

Aprimorando nossa Macro

Transcrição

Continuaremos a desenvolver a nossa macro. Foi colocado alguns comentários no código anterior sobre os comandos que utilizamos:

```
Sub auto_open()
    'Criar variavel info do tipo String
    Dim info As String
    'variavel recebe conteudo digitado
    info = InputBox("Digite uma senha")
    'Se conteudo da variavel for "abc" entao...
    If info = "abc" Then
        MsgBox "Bem vindo ao sistema!"
    Else
        MsgBox "Senha inválida!"
    End If
End Sub
```

Bom, imagine que a pessoa tenha errado a senha. De acordo com o código, será exibido "Senha inválida!" e o usuário continuará no arquivo, ou seja, ele continuará tendo acesso ao arquivo. Mas isso não é muito interessante, não é mesmo?

...O que é ideal?

O ideal é a pessoa errar a senha, exibir "Senha inválida!" e **fechar o Excel**, por exemplo. Assim não vamos permitir que a pessoa tenha acesso ao arquivo. Para fazer isso, utilizaremos o comando `Application.Quit`. Quando usamos `Application`, estamos nos referindo ao Excel por inteiro, e `.Quit` é um evento, uma ação que vai fechar o Excel.

```
Sub auto_open()
    'Criar variavel info do tipo String
    Dim info As String
    'variavel recebe conteudo digitado
    info = InputBox("Digite uma senha")
    'Se conteudo da variavel for "abc" entao...
    If info = "abc" Then
        MsgBox "Bem vindo ao sistema!"
    Else
        MsgBox "Senha inválida!"
        Application.Quit
    End If
End Sub
```

Vamos salvar as alterações, fechar o arquivo e abri-lo novamente, para que nós possamos visualizar a nova condição.

Após o arquivo ser reaberto, colocaremos a senha *correta*, será mostrada a mensagem "Bem vindo ao sistema!". Agora é a vez de reabirmos o arquivo e colocar uma senha *incorrecta*. A mensagem mostrada é a de "Senha inválida!", e após clicarmos em "OK", o Excel se fecha.

Com isso, entendemos que não podemos trabalhar na planilha sem fornecer a **senha correta**. Então, vamos abrir novamente o arquivo que acaba de ser fechado. Mas antes de inserir a senha correta desta vez, perceba que temos a visão da informação quase que por inteira da planilha! Mas será que queremos permitir que a pessoa visualize a informação, mesmo pedindo uma senha para poder acessar esse arquivo?

De repente a informação tem que ser confidencial, onde não queremos que qualquer um possa ter acesso. Então o que podemos fazer nesse caso?

A primeira ideia é *criar uma nova planilha em branco*, e renomeá-la para `Entrada`:



Então, quando o usuário acessar o arquivo, será pedido a senha mas será mostrada a planilha em branco `Entrada`. E assim, somente se o usuário acertar, ele pode ir para a planilha de `Despesas`.

Na macro, para redirecionar para a planilha `Despesas`, após exibir a mensagem "Bem vindo ao sistema!", podemos fazer assim:

```
If info = "abc" Then
    MsgBox "Bem vindo ao sistema!"
    Sheets("despesas").Select
Else
    MsgBox "Senha inválida!"
    Application.Quit
End If
```

`Sheets` é um objeto de planilha, e assim colocamos, entre aspas, o nome da planilha que queremos referenciar. Nesse caso, não importa escrever em maiúsculo ou em minúsculo. Basta estar escrito do mesmo jeito que ela foi criada, com todos os acentos e espaços, se houver.

Ao salvar as alterações estando na planilha `Despesas`, o arquivo será aberto na planilha `Despesas`, e como sabemos, não queremos mostrar essa planilha para a pessoa que colocará a senha futuramente.

Como podemos resolver isso?

Quando o Excel for aberto, ele terá que ser posicionado na planilha `Entrada`. E então, nesta planilha, será pedido a senha para ter o acesso. Então, vamos acessar o VBA. No código, antes de pedir a senha, faremos assim:

```
Sub auto_open()
    'Criar variável info do tipo String
    Dim info As String
    Sheets("entrada").Select
    'variável recebe conteúdo digitado
    info = InputBox("Digite uma senha")
    'Se conteúdo da variável for "abc" então...
    If info = "abc" Then
```

```
    MsgBox "Bem vindo ao sistema!"  
    Sheets("despesas").Select  
Else  
    MsgBox "Senha inválida!"  
    Application.Quit  
End If  
End Sub
```

"Antes de pedir a senha, clique na planilha Entrada"! Vamos salvar e executar o script com o "F8" para que possamos ver o passo a passo.

A primeira ação que o script executou foi `Sheets("entrada").Select`, onde nós fomos redirecionados para essa planilha, e então ele nos pediu a senha. Se digitarmos a senha correta, aparecerá a mensagem, vamos para a planilha de Despesas, e então a macro é encerrada.

Perceba que pequenos detalhes fazem grandes diferenças!

Então, concluímos a navegação entre planilhas, e utilizamos um método novo, que é o `Application.Quit`.