

Atividade – Programação Orientada a Objetos

Main:

```
import java.util.ArrayList;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) throws Exception {

        ArrayList<Conta> contas = new ArrayList<Conta>();
        ArrayList<Integer> ordem = new ArrayList<Integer>();
        Scanner scan = new Scanner(System.in);
        int nroConta;
        String nomeTitular;
        String cpfTitular;
        System.out.println("\n<<Banco>>\n");
        while(true){
            System.out.println("\n1. Criar uma nova conta\n2 Exibir o
            saldo, ou extrato, de uma conta\n3 Sacar um valor de uma conta\n4
            Depositar um valor em uma conta\n5 Transferir valores de uma conta para
            outra\n0. Sair do programa");
            int i = scan.nextInt();
            if(i==0) break;
            if(i==1){ //Criar conta
                System.out.println("Digite o nro da conta: ");
                nroConta = scan.nextInt();
                System.out.println("Digite o nome do titular: ");
                nomeTitular = scan.next();
                System.out.println("Digite o cpf do titular: ");
                cpfTitular = scan.next();
                Conta conta = new Conta(nroConta, nomeTitular,
                cpfTitular);
                contas.add(conta);
                ordem.add(nroConta);
            } else
            if(i==2){ //Realizar extrato
                System.out.println("Digite o numero da conta no qual quer
                ver o extrato/saldo:");
                nroConta = scan.nextInt();
```

```

        int ContaOrdem = ordem.indexOf(nroConta);
        Conta contaRecuperada = (Conta) contas.get(ContaOrdem);
        contaRecuperada.extrato();
        /*System.out.println("Nro da conta:");
        System.out.println(contaRecuperada.getNroConta());
        System.out.println("Nome do titular:");
        System.out.println(contaRecuperada.getNomeTitular());
        System.out.println("CPF do titular:");
        System.out.println(contaRecuperada.getCpfTitular());
        System.out.println("Saldo:");
        System.out.println(contaRecuperada.getSaldo());*/
    } else
    if(i==3){ //Saque
        System.out.println("Digite o numero da conta no qual quer
realizar saque:");
        nroConta = scan.nextInt();

        int ContaOrdem = ordem.indexOf(nroConta);
        Conta contaRecuperada = (Conta) contas.get(ContaOrdem);

        System.out.println("Digite a quantidade a ser sacada:");
        Double valor = scan.nextDouble();
        if(contaRecuperada.sacar(valor)==true){
            System.out.println("Saque efetuado com sucesso!");
            //contaRecuperada.sacar(valor);
            contas.set(ContaOrdem,contaRecuperada);
        } else System.out.println("Saque não efetuado");
    } else
    if(i==4){ //Deposito
        System.out.println("Digite o numero da conta no qual quer
realizar deposito:");
        nroConta = scan.nextInt();

        int ContaOrdem = ordem.indexOf(nroConta);
        Conta contaRecuperada = (Conta) contas.get(ContaOrdem);

        System.out.println("Digite a quantidade a ser
depositada:");
        Double valor = scan.nextDouble();
        contaRecuperada.depositar(valor);
        System.out.println("Deposito efetuado com sucesso!");
        contas.set(ContaOrdem,contaRecuperada);
    } else
    if(i==5){
        System.out.println("Digite o numero da conta no qual quer
realizar transferência:");
        nroConta = scan.nextInt();
        System.out.println("Digite a quantidade a ser
transferida:");

```

```

        Double valor = scan.nextDouble();
        System.out.println("Digite o numero da conta que receberá
transferência:");
        int nroConta2 = scan.nextInt();

        int ContaOrdem = ordem.indexOf(nroConta);
        Conta contaRecuperada = (Conta) contas.get(ContaOrdem);

        ContaOrdem = ordem.indexOf(nroConta2);
        Conta contaRecuperada2 = (Conta) contas.get(ContaOrdem);

        if(contaRecuperada.transferir(valor,
contas.get(ContaOrdem))){
            System.out.println("Transferência realizada com
sucesso!");
        } else System.out.println("Transferência não efetuada");

    }
}
}
}
}

```

Conta:

```

public class Conta implements Comparable<Conta>{
    private int nroConta;
    private int agencia;
    private String nomeTitular;
    private String cpfTitular;
    //String nomeBanco;
    //int nroBanco;
    private double saldo;

    public Conta(int nroConta, int agencia, String nomeTitular, String
cpfTitular){
        this.nroConta = nroConta;
        this.agencia = agencia;
        this.nomeTitular = nomeTitular;
        this.cpfTitular = cpfTitular;
        this.saldo = 0.0;
    }

    public Conta(int nroConta, String nomeTitular, String cpfTitular){
        this.nroConta = nroConta;
    }
}

```

```
this.nomeTitular = nomeTitular;
this.cpfTitular = cpfTitular;

this.agencia = 123;
this.saldo = 0.0;
}

public double getSaldo(){
    return this.saldo;
}

public int getNroConta() {
    return this.nroConta;
}

public void setNroConta(int nroConta) {
    this.nroConta = nroConta;
}

public int getAgencia() {
    return this.agencia;
}

public void setAgencia(int agencia) {
    this.agencia = agencia;
}

public String getNomeTitular() {
    return this.nomeTitular;
}

public void setNomeTitular(String nomeTitular) {
    this.nomeTitular = nomeTitular;
}

public String getCpfTitular() {
    return this.cpfTitular;
}

public void setCpfTitular(String cpfTitular) {
    this.cpfTitular = cpfTitular;
}

public boolean depositar(double valor){
    if(valor > 0){
        this.saldo += valor;
        return true;
    }
}
```

```

    }
    return false;
}

public boolean sacar(double valor){
    if(valor > 0 && valor <= this.saldo){
        this.saldo -= valor;
        return true;
    }
    return false;
}

public boolean transferir(double valor, Conta conta){
    if(this.sacar(valor)){
        conta.depositar(valor);
        return true;
    }
    return false;
}

public void extrato(){//Totalmente errado fazer isso!
    System.out.println("<<<<< Dados da Conta >>>>>");
    System.out.println("Nome titular: "+this.nomeTitular);
    System.out.println("Número conta: "+this.nroConta);
    System.out.println("Saldo R$"+getSaldo());
    System.out.println("-----\n\n");
}

@Override
public int compareTo(Conta c) {
    if(this.nroConta > c.getNroConta()) return 1;
    if(this.nroConta < c.getNroConta()) return -1;
    return 0;
}

@Override
public boolean equals(Object obj) {

    if(obj.getClass() == Conta.class){
        Conta c = (Conta) obj;
        return this.nroConta == c.getNroConta();
    }
    return false;
}
}

```

