

Criando uma tabela

Transcrição

A nossa missão é criar uma tabela com os seguintes itens:

- Segurado
- Cidade
- Plano
- Valor Pessoa
- Valor Dependente
- Quantidade
- Valor Total

Criaremos essa tabela a partir da célula `A5` na planilha que criamos no vídeo anterior.

Clicamos no módulo **exercícios**, começaremos a digitar na área branca ao lado:

```
Sub Tabela_Simples()
```

```
End Sub
```

A palavra **Sub** indica o *início* da instrução, e automaticamente o **End Sub** é colocado para nós, indicando o *final* da instrução. Depois do **Sub**, colocamos o nome da nossa instrução.

A partir de agora, chamaremos esse conjunto de instruções de **Macro** ou **Script**.

Outras pessoas costumam chamar esse conjunto de **código**, e não tem problema se você usá-lo.

Repare que as *palavras reservadas* `Sub` e `End Sub` são escritas em maiúsculas, mesmo se você tentar escrevê-las em minúsculo. Isso acontece por causa das boas práticas do VBA! É importante seguir esses princípios, e aderir às boas práticas de código. Nós recomendamos que você escreva tudo em *minúsculo* ou que as primeiras letras estejam em *maiúsculo*, assim seu script será mais legível.

Primeiramente, a nossa primeira tarefa é chegar até a célula `A5` no Excel e escrever alguma coisa!

Tudo o que escrevermos entre o **Sub** e o **End Sub**, deslocaremos para a *direita*, como se fosse um parágrafo. Esta também é uma das boas práticas de código, fazendo com que fique mais fácil, mais legível de se ler. Chamamos esse deslocamento de **indentação**, e a melhor maneira de indentar um código é fazendo o uso da tecla **Tab**, (localizada ao lado esquerdo da letra `Q`).

Para indicar uma célula ao VBA, utilizamos o objeto de intervalo **Range**:

```
Sub Tabela_Simples()
```

```
    Range("a5")
```

```
End Sub
```

Como podemos ver, o `Range` é uma propriedade que retorna um intervalo da planilha ativa. Para realizar a alteração da célula, utilizamos o comando **Value**:

`Value` é um comando que nos permite alterar ou atribuir o valor em uma célula. No caso, colocaremos a palavra "Segurado".

```
Sub Tabela_Simples()  
    Range("a5").Value = "Segurado"  
End Sub
```

Relembrando: `Range("a5")` é o que chamamos de objeto, se refere à célula. `Value` altera ou atribui valor à célula.

Vamos seguir com o nosso script:

```
Sub Tabela_Simples()  
    Range("a5").Value = "Segurado"  
    Range("b5").Value = "Cidade"  
End Sub
```

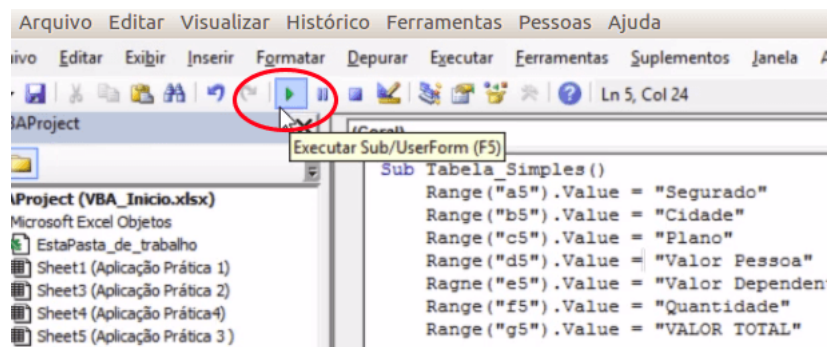
Quando usamos o `.` (ponto), é mostrado uma lista de comando que podemos utilizar. Podemos navegar por essa lista através das "setas para cima e para baixo" do teclado.

```
Sub Tabela_Simples()  
    Range("a5").Value = "Segurado"  
    Range("b5").Value = "Cidade"  
    Range("c5").Value = "Plano"  
    Range("d5").Value = "Valor Pessoa"  
    Range("e5").Value = "Valor Dependente"  
    Range("f5").Value = "Quantidade"  
    Range("g5").Value = "VALOR TOTAL"  
End Sub
```

O símbolo `=` vamos ler sempre como **recebe**, pois a tal célula irá receber tal valor. Com essa prática será mais fácil a interpretação.

Como vamos verificar se esses títulos das tabelas estão certos e que vão realmente funcionar?

Dentro da janela do VBA, temos uma **seta verde** localizada no menu acima.

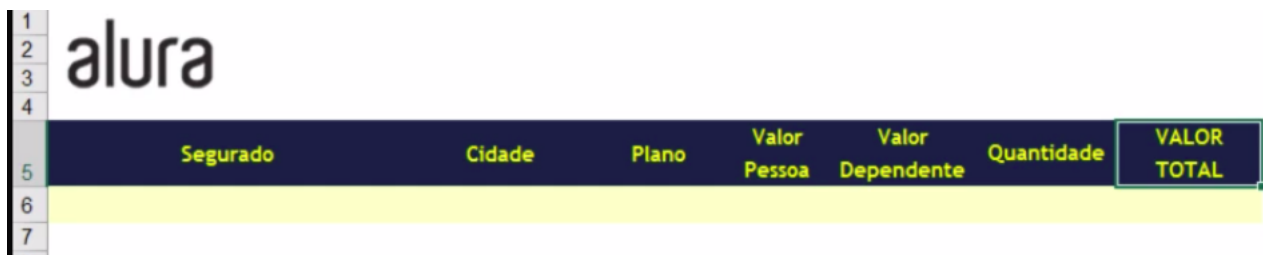


Após clicar no botão verde para executar o script, será lido todas as linhas de código e serão criadas as informações na planilha.

E se tivéssemos escrito alguma coisa errada?

Bom, nesse caso, o VBA irá mostrar uma mensagem de erro de compilação, e ele irá destacar a palavra para nós. Podemos perceber também, que a linha que contém o `Sub` ficará em amarelo. Isso quer dizer que ele tentou executar a macro, tentou entrar nela, mas acabou encontrando um erro.

Depois de ter arrumado, podemos executar novamente, clicando na seta verde. Não aconteceu nada, não foi apresentado nenhum erro, e também não fomos avisados se o script acabou de ser executado. Diante dessa situação, temos que ir até a planilha aberta do Excel, para verificar o resultado.



1	2	3	4	5	6	7	8
				Segurado	Cidade	Plano	Valor Pessoa
							Valor Dependente
							Quantidade
							VALOR TOTAL

Podemos ver que o `VALOR TOTAL` veio em maiúsculo, mas nós não queríamos isso. Você pode pensar dessa maneira: *"Vamos dar Ctrl + Z para cancelar o que eu fiz"*. Você pode até tentar, mas isso não vai resolver, ele não será executado. Isso acontece porque não dá mais para cancelar uma macro após ser executada! Por isso, tome muito cuidado.

Vamos arrumar esse valor diretamente no script para `Range("g5").Value = "Valor Total"`.

Também temos a opção de aumentar o tamanho dessa fonte de letra, assim ficará melhor para visualizar. No menu superior, clique em "Ferramentas > Opções > Formato do editor". Vamos escolher a fonte de tamanho 18.

Uma outra forma de executar o script, é usar a tecla **F5**! Vamos usá-la agora para realizar a última alteração. Como podemos ver, não há indícios de que realmente o script foi executado. Temos que ir até a planilha para conferir o resultado. E então, encontramos o `Valor Total` escrito da forma que queríamos.

Concluimos a primeira parte da tabela, que são os **títulos**, mas agora nós queremos cadastrar uma pessoa nessa tabela. Então, vamos informar os dados em nosso script. Mas para isso ocorrer bem e sem erros de compilação, precisamos usar métodos e recursos do VBA.

Colocaremos um comentário antes, desta forma:

```
Sub Tabela_Simples()  
    Range("a5").Value = "Segurado"  
    Range("b5").Value = "Cidade"  
    Range("c5").Value = "Plano"  
    Range("d5").Value = "Valor Pessoa"  
    Range("e5").Value = "Valor Dependente"  
    Range("f5").Value = "Quantidade"  
    Range("g5").Value = "VALOR TOTAL"  
    'Efetuar o cadastro do cliente  
    'inserir informações no objeto Célula
```

End Sub

O apóstrofo indica uma linha de comentário. Podemos colocá-los onde quisermos.

E se caso você queira colocar uma linha grande de comentário, que precisa ser quebrada em duas linhas? Nesse caso, começaremos o comentário com o ' (apóstrofo) e no final da linha, daremos um (espaço + *underline*).

```
Sub Tabela_Simples()
    'Criar o título da tabela com a utilização _
    de comandos em VBA
    Range("a5").Value = "Segurado"
    Range("b5").Value = "Cidade"
    Range("c5").Value = "Plano"
    Range("d5").Value = "Valor Pessoa"
    'outros ranges...
```

Seguindo as boas práticas de programação, os comentários podem vir sempre antes da linha de código, ou na mesma linha.

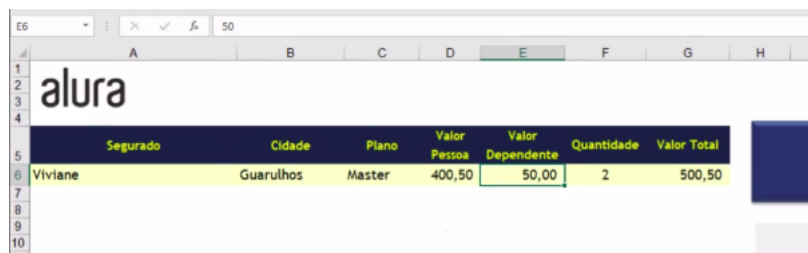
Agora vamos inserir as informações:

```
Sub Tabela_Simples()
    'Criar o título da tabela com a utilização _
    de comandos em VBA
    Range("a5").Value = "Segurado"
    Range("b5").Value = "Cidade"
    Range("c5").Value = "Plano"
    Range("d5").Value = "Valor Pessoa"
    Range("e5").Value = "Valor Dependente"
    Range("f5").Value = "Quantidade"
    Range("g5").Value = "VALOR TOTAL"
    'Efetuar o cadastro do cliente
    'inserir informações no objeto Célula
    Range("a6").Value = "Viviane"
    Range("b6").Value = "Guarulhos"
    Range("c6").Value = "Master"
    Range("d6").Value = 400.5
```

Caso o Valor Pessoa tenha casas decimais, usaremos o . (ponto), pois a padronização de separador americano é o ponto ao invés da vírgula. Vamos continuar.

```
Sub Tabela_Simples()
    'títulos da tabela
    'inserindo informações
    Range("a6").Value = "Viviane"
    Range("b6").Value = "Guarulhos"
    Range("c6").Value = "Master"
    Range("d6").Value = 400.5
    Range("e6").Value = 50
    Range("f6").Value = 2
    Range("g6").Value = 500.5
End Sub
```

Agora, vamos exibir esse código. Clicando em alguma das linhas do nosso script, teclaremos a tecla de função "F5".



Segurado	Cidade	Plano	Valor Pessoa	Valor Dependente	Quantidade	Valor Total
Viviane	Guarulhos	Master	400,50	50,00	2	500,50

Foi trazido todos os valores que cadastramos, não apenas somente o título, mas também o valor que inserimos no cadastro. Perceba que, os separadores de sinal que colocamos, foram trocados automaticamente para **vírgulas**.

No próximo video, teremos mais informações para te falar. Até já!