

Introdução

Transcrição

Bem-vindos ao curso de **VBA: Automatizando tarefas no Excel!**

Meu nome é Viviane, e estarei com vocês nesse curso da Alura, tratando sobre **VBA!**

Afinal, o que é VBA?

Vem do termo **Visual Basic For Application**, onde *Visual Basic* é uma linguagem de programação voltada para o uso do Excel!

Você pode se perguntar "*será que vou precisar instalar o VBA na minha máquina?*" Bom, fique tranquilo pois o Excel já traz consigo essa ferramenta, independente da versão.

Quais são as vantagens de se utilizar uma linguagem de programação?

Sabemos que, através dela, podemos unir vocabulários afim de resolver alguma ação ou tarefa. Por exemplo:

- Automação de atividades repetitivas
- Soluções novas através de programação
- Análise de informações
- Ganho de tempo significativo

Então, você pode automatizar tarefas tais como planilhas que é necessário formatar todos os dias, realizar vários cálculos, ou ainda aqueles arquivos que você precisa importar para realizar uma consolidação. Para automatizar, podemos criar *scripts* através do VBA para justamente resolver essas ações.

Também sabemos que o Excel possui milhares de recursos que te ajudam a resolver suas atividades. Entretanto, VBA também irá te trazer soluções inovadoras, não só para automatizar processos, mas também para analisar dados, garantindo a *acurácia* da informação.

Tudo isso te trará um ganho enorme de tempo!

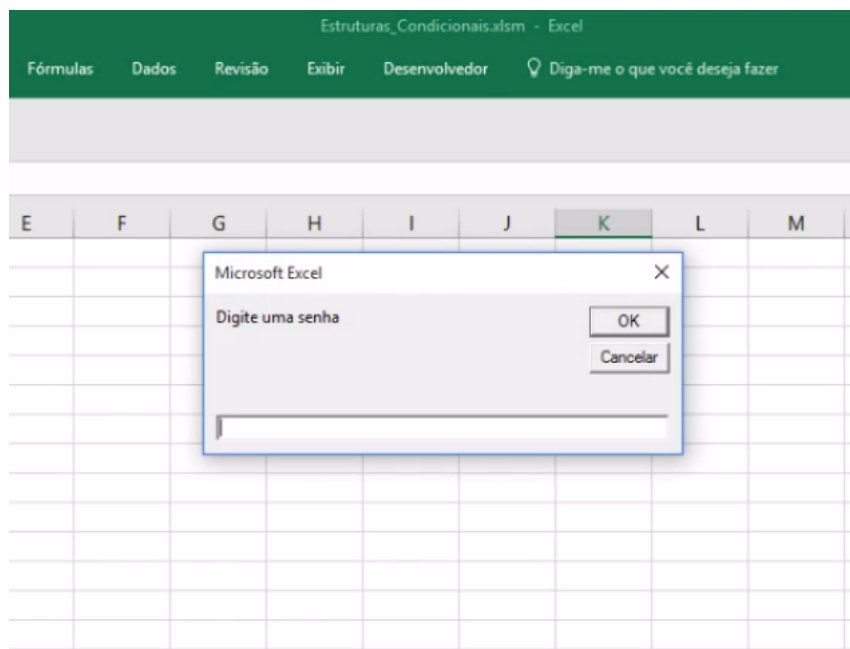
O que aprenderemos nesse curso?

Abordaremos assuntos como - Objetos do Excel e suas nomenclaturas - Como funciona a janela do Visual Basic - Como escrever e executar um *script* em VBA

Tudo isso, com exemplos práticos! Além dos exemplos, vocês também colocarão em prática, todo o conteúdo visto aqui no curso através dos vários exercícios.

Aqui estão alguns dos projetos que serão feitos ao longo do curso.

No primeiro exemplo, ao abrir o arquivo, o Excel pedirá uma senha para o usuário.

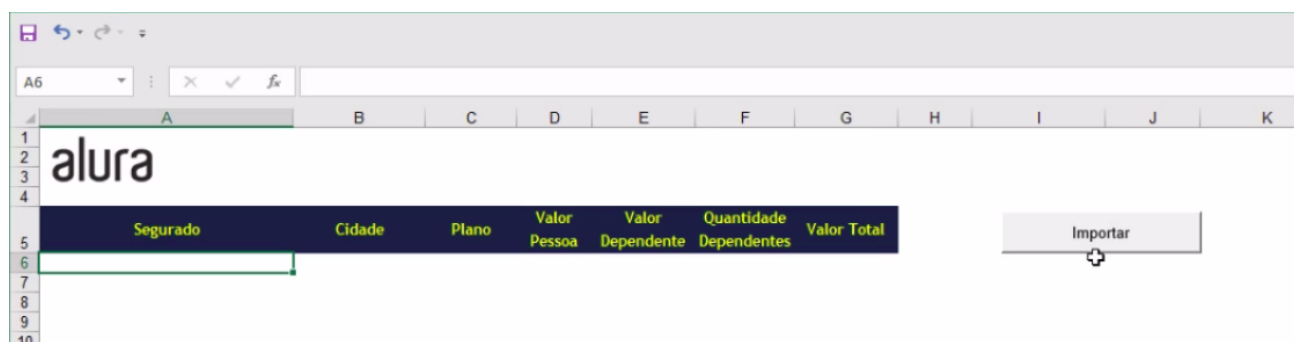


Se errarmos a senha, será mostrado uma mensagem para o usuário de "Senha Inválida", depois será mostrado a mensagem "Até breve!", e então o Excel será fechado. Caso nós acertamos a senha, será exibido as mensagens "Bom dia!", "Boa tarde!" ou "Boa noite!" dependendo do horário em que nós abrirmos o arquivo. Na sequência, teremos acesso às informações da planilha.

Tudo isso é feito através de códigos em *Visual Basic*. A partir do momento que você já estiver familiarizado com os comandos principais do VBA, os vocabulários, variáveis, abordaremos as **instruções de decisão**!

Por exemplo, na planilha do Excel temos a função `if()`, e no VBA temos a instrução `If` para decisões e a instrução `Select Case`.

No segundo projeto, aprenderemos como trabalhar com **arquivos** em VBA! Abordaremos como importar informações de outros arquivos. Para isso, aprenderemos a tratar arquivos com o comando `Workbooks`, e também veremos os comandos `Copy`, `PasteSpecial`, e `.GetOpenFilename`. O método `.GetOpenFilename` é responsável por exibir uma caixa para escolher o arquivo a ser importado:



Esse processo de importação ocorre de uma forma muito ágil!

Um outro recurso importante quando trabalhamos com qualquer tipo de linguagem de programação, são os **laços**!

Os laços otimizam a instrução dos comandos que se repetem, por isso, durante o nosso treinamento, trabalharemos com os conceitos do `While()`, do `Until()` e do `For()`.

```

Sub conceito While()
    Dim n As Byte
    n = 1
    Do While n <= 10 'executar enquanto n <= 10 (Faca enquanto)
        MsgBox n
        n = n + 1
    Loop
End Sub

Sub conceito Until()
    Dim n As Byte
    n = 1
    Do Until n > 10 'PARAR de executar QUANDO n > 10 (Faca até)
        MsgBox n
        n = n + 1
    Loop
End Sub

Sub conceito For()
    Dim n As Byte
    For n = 1 To 10 Step 1 'PARA variavel = <inicio> ate <fim> Com incremento de <
        MsgBox n
    Next
End Sub

```

Todos esses laços serão aplicados em uma planilha do Excel, onde por exemplo, temos que selecionar o tipo de cliente por cidade. Digitaremos a cidade de "São Paulo":

The screenshot shows an Excel spreadsheet with a table of client data. A VBA MsgBox dialog box is open, titled "Microsoft Excel", with the prompt "Digite a Cidade a ser Consultada". The input field contains "São Paulo", which is circled in red. The dialog box has "OK" and "Cancelar" buttons. The spreadsheet data is as follows:

	Cidade	Plano	Valor Pessoa			
a	São Paulo	Nacional	189,50			
	Guarulhos	Nacional	189,50			
	Osasco	Nacional	189,50			
	São Paulo	Nacional	189,50	50,00	0	189,50
	São Paulo	Nacional	189,50	50,00	3	339,50
	São Paulo	Nacional	189,50	50,00	1	239,50
ro	São Paulo	Nacional	189,50	50,00	0	189,50
	São Paulo	Nacional	189,50	50,00	1	239,50
	São Paulo	Nacional	189,50	50,00	1	239,50
	São Paulo	Nacional	189,50	50,00	1	239,50
i	São Paulo	Nacional	189,50	50,00	1	239,50
	Osasco	Nacional	189,50	50,00	2	289,50

Em seguida, será mostrado o total de clientes em São Paulo, e destacará as linhas que contêm a informação referente à São Paulo. Com isso, trabalharemos a parte lógica dentro da instrução de programação em VBA.

Mostraremos também que, através de simples comandos em VBA, você pode resolver uma tarefa difícil, e padronizar as suas informações. Por exemplo, temos a primeira planilha que está com o zoom de 150%, a segunda planilha está com um zoom menor, e a terceira está com um zoom um pouco maior que a segunda. Esse é um exemplo de **informação não padronizada**.

Observe essas poucas linhas de código:

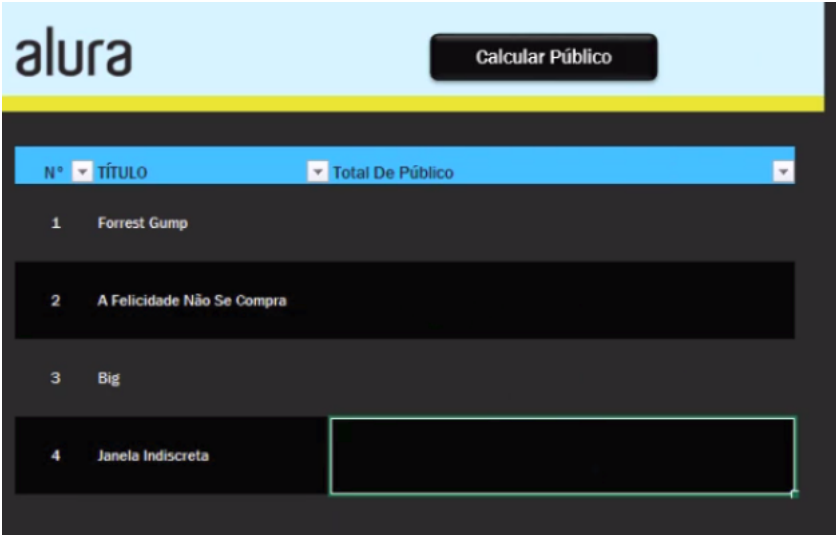
```

Sub ajustar_zoom()
    Dim plan As Object
    For Each plan In Worksheets
        plan.Select
        ActiveWindow.Zoom = 100
        Range("all").Select
    Next
    Sheets(1).Select
End

```

Ao executarmos essas linhas, as nossas planilhas vão estar iguais no zoom, estarão padronizadas.

A ideia agora, é mostrar no placar geral a quantidade de pessoas que assistiu cada um dos filmes mostrados:



The screenshot shows the Alura website interface. At the top, there is a light blue header with the 'alura' logo on the left and a black button labeled 'Calcular Público' on the right. Below the header is a dark grey table with a light blue header row. The header row contains three columns: 'Nº', 'TÍTULO', and 'Total De Público'. The table lists four movies: 'Forrest Gump', 'A Felicidade Não Se Compra', 'Big', and 'Janela Indiscreta'. The 'Total De Público' column for the last movie is highlighted with a green border, indicating it is the target of the VBA activity.

Nº	TÍTULO	Total De Público
1	Forrest Gump	
2	A Felicidade Não Se Compra	
3	Big	
4	Janela Indiscreta	

Vamos rastrear todas as informações e trazer para esse placar geral, clicando no botão "Calcular Público". Com isso, temos um ganho exponencial de tempo! Haverá outros exercícios, mas esses são alguns que faremos juntos no decorrer do curso.

Te espero no próximo vídeo.