

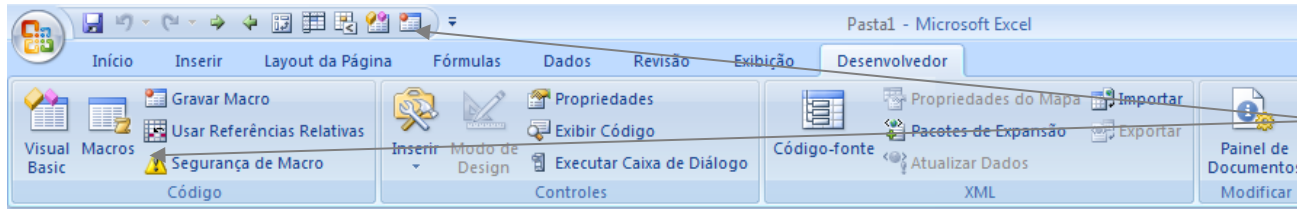
# Programa de Excel para Negócios

**Automatização com VBA**



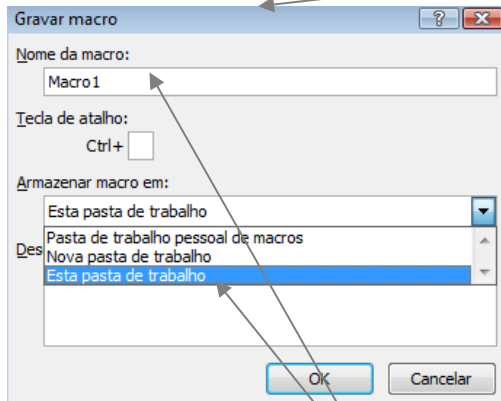
# Gravando macros

Macros são recursos valiosos do Excel, dado que se consegue automatizar diversas rotinas, executando diversos comandos em segundos.

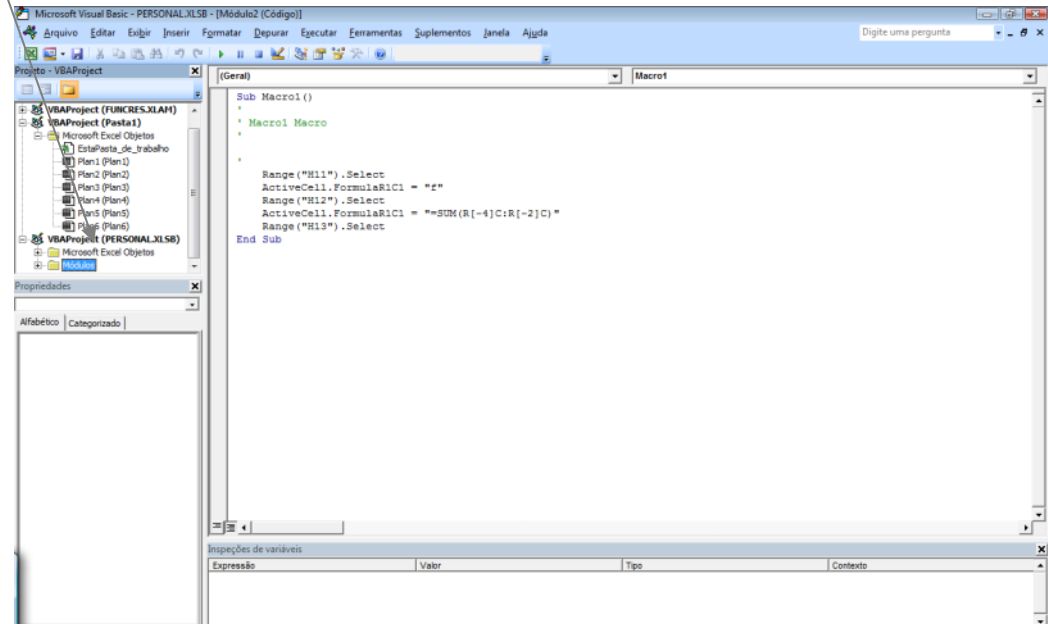


No módulo do desenvolvedor o usuário consegue acessar as funções de gravação de macro, assim como inserir o comando na barra de acesso rápido.

Após ativar o comando de gravação de macro, tudo que é feito no ambiente do Excel será gravado em pastas, que poderão ser acessadas pelo Microsoft Visual Basic.



Escolha um nome e uma pasta para salvar a macro. Ao acessar a "Pasta de trabalho pessoal de macros", a macro desenvolvida poderá ser acessada por qualquer outro arquivo.



## Tipos e declarações

### Principais Tipos

- Integer
- Long
- Double
- String
- Boolean

### Declarações em VBA

- Dim li\_variavel As Integer
- Dim ll\_variavel As Long
- Dim ld\_variavel As Double
- Dim ls\_variavel As String
- Dim lb\_variavel As Boolean

### Sintaxe

### Exemplos

```
li_variavel = 10  
ll_variavel = 10000000  
ld_variavel = 10,5  
ls_variavel = "variável"  
lb_variavel = TRUE
```

## Estruturando funções

### Estrutura

**Function** NomeDaFunção ( aParam1 **As Tipo**, ..., aParamN **As Tipo** ) **As Tipo**

*Aqui vem o corpo da função*

NomeDaFunção = Valor de Retorno

**End Function**

### Exemplo

**Function JurosSimples( aCapitalInicial As Double, aTaxa As Double, aNMeses As Integer ) As Double**

Dim Id\_CapitalAcrescido As Double

Id\_CapitalAcrescido = aCapitalInicial \* (aTaxa/100) \* aNMeses

JurosSimples = aCapitalInicial + Id\_CapitalAcrescido

**End Function**

## Estrutura para criar condicionais pelo VBA

### Estrutura

**If** *Condição* **Then**

Comando 1

Comando N

**Else**

Comando N+1

Comando N+M

**End If**

Existe a possibilidade de utilizar múltiplas condições, basta apenas adicionar o comando **Elseif** *Condição*

### Enunciado

### Exemplo

O cálculo de juros progressivos em um investimento financeiro. Se a conta contém um saldo inferior a R\$ 10.000,00, então o banco paga juros mensais de 2%. Para saldo superiores a R\$10.000,00 o banco paga 3%. A função TaxaProgressiva1 calcula o saldo da conta no final de um mês.

### Solução

**Function TaxaProgressiva1 (aSaldo As Double) As Double**

Dim Id\_Taxa As Double

**If** aSaldo < 10000 **Then**

Id\_Taxa = 0.02

**Else**

Id\_Taxa = 0.03

**End If**

TaxaProgressiva1 = aSaldo \* (1+ Id\_Taxa)

**End Function**

## Criando ciclos de comandos condicionados

### Estrutura

**Do While** *Condição*

Comando 1

Comando N

**Loop**

### Enunciado

### Exemplo

Podemos ter uma taxa de juros inicial de 2% e incrementos de 0.1% mensais: no primeiro mês a taxa de juros será de 2%, no segundo será 2,1%, etc. até o décimo - primeiro mês, quando será de 3%. Um dado capital inicial deverá sofrer juros compostos (acumulados) onde a taxa mensal varia progressivamente.

### Sugestão de resposta

#### Solução

**Function** JurosProgr(aCapIni **As Double**,aTaxaIni **As Double**,aIncrTaxaMensal **As Double**,aNMeses **As Integer**)  
**As Double**

**Dim** li\_Mes **As Integer** *'Número de meses desde o início do investimento*

**Dim** Id\_TaxaAtual **As Double** *'Taxa do mês corrente*

**Dim** Id\_CapAtual **As Double** *'Capital ao final do mês corrente*

Id\_CapAtual = aCapIni

Id\_TaxaAtual = aTaxaIni

li\_Mes = 1

*'Malha principal*

**Do While** li\_Mes <= aNMeses

    Id\_CapAtual = Id\_CapAtual \* (1 + Id\_TaxaAtual )

    li\_Mes = li\_Mes + 1

    Id\_TaxaAtual = Id\_TaxaAtual + aIncrTaxaMensal

**Loop**

JurosProgr = Id\_CapAtual

**End Function**

**Estrutura  
With**

```
With Worksheets("Planilha").sheets("Aba01")  
    .cells(1,3)="Excel2017"  
End With
```

**Estrutura dos  
Ciclos de  
Varredura**

---

```
Dim x as integer, y as integer  
For x = 1 to 10  
    if x=5 then  
        MsgBox("x é igual a 5")  
    end if  
Next x
```



## Macros

### Sub-rotina

```
Sub NomeSub( <Lista de Parâmetros> )  
End Sub
```

### Macro

Sub-rotina que não necessita de parâmetros

### Entrada de Parâmetros

```
Texto = InputBox( "Entre com um valor"
```

### Conversão

```
Dim li_Nprods As Integer
```

```
Dim Id_Prec01 As Double
```

```
li_Nprods = CInt( InputBox( "Entre a quantidade de produtos:" ) )
```

```
Id_Prec01 = Cdbl( InputBox( "Preço do primeiro produto" ) )
```

## Comando

## Definição

**Dim livro as workbook**  
**Set livro = thisworkbook**

### O objeto workbook

*Definição da variável livro como o workbook onde está inserido o módulo - thisworkbook*

**Dim aba as worksheet**  
**Set aba = thisworkbook.sheets("sheet1")**

### O objeto worksheet

*Definição da variável aba como a "sheet1" dentro do workbook onde está inserido o módulo - thisworkbook.sheet1*

**Dim nome as string**  
**Set aba = thisworkbook.sheets("sheet1")**  
**Nome = aba.range("B1")**

### O objeto range

*Após a declaração da variável nome, atribui-se a ela o valor do range thisworkbook.sheets("sheet1").range("B1") (range "B1" na aba "sheet1" do workbook onde está inserido o módulo)*

**Dim nome as string**  
**Set aba = thisworkbook.sheets("sheet1")**  
**nome = aba.cells(linha, coluna)**

### A propriedade cells

*Após a declaração da variável nome, atribui-se a ela o conteúdo da célula(linha, coluna) especificada*

### Exemplos

#### Sub Abas

Dim aba as worksheet

Set aba=thisworkbook.sheets("sheet1")

Thisworkbook.Worksheets("Sheet1").Cells(5, 3) = 14 (ou, simplificando, aba.cells(5,3) = 14)

Thisworkbook.Worksheets("Sheet1").Range("C5") = 14 (ou, simplificando, aba.Range("C5") = 14)

'Ambas os comandos carregam a célula B5 com o número 14

'Cuidado com a sintaxe de construção das instruções

#### End Sub

### Exemplos

#### Sub AbasII

Dim aba as worksheet

With thisworkbook.worksheets("Sheet1")

.Cells(5, 3) = 14 (ou, simplificando, aba.cells(5,3) = 14)

.Range("C5") = 14 (ou, simplificando, aba. .Range("C5") = 14)

End with

'A utilização da estrutura with para simplificação do código

#### End Sub

## Interação com o usuário

### Saída

O comando **MsgBox(String)** mostra o conteúdo do String ao usuário em uma janela. Por exemplo, supondo que a variável MIN é do tipo Double e contém o valor 27,32.

**MsgBox( "O melhor preço é: " & MIN )** faz aparecer uma janela com a mensagem: *O melhor preço é: 27,32*

### Solução

### Exemplo

```
Sub SomaNum()
```

```
    Dim Id_Soma As Double
```

```
    Dim li_Cont As Integer
```

```
    Dim Id_Valor As Double
```

```
    Id_Soma = 0
```

```
    li_Cont = 0
```

```
    Id_Valor = 0
```

```
    Do While li_Cont < 4
```

```
        Id_Valor = Cdbl(InputBox("Entre com o valor"))
```

```
        Id_Soma = Id_Soma + Id_Valor
```

```
        li_Cont = li_Cont + 1
```

```
    Loop
```

```
    MsgBox ("A soma é:" & Id_Soma)
```

```
End Sub
```

## Variáveis com diversas posições

### Declaração de Vetor

- **Dim vetor**(1 To Nposições) **as Variável**
- **Dim vetor**(Nposições) **as Variável**

### Declaração de Matriz

- **Dim matrix**(1 To NLinhas, 1 to MColunas ) **as Variável**
- **Dim matrix**(NLinhas, MColunas) **as Variável**

## Principais comandos e métricas

Ação	Comando
Valor	Valor = Cells(I, J)
Linha	Rows(I)
Coluna	Columns(J)
Range	Range("A1:Z60000")
Manipulando Várias Planilhas	Worksheets("nome da planilha").Cells(I,J)
Mudar planilha de atuação	Worksheets("nome da planilha").Select
Adicionar/Remover planilhas	Sheets.Add / Sheets.Delete

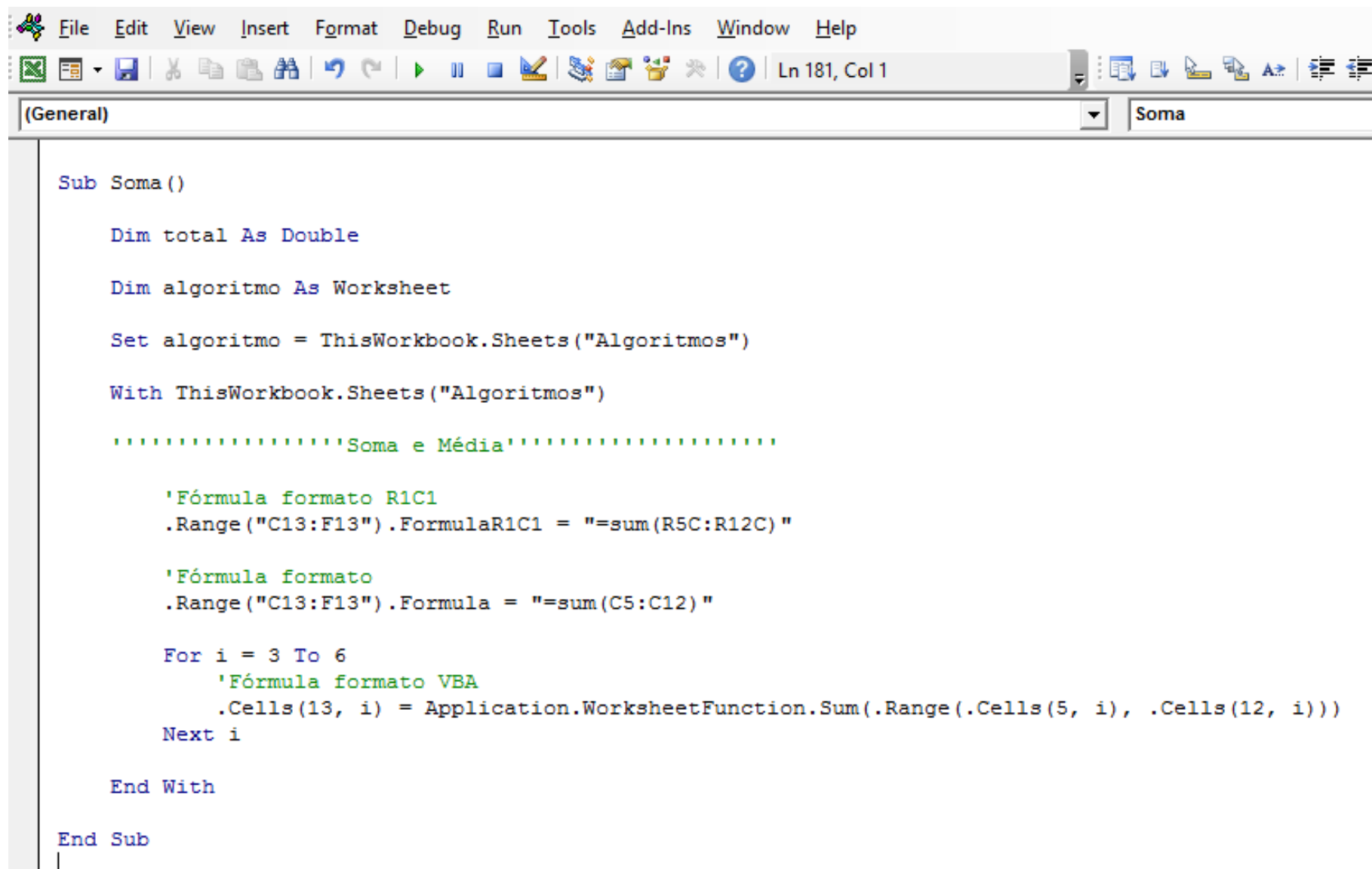
Estruturas	Comandos
A estrutura <code>".FormulaR1C1"</code>	<code>Sheets().range().Formular1c1="=string"</code>
A estrutura <code>".Formula"</code>	<code>Sheets().range().Formula = "=string"</code>
A estrutura <code>".FormulaArray"</code>	<code>Sheets().range().formulaarray = "=string"</code>
Inserindo um código pelo VBA	<code>Sheets().range().=application.worksheetfunction</code>

*Ao carregarmos uma string com o sinal de "=" na frente, o Excel entende tratar-se de uma fórmula, portanto sempre que quisermos inserir uma fórmula na planilha, a partir do VBA, devemos inserir o sinal de "=" antes da construção da fórmula.*

*Uma das maneiras de testarmos a sintaxe da fórmula é primeiro escrevermos a fórmula na planilha e depois, fazendo uso do recurso de gravação de macros, gravarmos a fórmula em um módulo do workbook e utilizarmos a gravação em nosso código. Para fazer isso, com a gravação ativa, devemos pressionar F2 e enter, em seguida parar a gravação.*

*O .FormulaArray é utilizado para carregarmos fórmulas em formato matricial (F2 + Ctrl + Enter)*





The screenshot shows the Microsoft Excel VBA editor interface. The menu bar includes File, Edit, View, Insert, Format, Debug, Run, Tools, Add-Ins, Window, and Help. The toolbar contains various icons for file operations, editing, and execution. The status bar at the bottom indicates 'Ln 181, Col 1'. The active sheet is named 'Soma'. The VBA code is as follows:

```
Sub Soma()  
  
    Dim total As Double  
  
    Dim algoritmo As Worksheet  
  
    Set algoritmo = ThisWorkbook.Sheets("Algoritmos")  
  
    With ThisWorkbook.Sheets("Algoritmos")  
  
        .....  
  
        'Fórmula formato R1C1  
        .Range("C13:F13").FormulaR1C1 = "=sum(R5C:R12C) "  
  
        'Fórmula formato  
        .Range("C13:F13").Formula = "=sum(C5:C12) "  
  
        For i = 3 To 6  
            'Fórmula formato VBA  
            .Cells(13, i) = Application.WorksheetFunction.Sum(.Range(.Cells(5, i), .Cells(12, i)))  
        Next i  
  
    End With  
  
End Sub
```

*O VBA é uma poderosa ferramenta para a automatização de tarefas rotineiras nas empresas.*

*Um uso comum é a utilização da ferramenta para a geração de relatórios, os mais diversos possíveis, dentro dos mais variados setores, tais como financeiro, consultorias, industrial e varejo.*

## Relatórios - Estrutura de módulos

*Em geral, trabalhamos no excel em conjunto com uma base de dados, de onde extraímos os dados a serem trabalhados. Uma das estruturas para que organizemos a geração dos relatórios via VBA é a seguinte:*

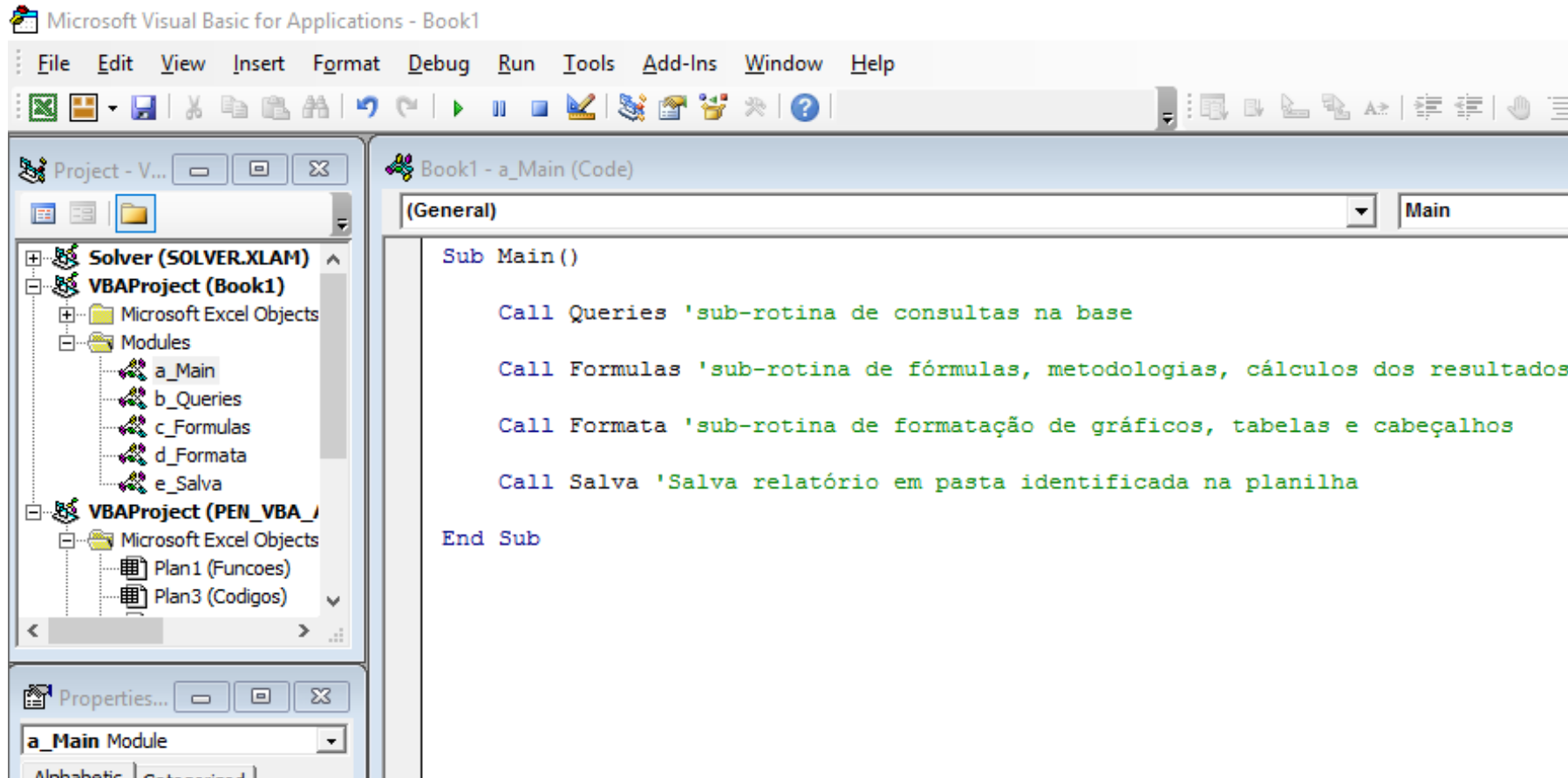
*a\_Main (Módulo Principal) - Chama todas as sub-rotinas componentes do programa*

*b\_Queries (Módulo de Consulta) - Módulo onde estão inseridas as consultas nas bases*

*c\_Formulas (Módulo de Cálculos) - Implementação de fórmulas, cálculos, metodologias*

*d\_Formata (Módulo de Formatação) - Formatação de cabeçalhos, tabelas, gráficos*

*e\_Salva (Módulo de Gravação) - Salva arquivo em pasta definida na planilha*

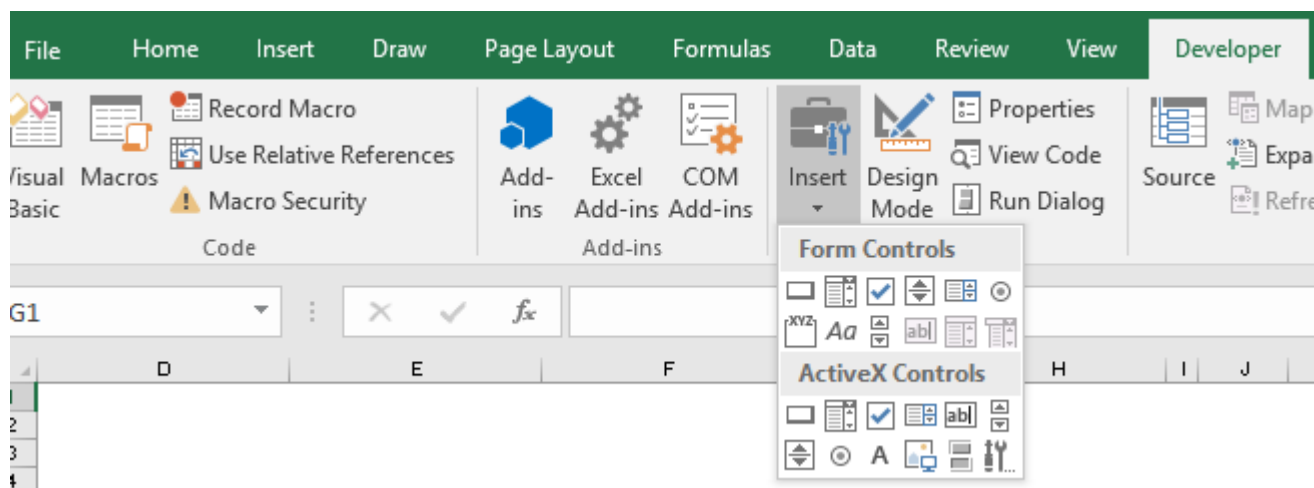


*Uma das maneiras de escrevermos os códigos de formatação e gravação utilizados no relatório é fazendo uso do recurso de gravação de macro.*

*O Excel possibilita a construção de formulários diversos, através dos controlos de formulários. Para disponibilizar a utilização dos controlos, é possível habilitá-los da seguinte forma:*

- *File (Arquivos)*
- *Options (Opções)*
- *Customize Ribbon (Personalizar)*
- *Main Tabs*
- *Check no box Developer*

*Após habilitar o box de desenvolvimento, uma nova aba surge (Developer/Desenvolvedor) - Sub-Menu Controls (Controles) – Insert (Inserir)*



Célula formatada como Label

## Option Button

Label

## Check Box

List Box

## List Box - Multiple Selection

*Podemos usar dois tipos de controles: Form Controls e ActiveX Controls (macro é disparada quando da ocorrência de um evento)*

