

## Adicionando o botão e corrigindo detalhes

### Transcrição

Executamos os itens passo a passo, e vimos que tudo funcionou corretamente.

Para melhorar e agilizar o processo de execução dessa macro, criaremos um botão! Na aba "Desenvolvedor > Inserir > Botão", o botão será de *formulário*. Temos uma outra opção para criar um botão. Na aba "Inserir > Formas", escolhemos algum quadrado com as bordas arredondadas, aplicamos um estilo de tema, colocamos o nome de (Calcular Público).

Vamos agora, vincular este botão a uma macro, clicando com o direito, e em "Atribuir macro". O nome da macro é projeto. É preciso apagar os dados de Placar para que possamos ver os resultados após clicar no botão. Vamos executar!

Olhe só. Foi mostrado na tela, vários MsgBox. Durante o teste, colocamos no código para que sempre exibisse a quantidade, mas agora vemos que ele não é mais necessário. Vamos apagá-lo.

```
Sub projeto()
    Dim filme As String
    Dim plan As Object
    Dim linha As Double
    Dim linfinal As Double
    Dim qtde As Double
    Sheets("placar").Select
    Range("d5").Select
    Do Until ActiveCell = ""
        filme = ActiveCell
        For Each plan In Worksheets
            plan.Select
            If plan.Name <> "Placar" Then
                linfinal = Range("d1048576").End(xlUp).Row
                For linha = 5 To linfinal
                    If Cells(linha, 4) = filme Then
                        qtde = qtde + Cells(linha, 5)
                        Exit For
                    End If
                Next
            End If
        Next
        ActiveCell.Offset(0, 1) = qtde
        qtde = 0
        ActiveCell.Offset(1, 0).Select
    Loop
End Sub
```

Pronto! Agora o resultado ficou bem mais legível. O problema é que sempre precisamos jogar a tela para cima, a fim de ver todo o resultado, e isso acaba sendo chato as vezes. Isso acontece porque após a execução, é selecionada a célula D9. E então, vamos pedir para que a macro pare na célula B1.

Isso acontecerá depois que ela executar o Do Until e chegar em uma célula vazia.

```

Sub projeto()
    Dim filme As String
    Dim plan As Object
    Dim linha As Double
    Dim linfinal As Double
    Dim qtde As Double
    Sheets("placar").Select
    Range("d5").Select
    Do Until ActiveCell = ""
        filme = ActiveCell
        For Each plan In Worksheets
            plan.Select
            If plan.Name <> "Placar" Then
                linfinal = Range("d1048576").End(xlUp).Row
                For linha = 5 To linfinal
                    If Cells(linha, 4) = filme Then
                        qtde = qtde + Cells(linha, 5)
                        Exit For
                    End If
                Next
            End If
        Next
        ActiveCell.Offset(0, 1) = qtde
        qtde = 0
        ActiveCell.Offset(1, 0).Select
    Loop
    Range("b2").Select
    MsgBox "Calculo concluído com êxito", vbExclamation
End Sub

```

Assim, a macro irá parar no `B2`, e exibirá a mensagem de "Calculo concluído com êxito". Ao clicar no botão várias vezes, podemos ver que existe um *salto de tela*. Esse salto acontece, porque a macro tem que percorrer várias planilhas e linhas. Dependendo de seu processador, isso pode ser muito grave, causando lentidão ao seu computador.

Então, vamos remover esse salto de tela com a instrução `Application.ScreenUpdating = False`. Esse código "desliga" a atualização de tela, pois ela já vem como `True` por padrão. Entretanto, é preciso colocar a mesma instrução, porém passando `True`.

```

Sub projeto()
    Application.ScreenUpdating = False
    Dim filme As String
    Dim plan As Object
    Dim linha As Double
    Dim linfinal As Double
    Dim qtde As Double
    Sheets("placar").Select
    Range("d5").Select
    Do Until ActiveCell = ""
        filme = ActiveCell
        For Each plan In Worksheets
            plan.Select
            If plan.Name <> "Placar" Then
                linfinal = Range("d1048576").End(xlUp).Row
                For linha = 5 To linfinal

```

```
If Cells(linha, 4) = filme Then
    qtde = qtde + Cells(linha, 5)
    Exit For
End If
Next
End If
Next
ActiveCell.Offset(0, 1) = qtde
qtde = 0
ActiveCell.Offset(1, 0).Select
Loop
Range("b2").Select
MsgBox "Calculo concluído com êxito", vbExclamation
Application.ScreenUpdating = True
End Sub
```

Vamos executar novamente a macro. E como você pode perceber, a tela nem piscou uma única vez!

Quisemos te apresentar esse exemplo, para que você entenda que a questão lógica é muito importante, e às vezes, nos preocupamos em chegar em um resultado concreto. É preciso fazer com calma, e sempre pensar sobre o que você faria caso não existisse o VBA.

O que foi apresentado, foi uma chuva de informações, mas foram informações que aprendemos a manuseá-las nos vídeos anteriores. Utilizamos também, a *indentação* (deslocamento), que permite uma coerência na programação, o código fica mais bonito e legível.

Uma outra boa prática de programação, é criar as variáveis no início, assim como nós fizemos!

A necessidade desse projeto, era trazer para a planilha `Placar` todos os dados de filmes. Muito simples, não é mesmo? Mas, antes de nos preocuparmos com quais laços iremos trabalhar, o que terá de código, etc., temos que nos preocupar em como nós faríamos caso não existisse o VBA. Se fosse uma folha de papel, por exemplo.

Quando paramos e analisamos como o lado humano se comporta, fica muito mais claro. E depois é só colocar a mão na massa!

Esperamos que você tenha absorvido todo o conteúdo, e que você o use no seu dia a dia.