

## **Gabarito – Livro: Programação de Computadores – C#**

**Prezado(a) Educador(a)**

**O material ora apresentado caracteriza-se em ser formado pelas respostas dos exercícios de fixação do livro em referência.**

**As respostas aqui dadas são fornecidas genericamente no sentido de atender a solução de certo problema do ponto de vista algorítmico e servirá como ponto de auxílio ao seu trabalho.**

**As respostas aqui indicadas são respostas propostas sem a pretensão de serem as únicas ou as melhores respostas.**

**As respostas dadas são respostas possíveis.**

**Quando em uso interface gráfica o código apresentado será relacionado ao botão da aplicação.**

**O conjunto de exercícios aqui apresentados é em média suficiente para a fixação dos detalhes sobre o uso básico e inicial da linguagem de programação de computadores em foco.**

**Espero com isso estar fornecendo uma maior facilidade para o(a) colega poder ministrar a base de conhecimento sobre programação de computadores.**

**Atenciosamente**

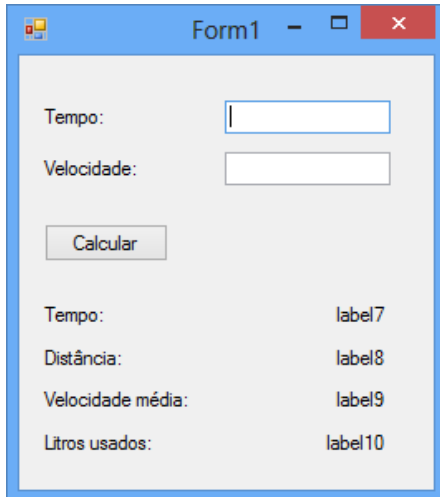
**José Augusto N. G. Manzano**

---

## Capítulo 2

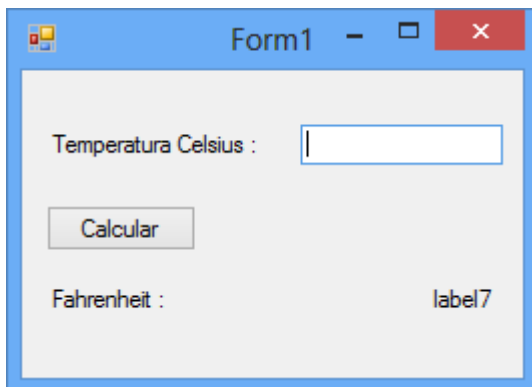
---

### Exercício 1



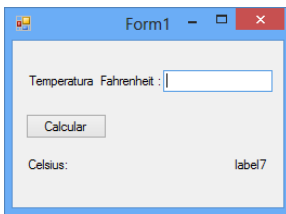
```
private void button1_Click(object sender, EventArgs e)
{
    float TEMPO, VELOCIDADE, DISTANCIA, LITROS_USADOS;
    TEMPO = Convert.ToSingle(textBox1.Text);
    VELOCIDADE = Convert.ToSingle(textBox2.Text);
    DISTANCIA = TEMPO * VELOCIDADE;
    LITROS_USADOS = DISTANCIA / 12;
    label7.Text = Convert.ToString(TEMPO.ToString("##,##0.00"));
    label8.Text = Convert.ToString(DISTANCIA.ToString("##,##0.00"));
    label9.Text = Convert.ToString(VELOCIDADE.ToString("##,##0.00"));
    label10.Text = Convert.ToString(LITROS_USADOS.ToString("##,##0.00"));
}
```

### Exercício 2



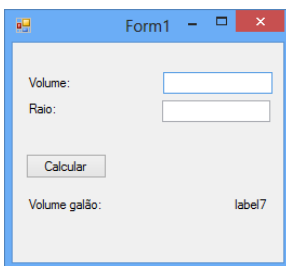
```
private void button1_Click(object sender, EventArgs e)
{
    float C, F;
    C = Convert.ToSingle(textBox1.Text);
    F = (9 * C + 160) / 5;
    label7.Text = Convert.ToString(F.ToString("##,##0.00"));
}
```

## Exercício 3



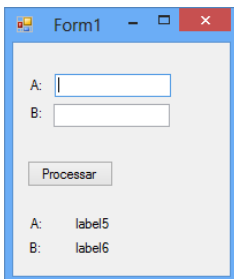
```
private void button1_Click(object sender, EventArgs e)
{
    float C, F;
    F = Convert.ToSingle(textBox1.Text);
    C = ((F - 32) * 5) / 9;
    label7.Text = Convert.ToString(C.ToString("##,##0.00"));
}
```

## Exercício 4



```
private void button1_Click(object sender, EventArgs e)
{
    double V, R, H;
    H = Convert.ToSingle(textBox1.Text);
    R = Convert.ToSingle(textBox2.Text);
    V = 3.14159 * R * R * H;
    label7.Text = Convert.ToString(V.ToString("##,##0.00"));
}
```

## Exercício 5



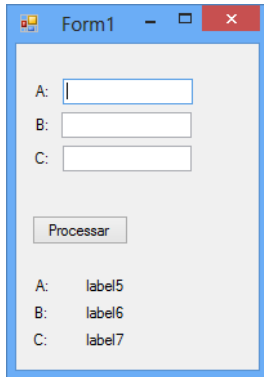
```
private void button1_Click(object sender, EventArgs e)
{
    int A, B, X;
    A = Convert.ToInt32(textBox1.Text);
    B = Convert.ToInt32(textBox2.Text);
    X = A;
    A = B;
    B = X;
    label5.Text = Convert.ToString(A);
    label6.Text = Convert.ToString(B);
}
```

---

## Capítulo 3

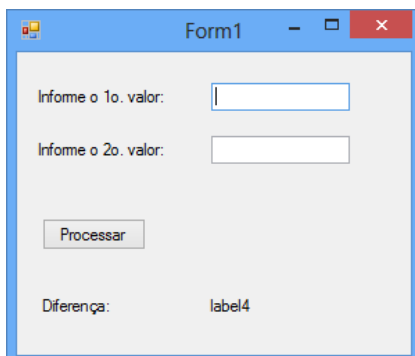
---

### Exercício 1



```
private void button1_Click(object sender, EventArgs e)
{
    int A, B, C, X;
    A = Convert.ToInt32(textBox1.Text);
    B = Convert.ToInt32(textBox2.Text);
    C = Convert.ToInt32(textBox3.Text);
    if (A > B)
    {
        X = A;
        A = B;
        B = X;
    }
    if (A > C)
    {
        X = A;
        A = C;
        C = X;
    }
    if (B > C)
    {
        X = B;
        B = C;
        C = X;
    }
    label5.Text = Convert.ToString(A);
    label6.Text = Convert.ToString(B);
    label7.Text = Convert.ToString(C);
}
```

### Exercício 2



```

private void button1_Click(object sender, EventArgs e)
{
    int V1, V2, DIF;
    V1 = Convert.ToInt32(textBox1.Text);
    V2 = Convert.ToInt32(textBox2.Text);
    if (V1 > V2)
    {
        DIF = V1 - V2;
    }
    else
    {
        DIF = V2 - V1;
    }
    label4.Text = Convert.ToString(DIF);
}

```

### Exercício 3

```

private void button1_Click(object sender, EventArgs e)
{
    int V1, V2, V3, V4, R2, R3;
    V1 = Convert.ToInt32(textBox1.Text);
    V2 = Convert.ToInt32(textBox2.Text);
    V3 = Convert.ToInt32(textBox3.Text);
    V4 = Convert.ToInt32(textBox4.Text);
    label5.Text = "";
    R2 = V1 % 2;
    R3 = V1 % 3;
    if (R2 == 0 && R3 == 0)
    {
        label5.Text += Convert.ToString(V1) + " ";
    }
    R2 = V2 % 2;
    R3 = V2 % 3;
    if (R2 == 0 && R3 == 0)
    {
        label5.Text += Convert.ToString(V2) + " ";
    }
    R2 = V3 % 2;
    R3 = V3 % 3;
    if (R2 == 0 && R3 == 0)
    {
        label5.Text += Convert.ToString(V3) + " ";
    }
    R2 = V4 % 2;
    R3 = V4 % 3;
    if (R2 == 0 && R3 == 0)
    {
        label5.Text += Convert.ToString(V4) + " ";
    }
}

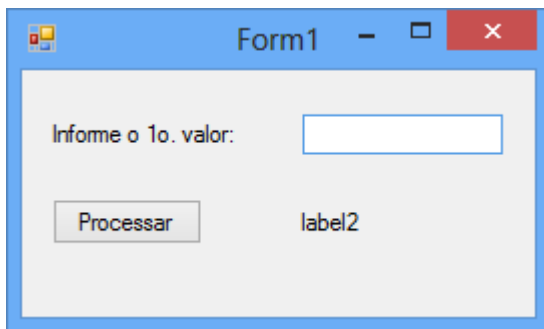
```

```

    {
        label5.Text += Convert.ToString(V3) + " ";
    }
    R2 = V4 % 2;
    R3 = V4 % 3;
    if (R2 == 0 && R3 == 0)
    {
        label5.Text += Convert.ToString(V4) + " ";
    }
}

```

## Exercício 4



```

private void button1_Click(object sender, EventArgs e)
{
    int V, R;
    V = Convert.ToInt32(textBox1.Text);
    R = V % 2;
    if (R == 0)
    {
        label2.Text = "Valor par";
    }
    else
    {
        label2.Text = "Valor impar";
    }
}

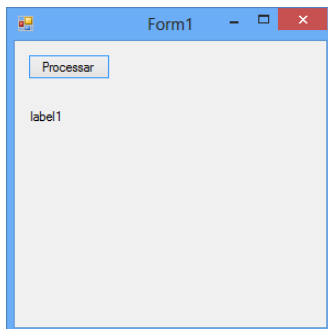
```

---

## Capítulo 4

---

### Exercício 1A - Laço: while



```

private void button1_Click(object sender, EventArgs e)
{

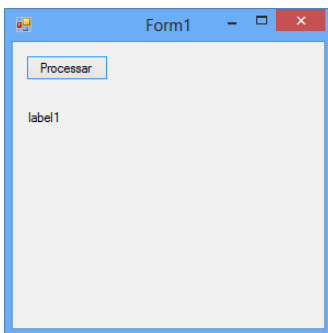
```

```

        int I = 0, R;
        label1.Text = "";
        while (I <= 20)
        {
            R = I % 2;
            if (R != 0)
            {
                label1.Text += Convert.ToString(I) + " ";
            }
            I++;
        }
    }
}

```

## Exercício 2A - Laço: do...while

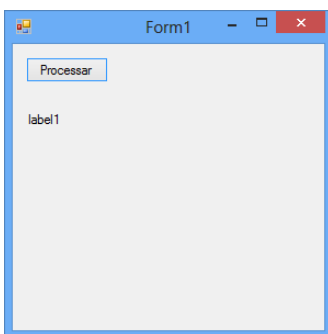


```

private void button1_Click(object sender, EventArgs e)
{
    int I = 0, R;
    label1.Text = "";
    do
    {
        R = I % 2;
        if (R != 0)
        {
            label1.Text += Convert.ToString(I) + " ";
        }
        I++;
    }
    while (I <= 20);
}

```

## Exercício 3A - Laço: for



```

private void button1_Click(object sender, EventArgs e)
{
    int I, R;
    label1.Text = "";
    for (I = 0; I <= 20; I++ )
    {
        R = I % 2;
        if (R != 0)

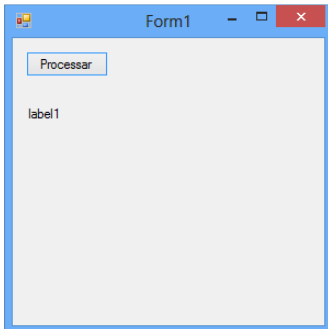
```

```

        {
            label1.Text += Convert.ToString(I) + " ";
        }
    }
}

```

## Exercício 1B - Laço: while

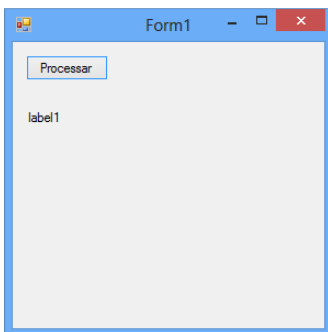


```

private void button1_Click(object sender, EventArgs e)
{
    int I, S = 0;
    label1.Text = "";
    I = 1;
    do
    {
        S += I;
        I++;
    }
    while (I <= 100);
    label1.Text = Convert.ToString(S);
}

```

## Exercício 2B - Laço: do...while



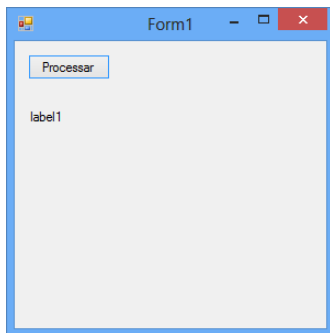
```

private void button1_Click(object sender, EventArgs e)
{
    int I, S = 0;
    label1.Text = "";
    I = 1;
    while (I <= 100)
    {
        S += I;
        I++;
    }
    label1.Text = Convert.ToString(S);
}

```

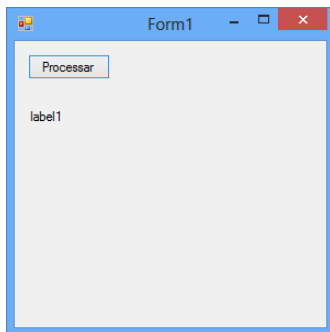
## Exercício 3B - Laço: for





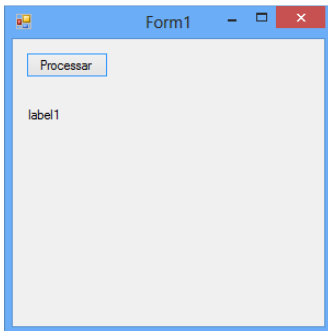
```
private void button1_Click(object sender, EventArgs e)
{
    int I, S = 0;
    label1.Text = "";
    for (I = 1; I <= 100; I++ )
    {
        S += I;
    }
    label1.Text = Convert.ToString(S);
}
```

## Exercício 1C - Laço: while



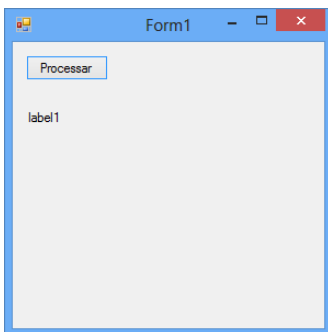
```
private void button1_Click(object sender, EventArgs e)
{
    int I = 0, R;
    label1.Text = "";
    while (I < 30)
    {
        R = I % 4;
        if (R == 0)
        {
            label1.Text += Convert.ToString(I) + " ";
        }
        I++;
    }
}
```

## Exercício 2C - Laço: do...while



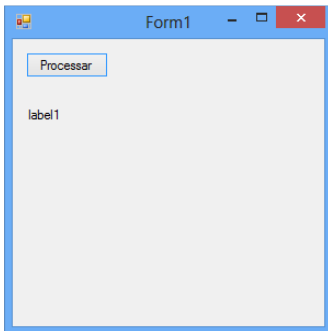
```
private void button1_Click(object sender, EventArgs e)
{
    int I = 0, R;
    label1.Text = "";
    do
    {
        R = I % 4;
        if (R == 0)
        {
            label1.Text += Convert.ToString(I) + " ";
        }
        I++;
    }
    while (I < 30);
}
```

### Exercício 3C - Laço: for



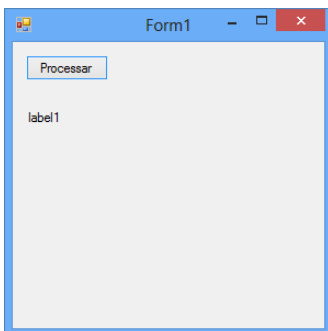
```
private void button1_Click(object sender, EventArgs e)
{
    int I, R;
    label1.Text = "";
    for (I = 1; I < 30; I++)
    {
        R = I % 4;
        if (R == 0)
        {
            label1.Text += Convert.ToString(I) + " ";
        }
    }
}
```

### Exercício 1D - Laço: while



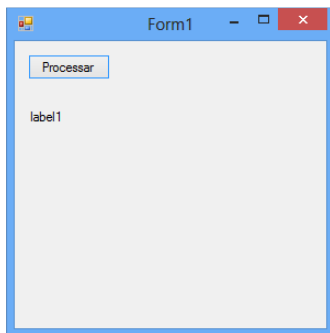
```
private void button1_Click(object sender, EventArgs e)
{
    double I = 15, R;
    label1.Text = "";
    while (I <= 200)
    {
        R = Math.Pow(I, 2);
        label1.Text += Convert.ToString(R) + " ";
        I++;
    }
}
```

## Exercício 2D - Laço: do...while



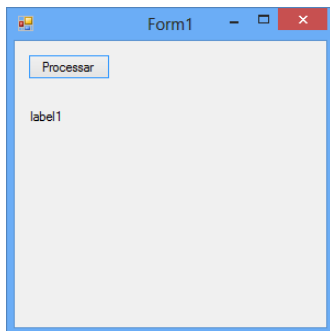
```
private void button1_Click(object sender, EventArgs e)
{
    double I = 15, R;
    label1.Text = "";
    do
    {
        R = Math.Pow(I, 2);
        label1.Text += Convert.ToString(R) + " ";
        I++;
    }
    while (I <= 200);
}
```

## Exercício 3D - Laço: for



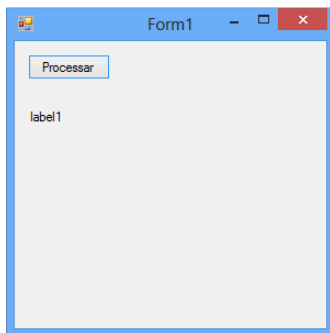
```
private void button1_Click(object sender, EventArgs e)
{
    double I, R;
    label1.Text = "";
    for (I = 15; I <= 200; I++)
    {
        R = Math.Pow(I, 2);
        label1.Text += Convert.ToString(R) + " ";
    }
}
```

## Exercício 1E - Laço: while



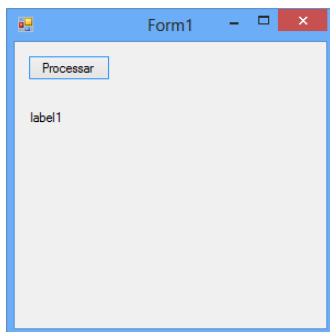
```
private void button1_Click(object sender, EventArgs e)
{
    int I, S = 0, R;
    label1.Text = "";
    I = 1;
    while (I <= 50)
    {
        R = I % 2;
        if (R == 0)
        {
            S += I;
        }
        I++;
    }
    label1.Text = Convert.ToString(S);
}
```

## Exercício 2E - Laço: do...while



```
private void button1_Click(object sender, EventArgs e)
{
    int I, S = 0, R;
    label1.Text = "";
    I = 1;
    do
    {
        R = I % 2;
        if (R == 0)
        {
            S += I;
        }
        I++;
    }
    while (I <= 50);
    label1.Text = Convert.ToString(S);
}
```

## Exercício 3E - Laço: for



```
private void button1_Click(object sender, EventArgs e)
{
    int I, S = 0, R;
    label1.Text = "";
    for (I = 1; I <= 50; I++)
    {
        R = I % 2;
        if (R == 0)
        {
            S += I;
        }
    }
    label1.Text = Convert.ToString(S);
}
```

---

# Capítulo 5

---

## Exercício 1A

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Gab06_Ex01
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] A = new int[8];
            int[] B = new int[8];

            int I;

            for (I = 0; I <= 7; I++)
            {
                Console.Write("Entre {0,2}o. valor: ", I + 1);
                A[I] = int.Parse(Console.ReadLine());
            }

            for (I = 0; I <= 7; I++)
            {
                B[I] = A[I] * 3;
            }
            Console.WriteLine();

            for (I = 0; I <= 7; I++)
            {
                Console.WriteLine("B[{0}] = {1,4}", I + 1, B[I]);
            }

            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }
    }
}

```

## Exercício 1B

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Gab06_Ex03
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] A = new int[20];
            int[] B = new int[20];
            int[] C = new int[20];

            int I;

            for (I = 0; I <= 19; I++)

```

```

    {
        Console.Write("Entre {0,2}o. valor - arranjo [A]: ", I + 1);
        A[I] = int.Parse(Console.ReadLine());
    }
    Console.WriteLine();

    for (I = 0; I <= 19; I++)
    {
        Console.Write("Entre {0,2}o. valor - arranjo [B]: ", I + 1);
        B[I] = int.Parse(Console.ReadLine());
    }

    for (I = 0; I <= 19; I++)
    {
        C[I] = A[I] - B[I];
    }
    Console.WriteLine();

    for (I = 0; I <= 19; I++)
    {
        Console.WriteLine("C[{0}] = {1,4}", I + 1, B[I]);
    }

    Console.WriteLine();
    Console.Write("Tecle algo para encerrar... ");
    Console.ReadKey();
}
}
}

```

## Exercício 1C

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Gab06_Ex05
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] A = new int[5];
            int[] B = new int[10];
            int[] C = new int[15];

            int I;

            for (I = 0; I <= 4; I++)
            {
                Console.Write("Entre {0,2}o. valor - arranjo [A]: ", I + 1);
                A[I] = int.Parse(Console.ReadLine());
            }
            Console.WriteLine();

            for (I = 0; I <= 9; I++)
            {
                Console.Write("Entre {0,2}o. valor - arranjo [B]: ", I + 1);
                B[I] = int.Parse(Console.ReadLine());
            }

            for (I = 0; I <= 14; I++)
            {

```

```

        if (I <= 4)
        {
            C[I] = A[I];
        }
        else
        {
            C[I] = B[I - 5];
        }
    }
    Console.WriteLine();

    for (I = 0; I <= 14; I++)
    {
        Console.WriteLine("C[{0,2}] = {1,4}", I + 1, C[I]);
    }

    Console.WriteLine();
    Console.Write("Tecle algo para encerrar... ");
    Console.ReadKey();
}
}
}

```

## Exercício 1D

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Gab06_Ex06
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] A = new int[15];
            int[] B = new int[15];

            int I;

            for (I = 0; I <= 14; I++)
            {
                Console.Write("Entre {0,2}o. valor - arranjo [A]: ", I + 1);
                A[I] = int.Parse(Console.ReadLine());
            }

            for (I = 0; I <= 14; I++)
            {
                B[I] = A[I] * A[I];
            }
            Console.WriteLine();

            for (I = 0; I <= 14; I++)
            {
                Console.WriteLine("B[{0,2}] = {1,4}", I + 1, B[I]);
            }

            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }
    }
}

```



## Exercício 1E

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Gab06_Ex07
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] A = new int[20];
            int[] B = new int[20];

            int I;

            for (I = 0; I <= 19; I++)
            {
                Console.Write("Entre {0,2}o. valor - arranjo [A]: ", I + 1);
                A[I] = int.Parse(Console.ReadLine());
            }

            for (I = 0; I <= 19; I++)
            {
                B[I] = A[19 - I];
            }
            Console.WriteLine();

            for (I = 0; I <= 19; I++)
            {
                Console.WriteLine("A[{0}] = {1} / B[{0}] = {2}", I + 1, A[I], B[I]);
            }

            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }
    }
}
```

## Exercício 2A

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc02a
{
    class Program
    {
        static void Main(string[] args)
        {
            int[,] A = new int[5, 3];
            int[,] B = new int[5, 3];
            int[,] C = new int[5, 3];

            int I, J, LIN, COL;

            Console.Clear();
            LIN = 7;
```

```

COL = 0;
Console.SetCursorPosition(0, 0);
Console.Write("Entre um valor na coordenada da matriz A:");
Console.SetCursorPosition(0, 2);
Console.Write("Linha ...: ");
Console.SetCursorPosition(0, 3);
Console.Write("Coluna ..: ");

for (I = 0; I <= 4; I++)
{
    for (J = 0; J <= 2; J++)
    {
        Console.SetCursorPosition(11, 2);
        Console.Write("{0}", I + 1);
        Console.SetCursorPosition(11, 3);
        Console.Write("{0}", J + 1);
        Console.SetCursorPosition(0, 5);
        Console.Write("--> ");
        A[I, J] = Convert.ToInt32(Console.ReadLine());
        Console.SetCursorPosition(4, 5);
        Console.Write(" ");
        Console.SetCursorPosition(COL, LIN);
        Console.Write("{0,4}", A[I, J]);
        COL += 6;
    }
    LIN += 2;
    COL -= 18;
}

Console.Clear();
LIN = 7;
COL = 0;
Console.SetCursorPosition(0, 0);
Console.Write("Entre um valor na coordenada da matriz B:");
Console.SetCursorPosition(0, 2);
Console.Write("Linha ...: ");
Console.SetCursorPosition(0, 3);
Console.Write("Coluna ..: ");

for (I = 0; I <= 4; I++)
{
    for (J = 0; J <= 2; J++)
    {
        Console.SetCursorPosition(11, 2);
        Console.Write("{0}", I + 1);
        Console.SetCursorPosition(11, 3);
        Console.Write("{0}", J + 1);
        Console.SetCursorPosition(0, 5);
        Console.Write("--> ");
        B[I, J] = Convert.ToInt32(Console.ReadLine());
        Console.SetCursorPosition(4, 5);
        Console.Write(" ");
        Console.SetCursorPosition(COL, LIN);
        Console.Write("{0,4}", B[I, J]);
        COL += 6;
    }
    LIN += 2;
    COL -= 18;
}

Console.Clear();
Console.SetCursorPosition(0, 13);
Console.Write("Saida - Adicao");
Console.SetCursorPosition(0, 15);

for (I = 0; I <= 4; I++)
{
    for (J = 0; J <= 2; J++)

```

```

        {
            C[I, J] = A[I, J] + B[I, J];
        }
    }

    for (I = 0; I <= 4; I++)
    {
        for (J = 0; J <= 2; J++)
        {
            Console.Write("C[{0},{1}]", I + 1, J + 1);
            Console.WriteLine(" = {0,4}", C[I, J]);
        }
    }

    Console.WriteLine();
    Console.Write("Tecle algo para encerrar... ");
    Console.ReadKey();
}
}
}

```

## Exercício 2B

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc02b
{
    class Program
    {
        static void Main(string[] args)
        {
            int[, ] A = new int[4, 5];

            int I, J, LIN, COL;

            Console.Clear();
            LIN = 7;
            COL = 0;
            Console.SetCursorPosition(0, 0);
            Console.Write("Entre um valor na coordenada da matriz A:");
            Console.SetCursorPosition(0, 2);
            Console.Write("Linha ...: ");
            Console.SetCursorPosition(0, 3);
            Console.Write("Coluna ...: ");

            for (I = 0; I <= 3; I++)
            {
                for (J = 0; J <= 4; J++)
                {
                    Console.SetCursorPosition(11, 2);
                    Console.Write("{0}", I + 1);
                    Console.SetCursorPosition(11, 3);
                    Console.Write("{0}", J + 1);
                    Console.SetCursorPosition(0, 5);
                    Console.Write("--> ");
                    A[I, J] = Convert.ToInt32(Console.ReadLine());
                    Console.SetCursorPosition(4, 5);
                    Console.Write("    ");
                    Console.SetCursorPosition(COL, LIN);
                    Console.Write("{0,4}", A[I, J]);
                    COL += 6;
                }
            }
        }
    }
}

```

```

        LIN += 2;
        COL = 0;
    }

    Console.Clear();
    for (I = 0; I <= 3; I++)
    {
        for (J = 0; J <= 4; J++)
        {
            Console.Write("A[{0},{1}]", I + 1, J + 1);
            Console.WriteLine(" = {0,4}", A[I, J]);
        }
    }

    Console.WriteLine();
    Console.Write("Tecle algo para encerrar... ");
    Console.ReadKey();
}
}
}

```

## Exercício 2C

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc02a
{
    class Program
    {
        static void Main(string[] args)
        {
            float[,] A = new float[4, 4];
            float[,] B = new float[4, 4];
            float[,] C = new float[4, 4];

            int I, J, LIN, COL;

            Console.Clear();
            LIN = 7;
            COL = 0;
            Console.SetCursorPosition(0, 0);
            Console.Write("Entre um valor na coordenada da matriz A:");
            Console.SetCursorPosition(0, 2);
            Console.Write("Linha ...: ");
            Console.SetCursorPosition(0, 3);
            Console.Write("Coluna ..: ");

            for (I = 0; I <= 3; I++)
            {
                for (J = 0; J <= 3; J++)
                {
                    Console.SetCursorPosition(11, 2);
                    Console.Write("{0}", I + 1);
                    Console.SetCursorPosition(11, 3);
                    Console.Write("{0}", J + 1);
                    Console.SetCursorPosition(0, 5);
                    Console.Write("--> ");
                    A[I, J] = Convert.ToSingle(Console.ReadLine());
                    Console.SetCursorPosition(4, 5);
                    Console.Write(" ");
                    Console.SetCursorPosition(COL, LIN);
                    Console.Write("{0,4}", A[I, J]);
                }
            }
        }
    }
}

```

```

        COL += 6;
    }
    LIN += 2;
    COL = 0;
}

Console.Clear();
LIN = 7;
COL = 0;
Console.SetCursorPosition(0, 0);
Console.Write("Entre um valor na coordenada da matriz B:");
Console.SetCursorPosition(0, 2);
Console.Write("Linha ...: ");
Console.SetCursorPosition(0, 3);
Console.Write("Coluna ...: ");

for (I = 0; I <= 3; I++)
{
    for (J = 0; J <= 3; J++)
    {
        Console.SetCursorPosition(11, 2);
        Console.Write("{0}", I + 1);
        Console.SetCursorPosition(11, 3);
        Console.Write("{0}", J + 1);
        Console.SetCursorPosition(0, 5);
        Console.Write("--> ");
        B[I, J] = Convert.ToSingle(Console.ReadLine());
        Console.SetCursorPosition(4, 5);
        Console.Write(" ");
        Console.SetCursorPosition(COL, LIN);
        Console.Write("{0,4}", B[I, J]);
        COL += 6;
    }
    LIN += 2;
    COL = 0;
}

Console.Clear();
Console.SetCursorPosition(0, 13);
Console.Write("Saida - Subtracao");
Console.SetCursorPosition(0, 15);

for (I = 0; I <= 3; I++)
{
    for (J = 0; J <= 3; J++)
    {
        C[I, J] = A[I, J] - B[I, J];
    }
}

for (I = 0; I <= 3; I++)
{
    for (J = 0; J <= 3; J++)
    {
        Console.Write("C[{0},{1}]", I + 1, J + 1);
        Console.WriteLine(" = {0,4}", C[I, J]);
    }
}

Console.WriteLine();
Console.Write("Tecle algo para encerrar... ");
Console.ReadKey();
}
}
}

```

## Exercício 2D

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap05_Ex03
{
    class Program
    {
        static void Main(string[] args)
        {
            float[,] A = new float[4, 4];

            int I, J, LIN = 7, COL = 0;

            Console.Clear();
            Console.SetCursorPosition(0, 0);
            Console.Write("Entre um valor na coordenada:");
            Console.SetCursorPosition(0, 2);
            Console.Write("Linha ...: ");
            Console.SetCursorPosition(0, 3);
            Console.Write("Coluna ...: ");

            for (I = 0; I <= 3; I++)
            {
                for (J = 0; J <= 3; J++)
                {
                    Console.SetCursorPosition(11, 2);
                    Console.Write("{0}", I + 1);
                    Console.SetCursorPosition(11, 3);
                    Console.Write("{0}", J + 1);
                    Console.SetCursorPosition(0, 5);
                    Console.Write("--> ");
                    A[I, J] = Convert.ToSingle(Console.ReadLine());
                    Console.SetCursorPosition(4, 5);
                    Console.Write(" ");
                    Console.SetCursorPosition(COL, LIN);
                    Console.Write("{0,4}", A[I, J]);
                    COL += 6;
                }
                LIN += 2;
                COL = 0;
            }

            Console.Clear();
            Console.SetCursorPosition(0, 13);
            Console.Write("Saida - Diagonal Principal");
            Console.SetCursorPosition(0, 15);

            for (I = 0; I <= 3; I++)
            {
                for (J = 0; J <= 3; J++)
                {
                    if (I == J)
                    {
                        Console.Write("A[{0},{1}]", I + 1, J + 1);
                        Console.WriteLine(" = {0,4}", A[I, J]);
                    }
                }
            }

            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }
    }
}
```

```

    }
}

```

## Exercício 2E

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap05_Ex03
{
    class Program
    {
        static void Main(string[] args)
        {
            float[,] A = new float[3, 3];

            int I, J, LIN = 7, COL = 0;

            Console.Clear();
            Console.SetCursorPosition(0, 0);
            Console.Write("Entre um valor na coordenada:");
            Console.SetCursorPosition(0, 2);
            Console.Write("Linha ... ");
            Console.SetCursorPosition(0, 3);
            Console.Write("Coluna ... ");

            for (I = 0; I <= 2; I++)
            {
                for (J = 0; J <= 2; J++)
                {
                    Console.SetCursorPosition(11, 2);
                    Console.Write("{0}", I + 1);
                    Console.SetCursorPosition(11, 3);
                    Console.Write("{0}", J + 1);
                    Console.SetCursorPosition(0, 5);
                    Console.Write("--> ");
                    A[I, J] = Convert.ToSingle(Console.ReadLine());
                    Console.SetCursorPosition(4, 5);
                    Console.Write(" ");
                    Console.SetCursorPosition(COL, LIN);
                    Console.Write("{0,4}", A[I, J]);
                    COL += 6;
                }
                LIN += 2;
                COL = 0;
            }

            Console.Clear();
            Console.Write("Saida - Diagonal Principal");
            Console.SetCursorPosition(0, 3);

            for (I = 0; I <= 2; I++)
            {
                for (J = 0; J <= 2; J++)
                {
                    if (I == J)
                    {
                        Console.Write("A[{0},{1}]", I + 1, J + 1);
                        Console.WriteLine(" = {0,4}", A[I, J] * 2);
                    }
                    else
                    {

```

```

        Console.WriteLine("A[{0},{1}]", I + 1, J + 1);
        Console.WriteLine(" = {0,4}", A[I, J] * 3);
    }
}

Console.WriteLine();
Console.Write("Tecle algo para encerrar... ");
Console.ReadKey();
}
}
}

```

---

## Capítulo 6

---

### Exercício 1A - proc. pas. de parâm. por valor

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc01a_pppv
{
    class Program
    {
        static void Main(string[] args)
        {
            int X;
            Console.Write("Entre um valor numerico: ");
            X = int.Parse(Console.ReadLine());
            Console.WriteLine();
            Somatorio(X);
            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }

        public static void Somatorio(int N)
        {
            long SOMA = 0;
            for (int I = 1; I <= N; I++)
            {
                SOMA += I;
            }
            Console.WriteLine("Somatorio de ate {0} = {1}", N, SOMA);
        }
    }
}

```

### Exercício 1A - proc. pas. de parâm. por referência

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc01a_pppr
{
    class Program
    {
        static void Main(string[] args)

```



```

    {
        int X;
        long Y = 0;
        Console.Write("Entre um valor numerico: ");
        X = int.Parse(Console.ReadLine());
        Console.WriteLine();
        Somatorio(X, ref Y);
        Console.WriteLine("Somatorio de ate {0} = {1}", X, Y);
        Console.WriteLine();
        Console.Write("Tecle algo para encerrar... ");
        Console.ReadKey();
    }

    public static void Somatorio(int N, ref long SOMA)
    {
        for (int I = 1; I <= N; I++)
        {
            SOMA += I;
        }
    }
}

```

## Exercício 1A - função

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc01a_f
{
    class Program
    {
        static void Main(string[] args)
        {
            int X;
            long R;
            Console.Write("Entre um valor numerico: ");
            X = int.Parse(Console.ReadLine());
            Console.WriteLine();
            R = Somatorio(X);
            Console.WriteLine("Somatorio de ate {0} = {1}", X, R);
            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }

        public static long Somatorio(int N)
        {
            long SOMA = 0;
            for (int I = 1; I <= N; I++)
            {
                SOMA += I;
            }
            return SOMA;
        }
    }
}

```

## Exercício 1B - proc. pas. de parâm. por valor

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc01b_pppv
{
    class Program
    {
        static void Main(string[] args)
        {
            int X;
            Console.Write("Entre o termo Fibonacci: ");
            X = int.Parse(Console.ReadLine());
            Console.WriteLine();
            Fibonacci(X);
            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }

        public static void Fibonacci(int N)
        {
            int ATUAL, ANTERIOR, PROXIMO;
            ANTERIOR = 0;
            ATUAL = 1;
            for (int I = 1; I <= N; I++)
            {
                PROXIMO = ATUAL + ANTERIOR;
                ANTERIOR = ATUAL;
                ATUAL = PROXIMO;
            }
            Console.WriteLine("Resultado = {0}", ANTERIOR);
        }
    }
}

```

## Exercício 1B - proc. pas. de parâm. por referência

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc01b_pppr
{
    class Program
    {
        static void Main(string[] args)
        {
            int X, Y = 0;
            Console.Write("Entre o termo Fibonacci: ");
            X = int.Parse(Console.ReadLine());
            Console.WriteLine();
            Fibonacci(X, ref Y);
            Console.WriteLine("Resultado = {0}", Y);
            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }

        public static void Fibonacci(int N, ref int ANTERIOR)
        {
            int ATUAL, PROXIMO;
            ATUAL = 1;
            for (int I = 1; I <= N; I++)
            {

```

```

        PROXIMO = ATUAL + ANTERIOR;
        ANTERIOR = ATUAL;
        ATUAL = PROXIMO;
    }
}
}

```

## Exercício 1B - função

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc01b_f
{
    class Program
    {
        static void Main(string[] args)
        {
            int X, R;
            Console.Write("Entre o termo Fibonacci: ");
            X = int.Parse(Console.ReadLine());
            Console.WriteLine();
            R = Fibonacci(X);
            Console.WriteLine("Resultado = {0}", R);
            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }

        public static int Fibonacci(int N)
        {
            int ATUAL, ANTERIOR, PROXIMO;
            ANTERIOR = 0;
            ATUAL = 1;
            for (int I = 1; I <= N; I++)
            {
                PROXIMO = ATUAL + ANTERIOR;
                ANTERIOR = ATUAL;
                ATUAL = PROXIMO;
            }
            return ANTERIOR;
        }
    }
}

```

## Exercício 1C - proc. pas. de parâm. por valor

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc01c_pppv
{
    class Program
    {
        public static void Prestacao(float V, float TX, float T)
        {
            float P = V + (V * (TX/100) * T);
            Console.WriteLine("Prestacao R$ {0:##,##0.00}", P);
        }

        static void Main(string[] args)
        {

```

```

    {
        float VALOR, TAXA, TEMPO;
        Console.Write("Entre o valor ....: ");
        VALOR = float.Parse(Console.ReadLine());
        Console.Write("Entre a taxa .....: ");
        TAXA = float.Parse(Console.ReadLine());
        Console.Write("Entre o tempo ....: ");
        TEMPO = float.Parse(Console.ReadLine());
        Console.WriteLine();
        Prestacao(VALOR, TAXA, TEMPO);
        Console.WriteLine();
        Console.Write("Tecle algo para encerrar... ");
        Console.ReadKey();
    }
}

```

## Exercício 1C - proc. pas. de parâm. por referência

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc01c_pppr
{
    class Program
    {
        public static void Prestacao(float V, float TX, float T, ref float P)
        {
            P = V + (V * (TX/100) * T);
        }

        static void Main(string[] args)
        {
            float VALOR, TAXA, TEMPO, PREST = 0;
            Console.Write("Entre o valor ....: ");
            VALOR = float.Parse(Console.ReadLine());
            Console.Write("Entre a taxa .....: ");
            TAXA = float.Parse(Console.ReadLine());
            Console.Write("Entre o tempo ....: ");
            TEMPO = float.Parse(Console.ReadLine());
            Console.WriteLine();
            Prestacao(VALOR, TAXA, TEMPO, ref PREST);
            Console.WriteLine("Prestacao R$ {0:##,##0.00}", PREST);
            Console.WriteLine();
            Console.Write("Tecle algo para encerrar... ");
            Console.ReadKey();
        }
    }
}

```

## Exercício 1C - função

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace Cap06Exerc01c_f
{
    class Program
    {

```

```

    public static float Prestacao(float V, float TX, float T)
    {
        float P = V + (V * (TX / 100) * T);
        return P;
    }

    static void Main(string[] args)
    {
        float PRESTACAO, VALOR, TAXA, TEMPO;
        Console.Write("Entre o valor ....: ");
        VALOR = float.Parse(Console.ReadLine());
        Console.Write("Entre a taxa .....: ");
        TAXA = float.Parse(Console.ReadLine());
        Console.Write("Entre o tempo ....: ");
        TEMPO = float.Parse(Console.ReadLine());
        Console.WriteLine();
        PRESTACAO = Prestacao(VALOR, TAXA, TEMPO);
        Console.WriteLine("Prestacao R$ {0:##,##0.00}", PRESTACAO);
        Console.WriteLine();
        Console.Write("Tecle algo para encerrar... ");
        Console.ReadKey();
    }
}

```

---

## Capítulo 7

---

### Exercício 1a

Com o projeto Cap07Exerc01a criado execute o comando de menu PROJECT/Add Class, na caixa de diálogo Add New Item - Cap07Exerc01a, informe o nome da classe Pessoa no campo Name e acione o botão Add. Na apresentação da guia do código Pessoa.cs informe o trecho seguinte dentro de class Pessoa:

```

public string NOME;
public int IDADE;
public string SEXO;

```

De modo que o código da guia Pessoa.cs fique como:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Cap07Exerc01a
{
    struct Pessoa
    {
        public string NOME;
        public int IDADE;
        public string SEXO;
    }
}

```

Selecione a guia Program.cs para instanciar o objeto junto a variável X na função Main() a partir do trecho de código:

```

static Pessoa[] X = new Pessoa[10];

```

De modo que o código da guia Program.cs fique como:

```

using System;
using System.Collections.Generic;

```

```

using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Cap07Exerc01a
{
    class Program
    {
        static Pessoa[] X = new Pessoa[10];
        static string MSG = "Tecle algo para continuar... ";

        static void Main(string[] args)
        {
            string OPCAO = "0";

            while (OPCAO != "3")
            {
                Console.Clear();
                Console.WriteLine("Menu Principal");
                Console.WriteLine("-----");
                Console.WriteLine();
                Console.WriteLine("1 - Entrar dados");
                Console.WriteLine("2 - Mostrar dados");
                Console.WriteLine("3 - Fim de Programa");
                Console.WriteLine();
                Console.Write("Escolha uma opcao: ");
                OPCAO = Console.ReadLine();
                if (OPCAO != "3")
                {
                    switch (OPCAO)
                    {
                        case "1": Entrada(); break;
                        case "2": Saida(); break;
                        default:
                            Console.WriteLine("Opcao invalida");
                            Console.Write(MSG);
                            Console.ReadKey();
                            break;
                    }
                }
            }
        }

        private static void Entrada()
        {
            Console.Clear();
            Console.WriteLine("Entrada de Dados");
            Console.WriteLine("-----");
            for (int I = 0; I <= 9; I++)
            {
                Console.WriteLine();
                Console.WriteLine("Entre os dados da {0,2}a. pessoa.", I + 1);
                Console.WriteLine();
                Console.Write("Nome ...: ");
                X[I].NOME = Console.ReadLine();
                Console.Write("Sexo ...: ");
                X[I].SEXO = Console.ReadLine();
                Console.Write("Idade ..: ");
                X[I].IDADE = Convert.ToInt32(Console.ReadLine());
            }
        }

        private static void Saida()
        {
            Console.Clear();
            Console.WriteLine("Saida de Dados");
        }
    }
}

```

```

        Console.WriteLine("-----");
        Console.WriteLine();
        for (int I = 0; I <= 9; I++)
        {
            Console.WriteLine("Pessoa {0,2}.", I + 1);
            Console.WriteLine();
            Console.WriteLine("Nome ...: {0}", X[I].NOME);
            Console.WriteLine("Sexo ...: {0}", X[I].SEXO);
            Console.WriteLine("Idade ...: {0}", X[I].IDADE);
            Console.WriteLine();
        }
        Console.Write(MSG);
        Console.ReadKey();
    }
}

```

## Exercício 1b

Com o projeto Cap07Exerc01b criado execute o comando de menu PROJECT/Add Class, na caixa de diálogo Add New Item - Cap07Exerc01b, informe o nome da classe Pessoa no campo Name e acione o botão Add. Na apresentação da guia do código Pessoa.cs informe o trecho seguinte dentro de class Pessoa:

```

private string NOME;
private int IDADE;
private string SEXO;

```

Após definir os campos defina os métodos get/set de acesso aos campos privados com o trecho de código:

```

public string AcessaNOME
{
    get
    {
        return NOME;
    }
    set
    {
        NOME = value;
    }
}

public int AcessaIDADE
{
    get
    {
        return IDADE;
    }
    set
    {
        IDADE = value;
    }
}

public string AcessaSEXO
{
    get
    {
        return SEXO;
    }
    set
    {
        SEXO = value;
    }
}

```

De modo que o código da guia Pessoa.cs fique como:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Cap07Exerc01b
{
    struct Pessoa
    {
        private string NOME;
        private int IDADE;
        private string SEXO;

        public string AcessaNOME
        {
            get
            {
                return NOME;
            }
            set
            {
                NOME = value;
            }
        }

        public int AcessaIDADE
        {
            get
            {
                return IDADE;
            }
            set
            {
                IDADE = value;
            }
        }

        public string AcessaSEXO
        {
            get
            {
                return SEXO;
            }
            set
            {
                SEXO = value;
            }
        }
    }
}
```

Selecione a guia Program.cs para instanciar o objeto junto a variável X na função Main() a partir do trecho de código:

```
static Pessoa[] X = new Pessoa[10];
```

De modo que o código da guia Program.cs fique como:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
```



```

namespace Cap07Exerc01b
{
    class Program
    {
        static Pessoa[] X = new Pessoa[10];
        static string MSG = "Tecle algo para continuar... ";

        static void Main(string[] args)
        {
            string OPCAO = "0";

            while (OPCAO != "3")
            {
                Console.Clear();
                Console.WriteLine("Menu Principal");
                Console.WriteLine("-----");
                Console.WriteLine();
                Console.WriteLine("1 - Entrar dados");
                Console.WriteLine("2 - Mostrar dados");
                Console.WriteLine("3 - Fim de Programa");
                Console.WriteLine();
                Console.Write("Escolha uma opcao: ");
                OPCAO = Console.ReadLine();
                if (OPCAO != "3")
                {
                    switch (OPCAO)
                    {
                        case "1": Entrada(); break;
                        case "2": Saida(); break;
                        default:
                            Console.WriteLine("Opcao invalida");
                            Console.Write(MSG);
                            Console.ReadKey();
                            break;
                    }
                }
            }

            private static void Entrada()
            {
                Console.Clear();
                Console.WriteLine("Entrada de Dados");
                Console.WriteLine("-----");
                for (int I = 0; I <= 9; I++)
                {
                    Console.WriteLine();
                    Console.WriteLine("Entre os dados da {0,2}a. pessoa.", I + 1);
                    Console.WriteLine();
                    Console.Write("Nome ...: ");
                    X[I].AcessaNOME = Console.ReadLine();
                    Console.Write("Sexo ...: ");
                    X[I].AcessaSEXO = Console.ReadLine();
                    Console.Write("Idade ...: ");
                    X[I].AcessaIDADE = Convert.ToInt32(Console.ReadLine());
                }

            private static void Saida()
            {
                Console.Clear();
                Console.WriteLine("Saida de Dados");
                Console.WriteLine("-----");
                Console.WriteLine();
                for (int I = 0; I <= 9; I++)
                {
                    Console.WriteLine("Pessoa {0,2}.", I + 1);
                }
            }
        }
    }
}

```

```

        Console.WriteLine();
        Console.WriteLine("Nome ...: {0}", X[I].AcessaNOME);
        Console.WriteLine("Sexo ...: {0}", X[I].AcessaSEXO);
        Console.WriteLine("Idade ...: {0}", X[I].AcessaIDADE);
        Console.WriteLine();
    }
    Console.Write(MSG);
    Console.ReadKey();
}
}
}

```

## Exercício 1c

Com o projeto Cap07Exerc01c criado execute o comando de menu PROJECT/Add Class, na caixa de diálogo Add New Item - Cap07Exerc01c, informe o nome da classe ContaCorrente no campo Name e acione o botão Add. Na apresentação da guia do código ContaCorrente.cs informe o trecho seguinte dentro de class ContaCorrente:

```

public int NUMERO;
private double SALDO;

public void Sacar(double VALOR)
{
    SALDO -= VALOR;
}

public void Depositar(double VALOR)
{
    SALDO += VALOR;
}

public double AcessaSALDO
{
    get
    {
        return SALDO;
    }
}

```

De modo que o código da guia ContaCorrente.cs fique como:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Cap07Exerc01c
{
    class ContaCorrente
    {
        public int NUMERO;
        private double SALDO;

        public void Sacar(double VALOR)
        {
            SALDO -= VALOR;
        }

        public void Depositar(double VALOR)
        {
            SALDO += VALOR;
        }

        public double AcessaSALDO
        {

```

```

        get
        {
            return SALDO;
        }
    }
}

```

Selecione a guia Program.cs para instanciar o objeto junto a variável X na função Main() a partir do telhado de código:

```
static ContaCorrente Banco = new ContaCorrente();
```

De modo que o código da guia Program.cs fique como:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Cap07Exerc01c
{
    class Program
    {
        static ContaCorrente Banco = new ContaCorrente();
        static string MSG = "Tecle algo para continuar... ";

        static void Main(string[] args)
        {
            string OPCAO = "0";

            while (OPCAO != "3")
            {
                Console.Clear();
                Console.WriteLine("Saldo Bancario");
                Console.WriteLine("-----");
                Console.WriteLine();
                Console.WriteLine("Seu saldo = {0:0.00}", Banco.AcessaSALDO);
                Console.WriteLine();
                Console.WriteLine("1 - Depositar");
                Console.WriteLine("2 - Sacar");
                Console.WriteLine("3 - Fim de Programa");
                Console.WriteLine();
                Console.Write("Escolha uma opcao: ");
                OPCAO = Console.ReadLine();
                if (OPCAO != "3")
                {
                    switch (OPCAO)
                    {
                        case "1": Deposito(); break;
                        case "2": Saque(); break;
                        default:
                            Console.WriteLine("Opcao invalida");
                            Console.Write(MSG);
                            Console.ReadKey();
                            break;
                    }
                }
            }
        }

        private static void Deposito()
        {
            double VALOR;
            Console.Clear();

```

```

        Console.WriteLine("Acao: Depositar");
        Console.WriteLine("-----");
        Console.WriteLine();
        Console.WriteLine("Saldo antes do deposito = {0:0.00}", Banco.AcessaSALDO);
        Console.WriteLine();
        Console.Write("Entre valor de deposito: ");
        VALOR = Convert.ToDouble(Console.ReadLine());
        Banco.Depositar(VALOR);
        Console.WriteLine();
        Console.WriteLine("Saldo apos deposito = {0:0.00}", Banco.AcessaSALDO);
        Console.WriteLine();
        Console.Write(MSG);
        Console.ReadKey();
    }

    private static void Saque()
    {
        double VALOR;
        Console.Clear();
        Console.WriteLine("Acao: Sacar");
        Console.WriteLine("-----");
        Console.WriteLine();
        Console.WriteLine("Saldo antes do saque = {0:0.00}", Banco.AcessaSALDO);
        Console.WriteLine();
        Console.Write("Entre valor a ser sacado: ");
        VALOR = Convert.ToDouble(Console.ReadLine());
        Banco.Sacar(VALOR);
        Console.WriteLine();
        Console.WriteLine("Saldo apos saque = {0:0.00}", Banco.AcessaSALDO);
        Console.WriteLine();
        Console.Write(MSG);
        Console.ReadKey();
    }
}

```