Aula do dia 26/05/2020 – Ouro 9

Polimorfismo, outro exercício

Instruções:

1. Fazer individualmente

2. Não copiar do colega, pois o objetivo é aprender. Se houver cópia os dois serão zerados. Pode estudar em conjunto. Mas depois cada um faz o seu exercício.

Pontuação: 5 pontos.

Enunciado – A classe Conta é a classe mãe das classes ContaCorrente e ContaPoupanca. Todas estão dentro do pacote “conta”. Fazer uma classe para testar o programa e coloca-la em outro pacote a sua escolha.

A classe Conta possui como atributos:

número da conta,

titular da conta e

saldo da conta.

Crie os setters e os getters para os atributos da classe Conta bem como o construtor “default” e pelo menos um construtor para inicializar os atributos com valores a serem entrados pelo usuário. Além disso crie os métodos **saque** e **deposite** com os seguintes algoritmos:

public boolean saque(double valor){

boolean temSaldo = false;

if(valor > this.saldo)

temSaldo = false;

else

{

temSaldo = true;

this.saldo -= valor;

}

return temSaldo;

}

public void deposite(double valor){

this.saldo += valor;

}

Em seguida, estenda a classe Conta criando as classes ContaCorrente e ContaPoupanca no pacote “conta”.

O que a ContaCorrente tem de especial em relação à classe Conta? Ela possui um limite especial de crédito. Este atributo possui o get e o set respectivos. Quando o titular quiser sacar um valor da conta, o método saque acima faz uma operação de saque da Conta cuja lógica é verificar se o valor a ser sacado é maior do que o valor do saldo da conta ou não. Já a lógica para sacar na classe ContaCorrente é outra: se o valor a ser sacado for maior do que o valor do saldo mais o valor do limite especial de crédito da conta o saque não é realizado. Caso contrário o saque é realizado e o saldo atualizado. O método saque da ContaCorrente também retorna true ou false.

Em outras palavras, se o limite especial de crédito é de R$50,00 e o saldo atual da conta corrente é de R$150,00 se o usuário desejar sacar até R$200,00 ele conseguirá, pois 200,00 <= 50 + 150. Caso o titular deseje sacar R$201,00 não conseguirá pois o R$201,00 > 50 + 150.

Em ambos os casos, os dois métodos deverão retornar verdadeiro se o saque foi possível e falso caso o saque não seja possível.

a) Como se chama este procedimento de se criar dois métodos, um na mãe e outro no filho, com a mesma assinatura porém com os algoritmos internos diferentes?  
**R: Override ou reescrita de métodos.**

O banco tem que tomar algumas atitudes com relação aos saques, saldos e limites para sempre disponibilizar limites compatíveis com os saldos médios dos clientes em suas contas correntes.

Para implementar esta política o banco mandou os analistas criarem um método que deverá contabilizar os saldos de toda conta a cada depósito ou saque quer seja na conta corrente ou na conta poupança.

Como esta operação não é própria nem da conta corrente e nem da conta poupança, o analista deverá criar uma nova classe denominada Movimento com um atributo chamado totalDeSaldoDoCliente e um método denominado registraMovimento, o qual deverá ser usado na main toda vez que um titular de alguma conta fizer um saque/depósito quer na conta corrente ou na conta poupança. Seu algoritmo é simples: se houver um saque ele faz uma subtração do totalDeSaldoDoCliente. Se houver um depósito ele faz uma adição neste total. Ele deve fazer esta operação tanto para operações na conta corrente quanto na conta poupança.

Lembrar que já na inicialização dos objetos se for atribuído um saldo a uma conta este saldo já deve entrar como movimentação também.

b) Explicar por que o método registraMovimento é polimórfico.  
**R: Porque o métodos registraMovimentos pode receber diferentes objetos e manipular seus métodos conforme a aplicação desejada.**

Testar este sistema com os dados de um único cliente instanciando um objeto do tipo ContaCorrente com os dados :

Conta 1: conta corrente.

Número: 100.

Titular: Fabio Silva.

Saldo: 500,00.

E, outro objeto do tipo ContaPoupança com os seguintes dados:

Conta 2: conta poupança.

Número: 200.

Titular: Fabio Silva.

Saldo: 300,00.

Depois de instanciados, fazer a seguinte sequência de operações:

i. operação 1 🡪 sacar R$100,00 da conta 1.

ii. operação 2 🡪 sacar R$150,00 da conta 2.

iii. Imprimir o saldo total do cliente Fabio Silva.

iv. Depositar R$20,00 na conta 1.

v. Depositar R$80,00 na conta 2.

vi. Depositar mais R$130,00 na conta 2.

vii. Sacar R$30,00 na conta 1.

Em seguida criar na mãe o método transferir dinheiro de uma conta para outra. O primeiro argumento deste método de transferência é o valor a ser transferido e o segundo é a referencia para o objeto destino. Lembrar que você já fez isto num exercício anterior.

Para o software ter coerência e clareza, a cada transferência incluir dois movimentos um de subtração no objeto origem e a outra de adição no objeto destino.

viii. Transferir R$65,00 da poupança para a conta corrente.

ix. Transferir R$200,00 da conta corrente para a poupança.

Imprimir no final o total de saldo do cliente.

Montar o projeto. Gerar o javadoc do projeto. Gerar o jar do projeto.

Compilar, executar e fazer todos os testes.

No final, compacte o diretório do projeto e anexe junto este arquivo Word com as respostas às perguntas feitas anteriormente. Em seguida, faça o upload na página da disciplina no faionline.