

# Visual Basic: Conceitos Importantes

Bruno Marcelino

16 de setembro de 2023

## Resumo

Este é um resumo de como utilizar algumas ferramentas do Visual Basic for Applications. As informações serão atualizadas assim que possível. Para visualizar a guia do VBA, deve-se primeiramente habilitar a guia de Desenvolvedor em Arquivo > Opções > Personalizar Faixa de Opções > Guia do Desenvolvedor.

Os scripts de VBA são salvos em "módulos", que funcionam como pastas. Pode-se exportar e salvar os módulos para uso posterior. Módulos de classe permitem que o usuário crie seus próprios objetos.

Para salvar as macros criadas, devemos salvar a planilha como "Pasta de Trabalho Habilitada para Macro".

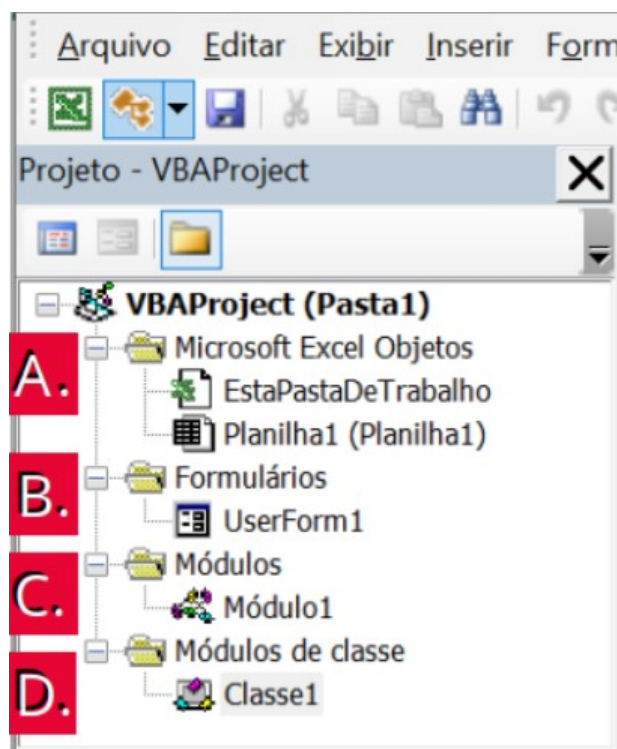


Figura 1: Interface Básica do VBA

# 1 Paradigma

- **Objetos:** são as estruturas de dados que existem no Excel, com as quais podemos interagir. Ex: Células, Planilhas etc. Ex: Range("A1").
- **Atributos ou Propriedades:** são as características de determinado objeto. Ex: Range("A1").Value
- **Métodos:** são funções aplicáveis somente àquele objeto Range("A1").Clear
- **Parâmetros:** são os parâmetros a serem designados aos métodos para interagir com os objetos. Ex: ThisWorkbook.PrintOut From:=1, To:=2
- **Eventos:** interação do usuário que executa o código.

# 2 Atalhos

- Alt + F11: Abre a tela do VBA
- F1: Excel VBA Help
- F5: Executa o código inteiro
- F8: Executa o código linha por linha
- Ctrl + Spacebar: Ferramenta de autocompletar códigos
- Ctrl + R: Exibe o Project Explorer, onde estão os arquivos contidos naquele VBA Project

# 3 Operações Lógicas

Executam blocos de código caso uma condição seja verdadeira

Exemplo:

```
If condicao1 Then  
    ...  
ElseIf condicao2 Then  
    ...  
Else  
    ...  
End If
```

## 4 Objetos Importantes

### 4.1 Range

Permite manipular uma célula ou coleção de células em uma planilha. Exemplos:

- `Range("A1")`
- `Range("B2:D3")`
- `Range("B:B")`
- `Range("Célula Nomeada")`

**Observação.:** o método `Offset` é muito interessante para a manipulação de planilhas, pois permite que o usuário localize uma célula com base na célula ativa. Funciona como um plano cartesiano orientado pelo segundo quadrante, logo números negativos se referem ao lado contrário.

Exemplo: `ActiveCell.Offset(1,1).Select` -> seleciona a célula situada a uma linha e uma coluna de distância.

### 4.2 ActiveCell

Referência Simples à célula ativa. Outros Exemplos: `ThisWorkbook`.

### 4.3 Sheets

Referência simples à uma planilha Exemplo: `Sheets("Plan1")`

## 5 UserForms

São caixas de diálogo que permitem a comunicação com o usuário. Podemos inserir um script nos botões de um formulário por exemplo, permitindo que o usuário execute tarefas.

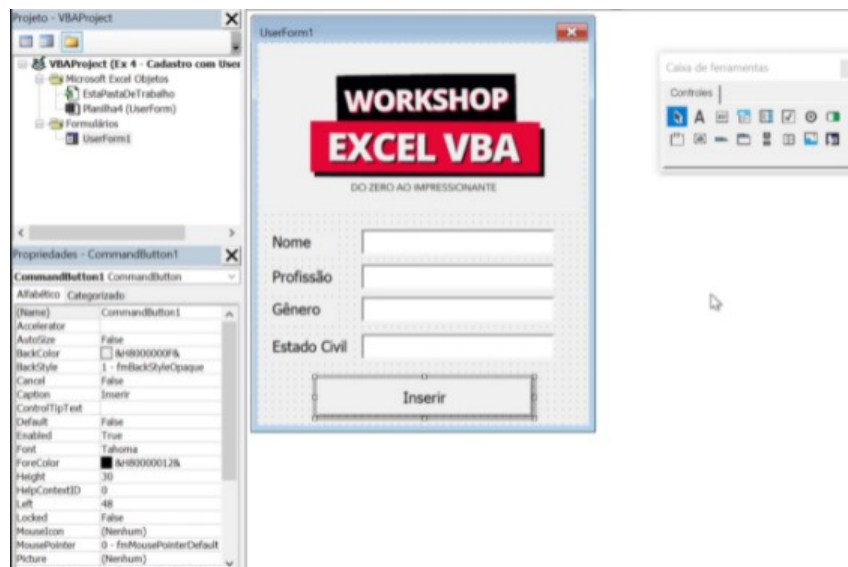


Figura 2: Exemplo da edição de um Formulário

## 6 Loops

Executam um bloco de código enquanto a condição for verdadeira.

Exemplo:

```

Do Until condicao1
    ...
Loop

```