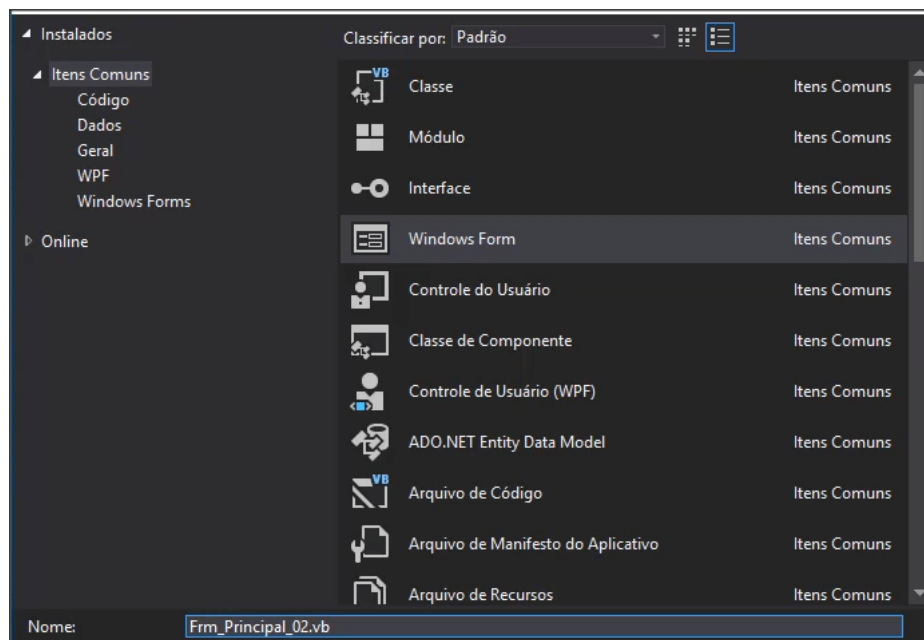


Mãos na massa

Chegou a hora de você pôr em prática o que foi visto na aula. Para isso, execute os passos listados abaixo.

1) Crie um segundo formulário no **projeto 1**. Clique com o botão da direita do mouse sobre o nome do projeto, selecione **Adicionar --> Novo Item** e escolha a opção **Windows Form**, dando o nome `Frm_Principal_02` :



2) Modifique o `Name` do novo formulário para `Frm_TelaPrincipal_02`, o `StartPosition` para `WindowsDefaultLocation` e `WindowState` para `Normal`.

3) Ao novo *Form*, adicione um *label*, com `Name` `Lbl_Principal_02`, e dois *GroupBox*, com `Name` `Grp_Gabriela` e `Grp_Bruno`, respectivamente.

4) Para cada *GroupBox*, adicione cinco *Label*, cinco *TextBox*, e três *Button*.

5) No `Grp_Gabriela`, os componentes terão os seguintes `Name` :

- `Lbl_BemVindo_Gabriela`
- `Lbl_SaldoAtual_Gabriela`
- `Lbl_Valor_Gabriela`
- `Lbl_Saldo_Gabriela`
- `Lbl_Resultado_Gabriela`
- `Txt_SaldoAtual_Gabriela`
- `Txt_Valor_Gabriela`
- `Txt_Saldo_Gabriela`
- `Txt_Resultado_Gabriela`
- `Txt_Extrato_Gabriela` (esse também terá o `Multiline` igual a `True` e a propriedade `ScrollBars` com o valor `Vertical`)
- `Btn_Depositar_Gabriela`
- `Btn_Sacar_Gabriela`

- Btn_Transferir_Gabriela

6) E no Grp_Bruno :

- Lbl_BemVindo_Bruno
- Lbl_SaldoAtual_Bruno
- Lbl_Valor_Bruno
- Lbl_Saldo_Bruno
- Lbl_Resultado_Bruno
- Txt_SaldoAtual_Bruno
- Txt_Valor_Bruno
- Txt_Saldo_Bruno
- Txt_Resultado_Bruno
- Txt_Extrato_Bruno (esse também terá o Multiline igual a True e a propriedade ScrollBars com o valor Vertical)
- Btn_Depositar_Bruno
- Btn_Sacar_Bruno
- Btn_Transferir_Bruno

7) Modifique a propriedade Text do formulário, dos GroupBox, dos Label e dos Button na sub-rotina New do formulário:

```
Public Sub New()
```

```
    ' Esta chamada é requerida pelo designer.  
    InitializeComponent()  
  
    ' Adicione qualquer inicialização após a chamada InitializeComponent().  
  
    Me.Text = "1 - Primeira Classe - Formulário 02"  
    Lbl_Principal_02.Text = "1 - Primeira Classe - Formulário 02"  
  
    ' Inicialização dos textos para Gabriela  
  
    Grp_Gabriela.Text = "Gabriela"  
    Lbl_Valor_Gabriela.Text = "Valor a ser depositado/sacado"  
    Lbl_Saldo_Gabriela.Text = "Novo Saldo"  
    Lbl_Resultado_Gabriela.Text = "Resultado do saque"  
    Lbl_SaldoAtual_Gabriela.Text = "Saldo Atual"  
    Btn_Sacar_Gabriela.Text = "Efetuar Saque"  
    Btn_Depositar_Gabriela.Text = "Efetuar Depósito"  
    Btn_Transferir_Gabriela.Text = "Efetuar Transferencia"  
  
    ' Inicialização dos textos para Bruno  
  
    Grp_Bruno.Text = "Bruno"  
    Lbl_Valor_Bruno.Text = "Valor a ser depositado/sacado"  
    Lbl_Saldo_Bruno.Text = "Novo Saldo"  
    Lbl_Resultado_Bruno.Text = "Resultado do saque"  
    Lbl_SaldoAtual_Bruno.Text = "Saldo Atual"  
    Btn_Sacar_Bruno.Text = "Efetuar Saque"  
    Btn_Depositar_Bruno.Text = "Efetuar Depósito"  
    Btn_Transferir_Bruno.Text = "Efetuar Transferencia"
```

End Sub

8) O único *TextBox* que pode ter seu valor editado pelo usuário é que representa o valor a ser depositado/sacado (*Txt_Valor_Gabriela* e *Txt_Valor_Bruno*). Para evitar esse problema, inicialize a propriedade *ReadOnly* dos *TextBox* que não poderão ser editados:

```
Public Sub New()  
  
    ' Esta chamada é requerida pelo designer.  
    InitializeComponent()  
  
    ' Adicione qualquer inicialização após a chamada InitializeComponent().  
  
    Me.Text = "1 - Primeira Classe - Formulário 02"  
    Lbl_Principal_02.Text = "1 - Primeira Classe - Formulário 02"  
  
    ' Inicialização dos textos para Gabriela  
  
    Grp_Gabriela.Text = "Gabriela"  
    Lbl_Valor_Gabriela.Text = "Valor a ser depositado/sacado"  
    Lbl_Saldo_Gabriela.Text = "Novo Saldo"  
    Lbl_Resultado_Gabriela.Text = "Resultado do saque"  
    Lbl_SaldoAtual_Gabriela.Text = "Saldo Atual"  
    Btn_Sacar_Gabriela.Text = "Efetuar Saque"  
    Btn_Depositar_Gabriela.Text = "Efetuar Depósito"  
    Btn_Transferir_Gabriela.Text = "Efetuar Transferencia"  
    Txt_SaldoAtual_Gabriela.ReadOnly = True  
    Txt_Saldo_Gabriela.ReadOnly = True  
    Txt_Resultado_Gabriela.ReadOnly = True  
    Txt_Extrato_Gabriela.ReadOnly = True  
  
    ' Inicialização dos textos para Bruno  
  
    Grp_Bruno.Text = "Bruno"  
    Lbl_Valor_Bruno.Text = "Valor a ser depositado/sacado"  
    Lbl_Saldo_Bruno.Text = "Novo Saldo"  
    Lbl_Resultado_Bruno.Text = "Resultado do saque"  
    Lbl_SaldoAtual_Bruno.Text = "Saldo Atual"  
    Btn_Sacar_Bruno.Text = "Efetuar Saque"  
    Btn_Depositar_Bruno.Text = "Efetuar Depósito"  
    Btn_Transferir_Bruno.Text = "Efetuar Transferencia"  
    Txt_SaldoAtual_Bruno.ReadOnly = True  
    Txt_Saldo_Bruno.ReadOnly = True  
    Txt_Resultado_Bruno.ReadOnly = True  
    Txt_Extrato_Bruno.ReadOnly = True  
  
End Sub
```

9) Fora da sub-rotina *New*, instancie duas *ContaCorrente*, dando o nome *ContaDaGabriela* e *ContaDoBruno*. Em seguida, já dentro da rotina *New*, para as duas contas, escreva uma mensagem de boas-vindas junto com os dados da conta corrente e exiba o saldo no seu respectivo *TextBox*:

```
Dim ContaDaGabriela As New ContaCorrente  
Dim ContaDoBruno As New ContaCorrente
```

```
Public Sub New()  
  
    ' Esta chamada é requerida pelo designer.  
    InitializeComponent()  
  
    ' Adicione qualquer inicialização após a chamada InitializeComponent().  
  
    Me.Text = "1 - Primeira Classe - Formulário 02"  
    Lbl_Principal_02.Text = "1 - Primeira Classe - Formulário 02"  
  
    ' Inicialização dos textos para Gabriela  
  
    Lbl_Valor_Gabriela.Text = "Valor a ser depositado/sacado"  
    Btn_Sacar_Gabriela.Text = "Efetuar Saque"  
    Btn_Depositar_Gabriela.Text = "Efetuar Depósito"  
    Lbl_Saldo_Gabriela.Text = "Novo Saldo"  
    Lbl_Resultado_Gabriela.Text = "Resultado do saque"  
    Lbl_SaldoAtual_Gabriela.Text = "Saldo Atual"  
    Grp_Gabriela.Text = "Gabriela"  
    Btn_Transferir_Gabriela.Text = "Efetuar Transferencia"  
    Txt_SaldoAtual_Gabriela.ReadOnly = True  
    Txt_Saldo_Gabriela.ReadOnly = True  
    Txt_Resultado_Gabriela.ReadOnly = True  
    Txt_Extrato_Gabriela.ReadOnly = True  
  
    ' Inicialização dos textos para Bruno  
  
    Lbl_Valor_Bruno.Text = "Valor a ser depositado/sacado"  
    Btn_Sacar_Bruno.Text = "Efetuar Saque"  
    Btn_Depositar_Bruno.Text = "Efetuar Depósito"  
    Lbl_Saldo_Bruno.Text = "Novo Saldo"  
    Lbl_Resultado_Bruno.Text = "Resultado do saque"  
    Lbl_SaldoAtual_Bruno.Text = "Saldo Atual"  
    Btn_Transferir_Bruno.Text = "Efetuar Transferencia"  
    Grp_Bruno.Text = "Bruno"  
    Txt_SaldoAtual_Bruno.ReadOnly = True  
    Txt_Saldo_Bruno.ReadOnly = True  
    Txt_Resultado_Bruno.ReadOnly = True  
    Txt_Extrato_Bruno.ReadOnly = True  
  
    ' Inicialização dos dados da classe da Gabriela  
  
    ContaDaGabriela.Titular = "Gabriela"  
    ContaDaGabriela.Agencia = 863  
    ContaDaGabriela.Conta = 863141  
  
    Lbl_BemVindo_Gabriela.Text = "Bem Vindo " + ContaDaGabriela.Titular +  
        " Agência: " + ContaDaGabriela.Agencia.ToString +  
        " Conta Corrente: " + ContaDaGabriela.Conta.ToString  
    Txt_SaldoAtual_Gabriela.Text = ContaDaGabriela.Saldo.ToString  
  
    ' Inicialização dos dados da classe do Bruno  
  
    ContaDoBruno.Titular = "Bruno"  
    ContaDoBruno.Agencia = 863  
    ContaDoBruno.Conta = 863224
```

```
Lbl_BemVindo_Bruno.Text = "Bem Vindo " + ContaDoBruno.Titular +
    " Agência: " + ContaDoBruno.Agencia.ToString +
    " Conta Corrente: " + ContaDoBruno.Conta.ToString
Txt_SaldoAtual_Bruno.Text = ContaDoBruno.Saldo.ToString
```

```
End Sub
```

10) Conforme a definição da programação orientada a objetos, uma classe tem os chamados **eventos**. Uma conta corrente de um banco possui diversas ações, como **depositar**, **sacar** e **transferir**. Comece modelando o saque, na classe `ContaCorrente`, implementando o método `Sacar`. Esse método receberá um valor por parâmetro e irá verificar se ele é válido. Se for, ele desconta o valor do saldo e retorna `True`, mas se não for, não faz o desconto e retorna `False`:

```
Public Function Sacar(ValorSacar As Double) As Boolean
```

```
    Dim Retorno As Boolean
    If Saldo < ValorSacar Then
        Retorno = False
    Else
        Saldo -= ValorSacar
        Retorno = True
    End If
    Return Retorno
```

```
End Function
```

11) Além disso, adicione a propriedade chamada `Extrato`, que irá acumular todas as operações efetuadas na conta corrente:

```
Public Extrato As String = ""
```

12) Voltando ao *Designer*, dê um duplo clique no botão de saque da Gabriela e vá ao código fonte. Em seguida, implemente a lógica do saque, utilizando o método que você acabou de implementar, ou seja, caso o retorno do método seja `False`, exiba em `Txt_Resultado_Gabriela` uma mensagem dizendo que o saque não foi possível de ser feito, caso contrário, atualize o saldo, exiba uma mensagem de sucesso, atualize o extrato da instância e exiba-o:

```
Private Sub Btn_Sacar_Gabriela_Click(sender As Object, e As EventArgs) Handles Btn_Sacar_Gabriela.Click
```

```
    Txt_Resultado_Gabriela.Text = ""
    Txt_Saldo_Gabriela.Text = ""
```

```
    Dim ValorSacar As Double = Val(Txt_Valor_Gabriela.Text)
    Dim RetornoSaque As Boolean = ContaDaGabriela.Sacar(ValorSacar)
```

```
    If RetornoSaque = False Then
        Txt_Resultado_Gabriela.Text = "Saque não é possível ser feito"
    Else
```

```
        Txt_Saldo_Gabriela.Text = ContaDaGabriela.Saldo.ToString
        Txt_Resultado_Gabriela.Text = "Saque efetuado com sucesso"
        Txt_SaldoAtual_Gabriela.Text = Txt_Saldo_Gabriela.Text
        ContaDaGabriela.Extrato += Now.ToString + " Saque de " + ValorSacar.ToString +
            " Saldo " + ContaDaGabriela.Saldo.ToString + vbCrLf
        Txt_Extrato_Gabriela.Text = ContaDaGabriela.Extrato
```

```
End If
```

```
End Sub
```

13) Volte à classe `ContaCorrente` e implemente o método `Depositar`, que recebe um valor por parâmetro e o soma ao saldo:

```
Public Sub Depositar(ValorDepositar As Double)
    Saldo += ValorDepositar
End Sub
```

14) E na ação do clique do botão de depósito da Gabriela, faça:

```
Private Sub Btn_Depositar_Gabriela_Click(sender As Object, e As EventArgs) Handles Btn_Depositar_Gal

    Txt_Resultado_Gabriela.Text = ""
    Txt_Saldo_Gabriela.Text = ""

    Dim ValorDepositar As Double = Val(Txt_Valor_Gabriela.Text)
    ContaDaGabriela.Depositar(ValorDepositar)

    Txt_Saldo_Gabriela.Text = ContaDaGabriela.Saldo.ToString
    Txt_Resultado_Gabriela.Text = "Depósito efetuado com sucesso"
    Txt_SaldoAtual_Gabriela.Text = Txt_Saldo_Gabriela.Text
    ContaDaGabriela.Extrato += Now.ToString + " Depósito de " + ValorDepositar.ToString +
        " Saldo " + ContaDaGabriela.Saldo.ToString + vbCrLf
    Txt_Extrato_Gabriela.Text = ContaDaGabriela.Extrato

End Sub
```

15) Faça o mesmo para o saque e depósito do Bruno:

```
Private Sub Btn_Sacar_Bruno_Click(sender As Object, e As EventArgs) Handles Btn_Sacar_Bruno.Click

    Txt_Resultado_Bruno.Text = ""
    Txt_Saldo_Bruno.Text = ""

    Dim ValorSacar As Double = Val(Txt_Valor_Bruno.Text)
    Dim RetornoSaque As Boolean = ContaDoBruno.Sacar(ValorSacar)

    If RetornoSaque = False Then
        Txt_Resultado_Bruno.Text = "Saque não é possível ser feito"
    Else
        Txt_Saldo_Bruno.Text = ContaDoBruno.Saldo.ToString
        Txt_Resultado_Bruno.Text = "Saque efetuado com sucesso"
        Txt_SaldoAtual_Bruno.Text = Txt_Saldo_Bruno.Text
        ContaDoBruno.Extrato += Now.ToString + " Saque de " + ValorSacar.ToString +
            " Saldo " + ContaDoBruno.Saldo.ToString + vbCrLf
        Txt_Extrato_Bruno.Text = ContaDoBruno.Extrato
    End If

End Sub
```

```
Private Sub Btn_Depositar_Bruno_Click(sender As Object, e As EventArgs) Handles Btn_Depositar_Bruno.

    Txt_Resultado_Bruno.Text = ""
    Txt_Saldo_Bruno.Text = ""

    Dim ValorDepositar As Double = Val(Txt_Valor_Bruno.Text)
    ContaDoBruno.Depositar(ValorDepositar)

    Txt_Saldo_Bruno.Text = ContaDoBruno.Saldo.ToString
    Txt_Resultado_Bruno.Text = "Depósito efetuado com sucesso"
    Txt_SaldoAtual_Bruno.Text = Txt_Saldo_Bruno.Text
    ContaDoBruno.Extrato += Now.ToString + " Depósito de " + ValorDepositar.ToString +
        " Saldo " + ContaDoBruno.Saldo.ToString + vbCrLf
    Txt_Extrato_Bruno.Text = ContaDoBruno.Extrato

End Sub
```

16) Para finalizar o código, vá à classe `ContaCorrente` e insira o código que irá efetuar a transferência entre contas:

```
Public Function Transferir(ValorTransferencia As Double, ByRef ContaDestino As ContaCorrente) As Boolean

    Dim Retorno As Boolean
    If Saldo < ValorTransferencia Then
        Retorno = False
    Else
        Saldo -= ValorTransferencia
        ContaDestino.Depositar(ValorTransferencia)
        Retorno = True
    End If
    Return Retorno

End Function
```

17) Agora, acesse o código do clique do botão `Btn_Transferir_Gabriela`, chame o método de transferência e efetue a mesma, caso seja possível:

```
Private Sub Btn_Transferir_Gabriela_Click(sender As Object, e As EventArgs) Handles Btn_Transferir_Gabriela.

    Txt_Resultado_Gabriela.Text = ""
    Txt_Saldo_Gabriela.Text = ""

    Dim ValorTransferir As Double = Val(Txt_Valor_Gabriela.Text)
    Dim RetornoTransferencia As Boolean = ContaDaGabriela.Transferir(ValorTransferir, ContaDoBruno)

    If RetornoTransferencia = False Then
        Txt_Resultado_Gabriela.Text = "Transferencia não é possível ser feito"
    Else
        Txt_Saldo_Gabriela.Text = ContaDaGabriela.Saldo.ToString
        Txt_Saldo_Bruno.Text = ContaDoBruno.Saldo.ToString

        Txt_Resultado_Gabriela.Text = "Transferencia efetuada com sucesso"
        Txt_SaldoAtual_Gabriela.Text = Txt_Saldo_Gabriela.Text
        Txt_SaldoAtual_Bruno.Text = Txt_Saldo_Bruno.Text
    End If

End Sub
```

```

ContaDaGabriela.Extrato += Now.ToString + " Transferencia de " + ValorTransferir.ToString +
    " Saldo " + ContaDaGabriela.Saldo.ToString + vbCrLf
ContaDoBruno.Extrato += Now.ToString + " Transferencia de " + ValorTransferir.ToString +
    " Saldo " + ContaDoBruno.Saldo.ToString + vbCrLf

Txt_Extrato_Gabriela.Text = ContaDaGabriela.Extrato
Txt_Extrato_Bruno.Text = ContaDoBruno.Extrato
End If

End Sub

```

18) Faça o mesmo para o botão de transferência do Bruno:

```

Private Sub Btn_Transferir_Bruno_Click(sender As Object, e As EventArgs) Handles Btn_Transferir_Bruno
    Txt_Resultado_Bruno.Text = ""
    Txt_Saldo_Bruno.Text = ""

    Dim ValorTransferir As Double = Val(Txt_Valor_Bruno.Text)
    Dim RetornoTransferencia As Boolean = ContaDoBruno.Transferir(ValorTransferir, ContaDaGabriela)

    If RetornoTransferencia = False Then
        Txt_Resultado_Bruno.Text = "Transferencia não é possível ser feito"
    Else
        Txt_Saldo_Gabriela.Text = ContaDaGabriela.Saldo.ToString
        Txt_Saldo_Bruno.Text = ContaDoBruno.Saldo.ToString

        Txt_Resultado_Bruno.Text = "Transferencia efetuada com sucesso"
        Txt_SaldoAtual_Gabriela.Text = Txt_Saldo_Gabriela.Text
        Txt_SaldoAtual_Bruno.Text = Txt_Saldo_Bruno.Text

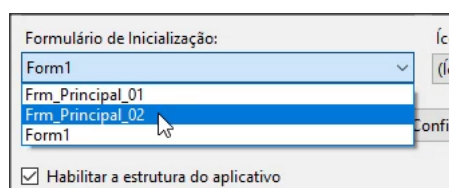
        ContaDaGabriela.Extrato += Now.ToString + " Transferencia de " + ValorTransferir.ToString +
            " Saldo " + ContaDaGabriela.Saldo.ToString + vbCrLf
        ContaDoBruno.Extrato += Now.ToString + " Transferencia de " + ValorTransferir.ToString +
            " Saldo " + ContaDoBruno.Saldo.ToString + vbCrLf

        Txt_Extrato_Gabriela.Text = ContaDaGabriela.Extrato
        Txt_Extrato_Bruno.Text = ContaDoBruno.Extrato
    End If

End Sub

```

19) Para executar é o **formulário 2** como padrão, clique com o botão da direita do mouse sobre o nome do projeto e escolha a opção **Propriedades**. Em **Formulário de Inicialização**, escolha o **Frm_Principal_02** como o padrão do projeto:



20) Salve o projeto, execute e verifique se a transferência está sendo feita corretamente:

21) O código completo da classe `ContaCorrente` ficará assim:

```
Public Class ContaCorrente
```

```
    Public Titular As String
    Public Agencia As Integer
    Public Conta As Integer
    Public Saldo As Double = 100
    Public Extrato As String = ""
```

```
    Public Function Sacar(ValorSacar As Double) As Boolean
```

```
        Dim Retorno As Boolean
        If Saldo < ValorSacar Then
            Retorno = False
        Else
            Saldo -= ValorSacar
            Retorno = True
        End If
        Return Retorno
```

```
End Function
```

```
    Public Function Transferir(ValorTransferencia As Double, ByRef ContaDestino As ContaCorrente) As
```

```
        Dim Retorno As Boolean
        If Saldo < ValorTransferencia Then
            Retorno = False
        Else
            Saldo -= ValorTransferencia
            ContaDestino.Depositar(ValorTransferencia)
            Retorno = True
        End If
        Return Retorno
```

```
End Function
```

```
    Public Sub Depositar(ValorDepositar As Double)
        Saldo += ValorDepositar
    End Sub
```

End Class

22) E o código completo do formulário `Frm_Principal_02` ficará assim:

```
Public Class Frm_Principal_02

    Dim ContaDaGabriela As New ContaCorrente
    Dim ContaDoBruno As New ContaCorrente

    Public Sub New()

        ' Esta chamada é requerida pelo designer.
        InitializeComponent()

        ' Adicione qualquer inicialização após a chamada InitializeComponent().

        Me.Text = "1 - Primeira Classe - Formulário 02"
        Lbl_Principal_02.Text = "1 - Primeira Classe - Formulário 02"

        ' Inicialização dos textos para Gabriela

        Lbl_Valor_Gabriela.Text = "Valor a ser depositado/sacado"
        Btn_Sacar_Gabriela.Text = "Efetuar Saque"
        Btn_Depositar_Gabriela.Text = "Efetuar Depósito"
        Lbl_Saldo_Gabriela.Text = "Novo Saldo"
        Lbl_Resultado_Gabriela.Text = "Resultado do saque"
        Lbl_SaldoAtual_Gabriela.Text = "Saldo Atual"
        Grp_Gabriela.Text = "Gabriela"
        Btn_Transferir_Gabriela.Text = "Efetuar Transferencia"
        Txt_SaldoAtual_Gabriela.ReadOnly = True
        Txt_Saldo_Gabriela.ReadOnly = True
        Txt_Resultado_Gabriela.ReadOnly = True
        Txt_Extrato_Gabriela.ReadOnly = True

        ' Inicialização dos textos para Bruno

        Lbl_Valor_Bruno.Text = "Valor a ser depositado/sacado"
        Btn_Sacar_Bruno.Text = "Efetuar Saque"
        Btn_Depositar_Bruno.Text = "Efetuar Depósito"
        Lbl_Saldo_Bruno.Text = "Novo Saldo"
        Lbl_Resultado_Bruno.Text = "Resultado do saque"
        Lbl_SaldoAtual_Bruno.Text = "Saldo Atual"
        Btn_Transferir_Bruno.Text = "Efetuar Transferencia"
        Grp_Bruno.Text = "Bruno"
        Txt_SaldoAtual_Bruno.ReadOnly = True
        Txt_Saldo_Bruno.ReadOnly = True
        Txt_Resultado_Bruno.ReadOnly = True
        Txt_Extrato_Bruno.ReadOnly = True

        ' Inicialização dos dados da classe da Gabriela

        ContaDaGabriela.Titular = "Gabriela"
        ContaDaGabriela.Agencia = 863
        ContaDaGabriela.Conta = 863141
```

```
Lbl_BemVindo_Gabriela.Text = "Bem Vindo " + ContaDaGabriela.Titular +
    " Agência: " + ContaDaGabriela.Agencia.ToString +
    " Conta Corrente: " + ContaDaGabriela.Conta.ToString
Txt_SaldoAtual_Gabriela.Text = ContaDaGabriela.Saldo.ToString
```

```
' Inicialização dos dados da classe do Bruno
```

```
ContaDoBruno.Titular = "Bruno"
ContaDoBruno.Agencia = 863
ContaDoBruno.Conta = 863224
```

```
Lbl_BemVindo_Bruno.Text = "Bem Vindo " + ContaDoBruno.Titular +
    " Agência: " + ContaDoBruno.Agencia.ToString +
    " Conta Corrente: " + ContaDoBruno.Conta.ToString
Txt_SaldoAtual_Bruno.Text = ContaDoBruno.Saldo.ToString
```

```
End Sub
```

```
Private Sub Btn_Sacar_Gabriela_Click(sender As Object, e As EventArgs) Handles Btn_Sacar_Gabriel
```

```
    Txt_Resultado_Gabriela.Text = ""
    Txt_Saldo_Gabriela.Text = ""
```

```
    Dim ValorSacar As Double = Val(Txt_Valor_Gabriela.Text)
    Dim RetornoSaque As Boolean = ContaDaGabriela.Sacar(ValorSacar)
```

```
    If RetornoSaque = False Then
        Txt_Resultado_Gabriela.Text = "Saque não é possível ser feito"
    Else
        Txt_Saldo_Gabriela.Text = ContaDaGabriela.Saldo.ToString
        Txt_Resultado_Gabriela.Text = "Saque efetuado com sucesso"
        Txt_SaldoAtual_Gabriela.Text = Txt_Saldo_Gabriela.Text
        ContaDaGabriela.Extrato += Now.ToString + " Saque de " + ValorSacar.ToString +
            " Saldo " + ContaDaGabriela.Saldo.ToString + vbCrLf
        Txt_Extrato_Gabriela.Text = ContaDaGabriela.Extrato
    End If
```

```
End Sub
```

```
Private Sub Btn_Depositar_Gabriela_Click(sender As Object, e As EventArgs) Handles Btn_Depositar
```

```
    Txt_Resultado_Gabriela.Text = ""
    Txt_Saldo_Gabriela.Text = ""
```

```
    Dim ValorDepositar As Double = Val(Txt_Valor_Gabriela.Text)
    ContaDaGabriela.Depositar(ValorDepositar)
```

```
    Txt_Saldo_Gabriela.Text = ContaDaGabriela.Saldo.ToString
    Txt_Resultado_Gabriela.Text = "Depósito efetuado com sucesso"
    Txt_SaldoAtual_Gabriela.Text = Txt_Saldo_Gabriela.Text
    ContaDaGabriela.Extrato += Now.ToString + " Depósito de " + ValorDepositar.ToString +
        " Saldo " + ContaDaGabriela.Saldo.ToString + vbCrLf
    Txt_Extrato_Gabriela.Text = ContaDaGabriela.Extrato
```

```
End Sub
```

```
Private Sub Btn_Depositar_Bruno_Click(sender As Object, e As EventArgs) Handles Btn_Depositar_B
```

```

    Txt_Resultado_Bruno.Text = ""
    Txt_Saldo_Bruno.Text = ""

    Dim ValorDepositar As Double = Val(Txt_Valor_Bruno.Text)
    ContaDoBruno.Depositar(ValorDepositar)

    Txt_Saldo_Bruno.Text = ContaDoBruno.Saldo.ToString
    Txt_Resultado_Bruno.Text = "Depósito efetuado com sucesso"
    Txt_SaldoAtual_Bruno.Text = Txt_Saldo_Bruno.Text
    ContaDoBruno.Extrato += Now.ToString + " Depósito de " + ValorDepositar.ToString +
        " Saldo " + ContaDoBruno.Saldo.ToString + vbCrLf
    Txt_Extrato_Bruno.Text = ContaDoBruno.Extrato

End Sub

Private Sub Btn_Sacar_Bruno_Click(sender As Object, e As EventArgs) Handles Btn_Sacar_Bruno.Click

    Txt_Resultado_Bruno.Text = ""
    Txt_Saldo_Bruno.Text = ""

    Dim ValorSacar As Double = Val(Txt_Valor_Bruno.Text)
    Dim RetornoSaque As Boolean = ContaDoBruno.Sacar(ValorSacar)

    If RetornoSaque = False Then
        Txt_Resultado_Bruno.Text = "Saque não é possível ser feito"
    Else
        Txt_Saldo_Bruno.Text = ContaDoBruno.Saldo.ToString
        Txt_Resultado_Bruno.Text = "Saque efetuado com sucesso"
        Txt_SaldoAtual_Bruno.Text = Txt_Saldo_Bruno.Text
        ContaDoBruno.Extrato += Now.ToString + " Saque de " + ValorSacar.ToString +
            " Saldo " + ContaDoBruno.Saldo.ToString + vbCrLf
        Txt_Extrato_Bruno.Text = ContaDoBruno.Extrato
    End If

End Sub

Private Sub Btn_Transferir_Gabriela_Click(sender As Object, e As EventArgs) Handles Btn_Transferir_Gabriela.Click

    Txt_Resultado_Gabriela.Text = ""
    Txt_Saldo_Gabriela.Text = ""

    Dim ValorTransferir As Double = Val(Txt_Valor_Gabriela.Text)
    Dim RetornoTransferencia As Boolean = ContaDaGabriela.Transferir(ValorTransferir, ContaDoBruno)

    If RetornoTransferencia = False Then
        Txt_Resultado_Gabriela.Text = "Transferencia não é possível ser feito"
    Else
        Txt_Saldo_Gabriela.Text = ContaDaGabriela.Saldo.ToString
        Txt_Saldo_Bruno.Text = ContaDoBruno.Saldo.ToString

        Txt_Resultado_Gabriela.Text = "Transferencia efetuada com sucesso"
        Txt_SaldoAtual_Gabriela.Text = Txt_Saldo_Gabriela.Text
        Txt_SaldoAtual_Bruno.Text = Txt_Saldo_Bruno.Text

        ContaDaGabriela.Extrato += Now.ToString + " Transferencia de " + ValorTransferir.ToString +
            " Saldo " + ContaDaGabriela.Saldo.ToString + vbCrLf
        ContaDoBruno.Extrato += Now.ToString + " Transferencia de " + ValorTransferir.ToString
    End If

End Sub

```

```
        " Saldo " + ContaDoBruno.Saldo.ToString + vbCrLf

        Txt_Extrato_Gabriela.Text = ContaDaGabriela.Extrato
        Txt_Extrato_Bruno.Text = ContaDoBruno.Extrato
    End If

End Sub

Private Sub Btn_Transferir_Bruno_Click(sender As Object, e As EventArgs) Handles Btn_Transferir_

    Txt_Resultado_Bruno.Text = ""
    Txt_Saldo_Bruno.Text = ""

    Dim ValorTransferir As Double = Val(Txt_Valor_Bruno.Text)
    Dim RetornoTransferencia As Boolean = ContaDoBruno.Transferir(ValorTransferir, ContaDaGabriela)

    If RetornoTransferencia = False Then
        Txt_Resultado_Bruno.Text = "Transferencia não é possível ser feito"
    Else
        Txt_Saldo_Gabriela.Text = ContaDaGabriela.Saldo.ToString
        Txt_Saldo_Bruno.Text = ContaDoBruno.Saldo.ToString

        Txt_Resultado_Bruno.Text = "Transferencia efetuada com sucesso"
        Txt_SaldoAtual_Gabriela.Text = Txt_Saldo_Gabriela.Text
        Txt_SaldoAtual_Bruno.Text = Txt_Saldo_Bruno.Text

        ContaDaGabriela.Extrato += Now.ToString + " Transferencia de " + ValorTransferir.ToString +
            " Saldo " + ContaDaGabriela.Saldo.ToString + vbCrLf
        ContaDoBruno.Extrato += Now.ToString + " Transferencia de " + ValorTransferir.ToString +
            " Saldo " + ContaDoBruno.Saldo.ToString + vbCrLf

        Txt_Extrato_Gabriela.Text = ContaDaGabriela.Extrato
        Txt_Extrato_Bruno.Text = ContaDoBruno.Extrato
    End If

End Sub

End Class
```