

## Laboratório 4 – Definindo Classes

Este laboratório introduz o processo de criação de classes em C#.

### 1 Descrevendo o processo de criação de classes e descrição de atributos e métodos

1. Crie um projeto “Console Application” com nome “Laboratorio4”.
2. Dentro da solução adicione um novo arquivo “ContaCorrente.cs” e declare a seguinte classe:

```
using System;
```

```
class ContaCorrente {  
  
}
```

3. Acrescente um atributo saldo do tipo decimal para a conta, logo após o abre-chaves da definição da classe:

```
private decimal saldo;
```

4. Acrescente um método para depósito de valores na conta:

```
public void Depositar(decimal val)  
{  
    saldo = saldo + val;  
}
```

5. Acrescente um método para saque de valores da conta:

```
public void Sacar(decimal val)  
{  
    saldo = saldo - val;  
}
```

6. Encontre o método Main e adicione o seguinte código:

```
ContaCorrente minhaConta = new ContaCorrente();  
minhaConta.Depositar(100);  
minhaConta.Sacar(50);
```

7. Compile o programa.

### 2 Adicionando propriedade de leitura

1. Dentro da definição da classe *ContaCorrente*, digite o código:

```
public decimal Saldo  
{  
    get { return saldo; }  
}
```

2. Dentro do método Main (apague seu conteúdo anterior), digite o seguinte código:

```
ContaCorrente minhaConta = new ContaCorrente();  
Console.WriteLine(minhaConta.Saldo);  
minhaConta.Depositar(100);  
Console.WriteLine(minhaConta.Saldo);
```

```
minhaConta.Sacar(50);
Console.WriteLine(minhaConta.Saldo);
```

3. Compile e execute o programa.

### 3 Definindo um método construtor para inicializar os atributos de uma classe

1. Dentro da definição da classe *ContaCorrente* digite o código:

```
public ContaCorrente(decimal val)
{
    saldo = val;
}
```

2. Dentro do método Main (apague seu conteúdo anterior), digite o seguinte código:

```
ContaCorrente minhaConta = new ContaCorrente(340);
Console.WriteLine(minhaConta.Saldo);
minhaConta.Depositar(100);
Console.WriteLine(minhaConta.Saldo);
minhaConta.Sacar(50);
Console.WriteLine(minhaConta.Saldo);
```

3. Compile e execute o programa.

### 4 Exercícios

1. Altere a classe *ContaCorrente* de forma que ela possua atributos para armazenar a data de criação da conta e o nome do titular. O nome do titular deve ser informado como parâmetro do construtor. A data de criação deve ser gerada no momento da inicialização (*DateTime.Now*). Crie também propriedades de acesso a estes atributos.

2. Altere a classe *ContaCorrente* de maneira que ela seja capaz de armazenar o saldo médio do correntista. Para tanto, cada vez que os métodos de depósito ou retirada forem acionados o saldo da conta, depois da operação, deve ser acumulado e um contador deve ser incrementado. O saldo médio é obtido dividindo-se este acumulador pelo número de operações realizadas.

3. Escreva um programa que instancia uma conta corrente, executa uma série de operações de depósito e retirada e, por fim, imprime o saldo da conta.