**Composição de Classes**

***Mário Leite***

...

### Alguns conceitos que existem na Programação Orientada a Objeto (OO) são bastante conhecidos dos programadores, como: Abstração, Herança, Encapsulamento, Polimorfismo, Sobrecarga, etc; mas a Composição, nem tanto. Este conceito é muito importante quando se trabalha com classes e é fundamental; onde uma classe contém instâncias de outras classes como parte de seus atributos. Essa relação entre classes é uma forma do tipo "*tem-um*" (temporalmente) ou "*é-feito-de*" (no caso de um *todo* formado por partes). Em outras palavras: uma classe composta tem objetos de outras classes como membros (atributos), e esses objetos são criados e destruídos junto com a classe que os contém. Na Composição uma classe não pode existir sem a outra; isto significa que a classe composta e a classe que ela contém possui uma relação de vida útil atrelada, ou seja, a instância da classe contida é criada e destruída automaticamente quando a classe composta é criada ou destruída.

O programa **“Composição”**, codificado em C#, mostraclasse a “ClsCliente” querepresenta um cliente com *nome* e *CPF*. A classe “ClsHistorico” representa o histórico de transações de uma conta com a data de abertura e uma lista de transações. A classe “ClsConta” recebe um nome de cliente com o histórico de sua conta e as transações são registradas no histórico sempre que um depósito/saque é realizado, fazendo uma composição. Neste caso, são criadas três classes: “ClsCliente”, “ClsHistorico” e “ClsConta” para representar *clientes*, *histórico de transações* e *contas*, respectivamente. O programa principal cria uma instância de “ClsCliente” e uma de “ClsConta”, realiza algumas transações e exibe o saldo e o histórico de transações da conta. A **figura 1** mostra a saída do programa.

### -------------------------------------------------------------------------------------------------------------------------------------

### 

### 

### Figura 1 - Saída do programa “Composição”

**using** System;

**using** System.Collections.Generic;

**namespace** Composição

{

**class** **ClsCliente**

{

**public** **string** NomeCli { **get**; **set**; }

**public** **string** CpfCli { **get**; **set**; }

**public** **ClsCliente**(**string** nomeCli, **string** cpfCli)

{

NomeCli = nomeCli;

CpfCli = cpfCli;

}

**public** **void** **ExibirCliente**()

{

**Console**.WriteLine($"Cliente: **{NomeCli}** - CPF: **{CpfCli}**");

}

}

//----------------------------------------------------------------------------

**class** **ClsHistorico**

{

**public** **DateTime** DataAbertura { **get**; **set**; }

**public** **List**<Transacao> LstTransacoes { **get**; **set**; }

**public** **ClsHistorico**(**DateTime** dataAbertura)

{

DataAbertura = dataAbertura;

LstTransacoes = **new** **List**<Transacao>();

}

**public** **void** **AdicionarTransacao**(**string** descricao, **double** valor)

{

LstTransacoes.Add(**new** Transacao

{

Descricao = descricao,

Valor = valor,

Data = **DateTime**.Now});

}

**public** **void** **ExibirTransacoes**()

{

**foreach** (**var** transacao **in** LstTransacoes)

{

**string** dataFormatada = transacao.Data.ToString("dd/MM/yyyy");

**Console**.WriteLine($"**{dataFormatada}** - **{transacao.Descricao}** –

R$**{transacao.Valor:F2}**");

}

}

}

//----------------------------------------------------------------------------

**class** **Transacao**

{

**public** **string** Descricao { **get**; **set**; }

**public** **double** Valor { **get**; **set**; }

**public** **DateTime** Data { **get**; **set**; }

}

**class** **ClsConta**

{

**public** **int** NumeroConta { **get**; **set**; }

**public** **ClsCliente** Cliente { **get**; **set**; }

**public** **double** SaldoConta { **get**; **set**; }

**public** **ClsHistorico** HistoricoConta { **get**; **set**; }

**public** **ClsConta**(**int** numeroConta, **ClsCliente** cliente, **double** saldoConta = 0)

{

NumeroConta = numeroConta;

Cliente = cliente;

SaldoConta = saldoConta;

HistoricoConta = **new** **ClsHistorico**(**DateTime**.Now);

}

**public** **void** **Depositar**(**double** valor)

{

SaldoConta += valor;

HistoricoConta.**AdicionarTransacao**("Depósito", valor);

}

**public** **void** **Sacar**(**double** valor)

{

**if** (valor <= SaldoConta)

{

SaldoConta -= valor;

HistoricoConta.**AdicionarTransacao**("Saque", valor);

}

**else**

{

**Console**.WriteLine("Saldo insuficiente!");

}

}

**public** **void** **ExibirSaldo**()

{

**Console**.WriteLine($"Saldo da conta **{NumeroConta}**: R$**{SaldoConta:F2}**");

}

**public** **void** **ExibirCliente**()

{

Cliente.**ExibirCliente**();

}

**public** **void** **ExibirHistorico**()

{

**Console**.WriteLine("\nHistórico de transações:");

HistoricoConta.**ExibirTransacoes**();

}

}

//============================================================================

//Programa principal

**class** **Program**

{

**static** **void** **Main**()

{

// Cria instâncias

**ClsCliente** cliente1 = **new** **ClsCliente**("João", "123.456.789-00");

**ClsConta** conta1 = **new** **ClsConta**(1, cliente1, 1000);

// Faz transações

conta1.**ExibirCliente**();

conta1.**ExibirSaldo**();

conta1.**Depositar**(500);

conta1.**Sacar**(200);

// Após as transações

conta1.**ExibirSaldo**();

conta1.**ExibirHistorico**();

**Console**.ReadKey();

}

}

}**//Fim do programa "Composição" --------------------------------------------------**