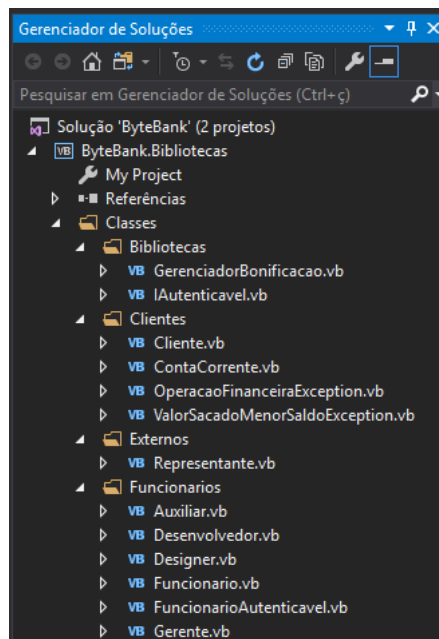


Mãos na massa

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

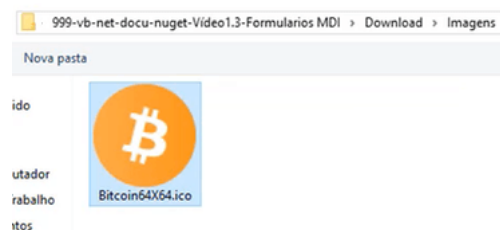
- 1) Crie um novo projeto do tipo **Visual Basic --> Windows Desktop --> Biblioteca de Classes (.NET Framework)**. Dê o nome da solução de **ByteBank** e dê o nome do projeto de **ByteBank.Bibliotecas**.
- 2) Do projeto, remova a `Classe1`. Em seguida, crie a pasta **Classes** na raiz do projeto, e dentro dela, crie a pasta **Funcionarios**.
- 3) Adicione ao projeto as classes desenvolvidas nos cursos anteriores. Caso você não as tenha, faça o download [aqui](https://caelum-online-public.s3.amazonaws.com/999-vb-net-bibliotecas-dll-documentacao-nuget/01/classes.zip) (<https://caelum-online-public.s3.amazonaws.com/999-vb-net-bibliotecas-dll-documentacao-nuget/01/classes.zip>). Clique com o botão da direita do mouse sobre a pasta **Funcionarios** e selecione **Adicionar --> Item Existente**. Navegue até diretório onde foi feito o download, e acesse o diretório **Classes/Funcionarios**, onde existirão seis arquivos. Selecione todos os seis e clique em **Adicionar**.
- 4) Faça o mesmo para as pastas **Bibliotecas**, **Clientes** e **Externos**. Dentro da pasta **Classes**, crie a pasta, clique sobre ela com o botão da direita do mouse e selecione **Adicionar --> Item Existente**. Em seguida, navegue até o diretório onde foi feito o download, acesse a respectiva pasta, selecione todos os arquivos e clique em **Adicionar**.
- 5) Ao final, observe que você terá agora, declaradas no seu projeto **ByteBank.Bibliotecas**, todas as Classes que foram criadas em treinamentos anteriores:



- 6) Crie um novo projeto do tipo **Visual Basic --> Windows Desktop --> Aplicativo do Windows Forms (.NET Framework)**. Dê o nome do projeto de **ByteBank.SistemaAgencia**, mantendo o nome da solução como **ByteBank**.
- 7) Na janela principal, adicione um *Form*, com as seguintes propriedades e valores:

Propriedade	Valor
Name	Frm_Principal
Text	Sistema de Agencias ByteBank
StartPosition	CenterScreen
WindowState	Maximized
Icon	Bitcoin64x64.ico
IsMdiContainer	True

Caso você não tenha, faça o download da imagem [aqui \(https://caelum-online-public.s3.amazonaws.com/999-vb-net-bibliotecas-dll-documentacao-nuget/01/imagens.zip\)](https://caelum-online-public.s3.amazonaws.com/999-vb-net-bibliotecas-dll-documentacao-nuget/01/imagens.zip). Na propriedade `Icon` do formulário, clique no campo com ... ao lado e, a partir do diretório onde você efetuou o download, selecione a imagem:



8) Adicione um *MenuStrip*, de Name `Menu_Principal` ao formulário principal. Nele, inclua as opções abaixo:

- Aula 01
 - Vídeo 01

9) Dê um duplo clique sobre a opção do menu **Vídeo 01** e inclua o código:

```
Private Sub Vídeo01ToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles Vídeo01ToolStri
    Dim F As New Frm_Secundario
    F.MdiParent = Me
    F.Show()
End Sub
```

10) Inclua uma segunda opção de menu, conforme abaixo:

- Janelas
 - Janelas Horizontais
 - Janelas Verticais
 - Janelas em Cascata

11) Dê um duplo clique sobre a opção do menu **Janelas Horizontais** e inclua o código:

```
Private Sub JanelasHorizontaisToolStripMenuItem_Click(sender As Object, e As EventArgs)
    Handles JanelasHorizontaisToolStripMenuItem.Click
    Me.LayoutMdi(MdiLayout.TileHorizontal)
End Sub
```

12) Dê um duplo clique sobre a opção do menu **Janelas Verticais** e inclua o código:

```
Private Sub JanelasVerticaisToolStripMenuItem_Click(sender As Object, e As EventArgs)
    Handles JanelasVerticaisToolStripMenuItem.Click
    Me.LayoutMdi(MdiLayout.TileVertical)
End Sub
```

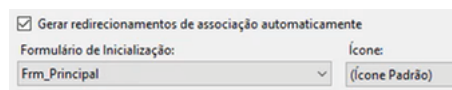
13) Dê um duplo clique sobre a opção do menu **Janelas em Cascata** e inclua o código:

```
Private Sub JanelasCascataToolStripMenuItem_Click(sender As Object, e As EventArgs)
    Handles JanelasCascataToolStripMenuItem.Click
    Me.LayoutMdi(MdiLayout.Cascade)
End Sub
```

14) Vá para o código fonte do formulário, e dentro da sub-rotina `New()`, inicialize a propriedade `Text` do formulário:

```
Me.Text = "Sistema de Agencias ByteBank"
```

15) Defina o formulário `Frm_Principal` como o formulário padrão da aplicação, dando um duplo clique no projeto **ByteBank.SistemaAgencia** e selecionando **Propriedades**. Selecione `Frm_Principal` como padrão, no quadro que se segue:



16) Adicione um novo formulário com as seguintes propriedades e valores:

Propriedade	Valor
Name	Frm_Secundario
Text	Sistema de Agencias ByteBank
StartPosition	CenterScreen
WindowState	Maximized
Icon	Bitcoin64x64.ico

17) No novo formulário, adicione um *PictureBox*, com as seguintes propriedades:

Propriedade	Valor
Name	Pic_Logo
Text	Janela Principal
Image	Bitcoin64x64.png

18) Adicione também um *Label*, de Name `Lbl_ContaCorrenteCriada`, e um *GroupBox*, de Name `Grp_ContaCorrente`. Dentro do *GroupBox*, adicione dois *Label*, de Name `Lbl_Agencia` e `Lbl_ContaCorrente`, respectivamente, dois *TextBox*, de Name `Txt_Agencia` e `Txt_ContaCorrente`, e um *Button*, de Name `Btn_Criar`.

19) Vá para o código fonte do formulário `Frm_Secundario`, e dentro da sub-rotina `New()`, inicialize as propriedades `Text` dos novos componentes:

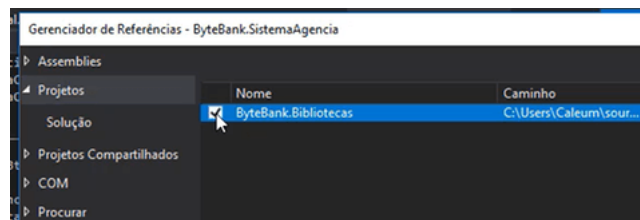
```
Me.Text = "Formulário Secundário"
Grp_ContaCorrente.Text = "Conta Corrente"
Lbl_Agencia.Text = "Agencia"
Lbl_ContaCorrente.Text = "Conta Corrente"
Lbl_ContaCorrenteCriada.Text = ""
Btn_Criar.Text = "Criar"
```

20) Insira o código do clique do botão `Btn_Criar`, conforme abaixo:

```
Private Sub Btn_Criar_Click(sender As Object, e As EventArgs) Handles Btn_Criar.Click
    Dim vAgencia As Integer = Val(Txt_Agencia.Text)
    Dim vContaCorrente As Integer = Val(Txt_ContaCorrente.Text)
    Dim Conta As New ContaCorrente(vAgencia, vContaCorrente)

    Lbl_ContaCorrenteCriada.Text = "Agencia: " + vAgencia.ToString +
        " - Conta: " + vContaCorrente.ToString
End Sub
```

21) Crie no projeto uma referência às bibliotecas definidas no projeto **ByteBank.Bibliotecas**, clicando com o botão da direita do mouse sobre **Referências --> Adicionar Referências --> Projetos --> ByteBank.Bibliotecas**:

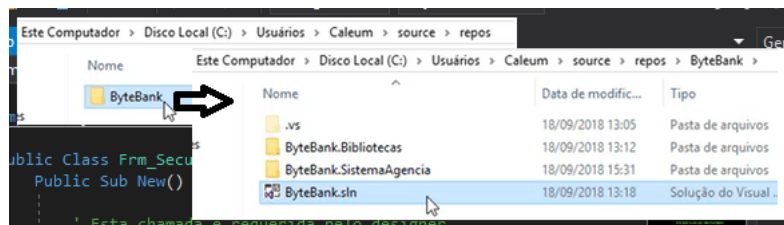


22) Coloque no código a declaração `Imports` :

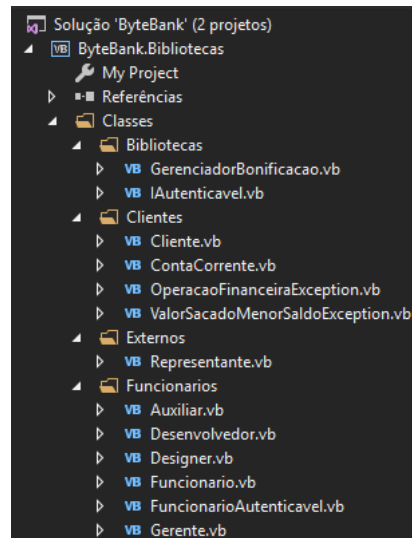
```
Imports ByteBank.Bibliotecas.Classes.Clientes
```

23) Abra o **Explorador de Arquivos** no diretório de repositório da solução e verifique no repositório a existência de um diretório chamado **ByteBank**. Este é o diretório onde estão armazenados todos os arquivos da solução.

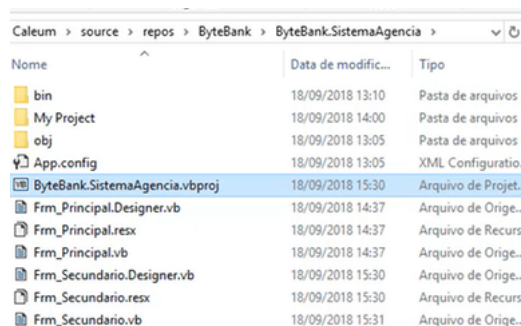
24) Abra o diretório e observe que dentro dele existem dois diretórios e um arquivo. O arquivo é a sua solução. Dê um duplo clique sobre ele e o Visual Studio abrirá com o código da sua solução. Em cada um dos diretórios, estão armazenados os dois projetos que compõem a sua solução, **ByteBank.SistemaAgencia** e **ByteBank.Bibliotecas**:



25) O primeiro diretório contém todas as bibliotecas do projeto, incluindo um diretório com todas as classes definidas na biblioteca. A estrutura de diretórios é montada semelhante à estrutura do projeto dentro da solução:

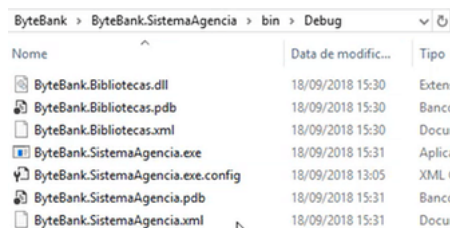


26) Dentro do diretório **ByteBank.SistemaAgencia**, se encontram os arquivos relacionados ao projeto. O arquivo destacado abaixo é o arquivo de projeto em Visual Basic (extensão **.vbproj**):



O arquivo de projeto é um arquivo XML e dentro dele você encontrará *tags* que referenciam DLLs externas ao seu programa, incluindo as referências para o projeto **ByteBank.Bibliotecas**.

27) Abra o diretório **bin/debug**. Aí estão todos os arquivos que fazem parte do sistema **ByteBank**:



28) Dê um duplo clique em **ByteBank.SistemaAgencia.exe**. Observe que o seu sistema executa. Este é arquivo executável de seu sistema **ByteBank.SistemaAgencia**.

Observe também que no diretório existe um arquivo chamado **ByteBank.Bibliotecas.dll**. DLLs são as bibliotecas do Windows. Isto significa que o seu projeto **ByteBank.Bibliotecas**, foi transformado em uma DLL e não em um executável. Isto aconteceu

porque, ao criarmos o projeto, definimos seu tipo como **Biblioteca de Classes**.

29) Uma DLL não é passível de ser executada diretamente pelo Windows, como um arquivo executável. Mas os arquivos executáveis sabem como utilizar as DLLs. Em nosso caso, o executável do **Sistema de Agências** tem como referências as bibliotecas do projeto **ByteBank.Biblioteca**. E as executa sempre que necessário, utilizando o arquivo **ByteBank.Bibliotecas.dll**. Se por algum motivo, esta DLL for deletada, o projeto exibirá um erro sempre que tentar utilizar alguma das classes existentes na biblioteca:

