrenome = sobrenome;
        this.cpf = cpf;
        this.saldo = 0.0;
    }

    public double consultarSaldo() {
        return saldo;
    }

    public void deposito(double valor) {
        saldo += valor;
        System.out.println("Depósito de R$" + valor + " realizado com sucesso.");
    }

    public void saque(double valor) {
        if (saldo >= valor) {
            saldo -= valor;
            System.out.println("Saque de R$" + valor + " realizado com sucesso.");
        } else {
            System.out.println("Saldo insuficiente para realizar o saque.");
        }
    }
}
```


3. Execução do código

Irei demonstrar o funcionamento do código com nome e CPF fictício:

```
Bem-vindo ao Gerenciamento Bancário!
Digite seu nome: João
Digite seu sobrenome: Campos
Digite seu CPF: 5325654870

=== Menu ===
1. Consultar saldo
2. Realizar depósito
3. Realizar saque
4. Encerrar o programa
Escolha uma opção: 1
Seu saldo atual é: R$0.0

=== Menu ===
1. Consultar saldo
2. Realizar depósito
3. Realizar saque
4. Encerrar o programa
Escolha uma opção: 2
Digite o valor do depósito: R$150
Depósito de R$150.0 realizado com sucesso.

=== Menu ===
1. Consultar saldo
2. Realizar depósito
3. Realizar saque
4. Encerrar o programa
Escolha uma opção: 3
Digite o valor do saque: R$100
Saque de R$100.0 realizado com sucesso.

=== Menu ===
1. Consultar saldo
2. Realizar depósito
3. Realizar saque
4. Encerrar o programa
Escolha uma opção: 1
Seu saldo atual é: R$50.0

=== Menu ===
1. Consultar saldo
2. Realizar depósito
3. Realizar saque
4. Encerrar o programa
Escolha uma opção: 4
Encerrando o programa...
```

4. Conclusão:

Com certeza foi um dos roteiros mais desafiadores, tive bastante dificuldade por nunca ter tido contato com Java e tive que quebrar a cabeça e estudar por diversos lugares para conseguir entender e construir esse pequeno sistema, porém foi gratificante quando tudo deu certo e consegui colocar todo o conhecimento em prática, funcionou como deveria e sem dúvidas foi um ótimo aprendizado para fixar os conceitos de classes em java e também o paradigma orientado a objetos.