

2-INTERFACE

A Interface com o Usuário é fundamental em qualquer aplicação, pois, é a parte visível pelo qual ele se comunica com o usuário para realização das suas tarefas.

O VBA possui modelos prontos de caixas de mensagens, com as quais podemos usá-las para guiar e ajudar o usuário na solução de problemas.

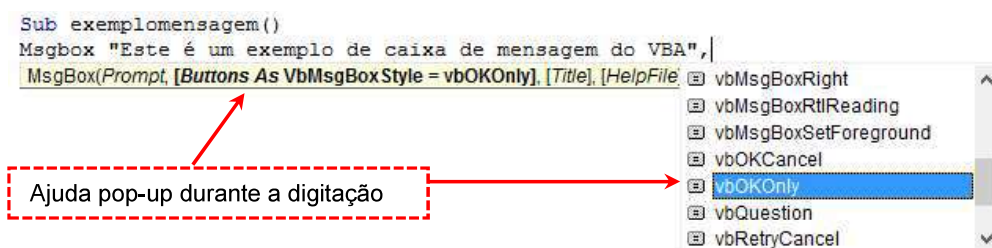
2.1-A Função MsgBox

Essa função é muito utilizada para exibir uma mensagem ao usuário, sendo que sua sintaxe é:

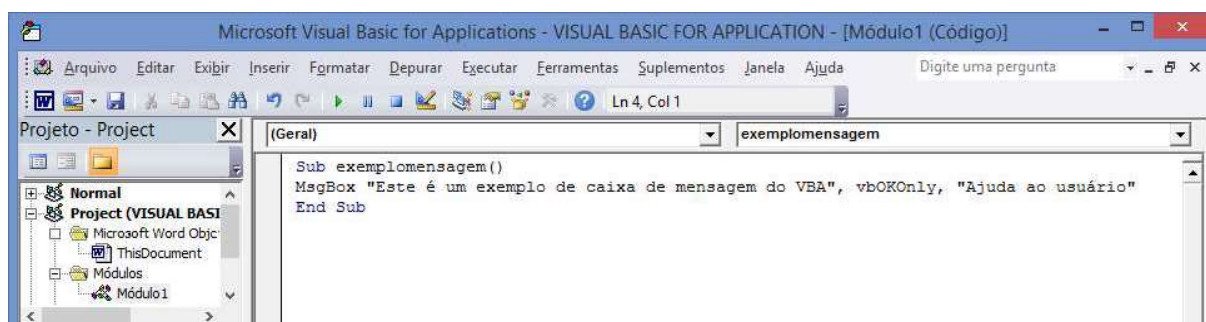
MsgBox “mensagem ao usuário”, tipo de caixa e/ou tipo de botão, “Título da caixa”.

O legal do ambiente de programação do VBA é sua ajuda pop-up que surge no momento que estamos escrevendo os códigos.

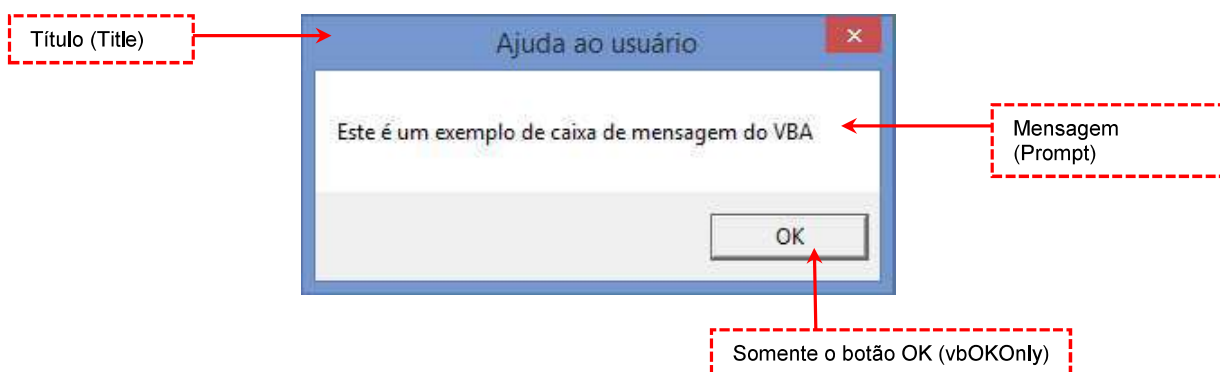
Observe a imagem abaixo:



Veja um exemplo de uma caixa de mensagem usando a função MsgBox, para isso criaremos uma macro chamada **exemplomensagem**.



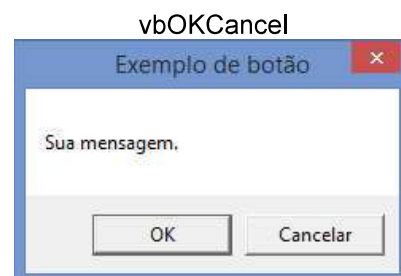
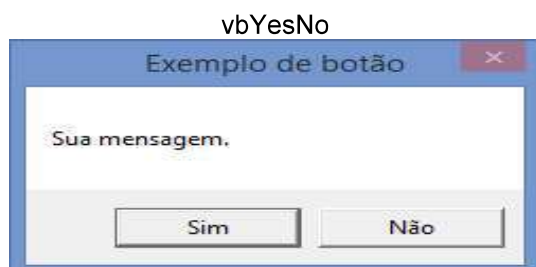
Quando o código acima é executado exibe a seguinte mensagem:



Para facilitar a comunicação com o usuário, o VBA possui botões prontos para sua caixa de mensagem e cada um desses botões tem um número correspondente, bem como um código VBA. Veja na tabela abaixo:

<i>Botão</i>	<i>Valor</i>	<i>Código VBA</i>
OK	1	vbOK
Cancelar	2	vbCancel
Anular	3	vbAbort
Repetir	4	vbRetry
Ignorar	5	vbIgnore
Sim	6	vbYes
Não	7	vbNo

Veja alguns exemplos:



2.2-A Função InputBox

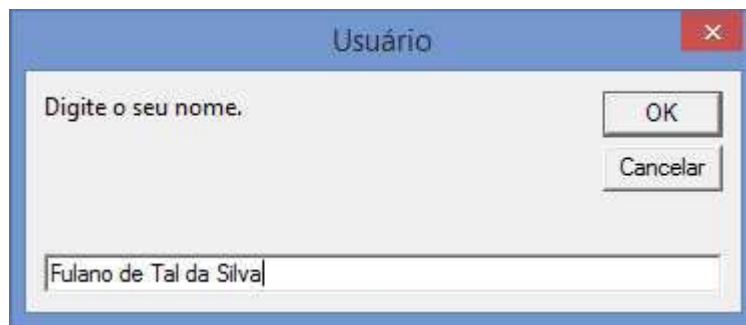
É uma caixa de mensagem mais interativa, pois nela o usuário pode inserir dados. Sua sintaxe é:

InputBox (“mensagem”, “Título”, “Resposta Padrão”, “Posição X”, “Posição Y”)

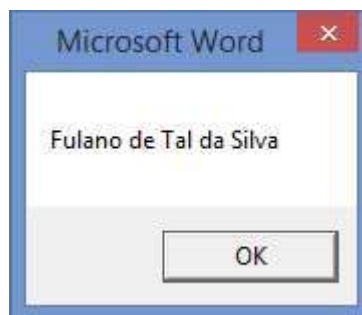
Exemplo:

```
Sub entradadedados()  
  Dim nom As String  
  nom = InputBox("Digite o seu nome.", "Usuário")  
  MsgBox nom  
End Sub
```

Ao executarmos a macro **entradadedados** será solicitado ao usuário que digite o seu nome e em seguida o mesmo é exibido em uma caixa de mensagem.



Ao clicar no botão OK será exibido o nome em uma caixa de mensagem.



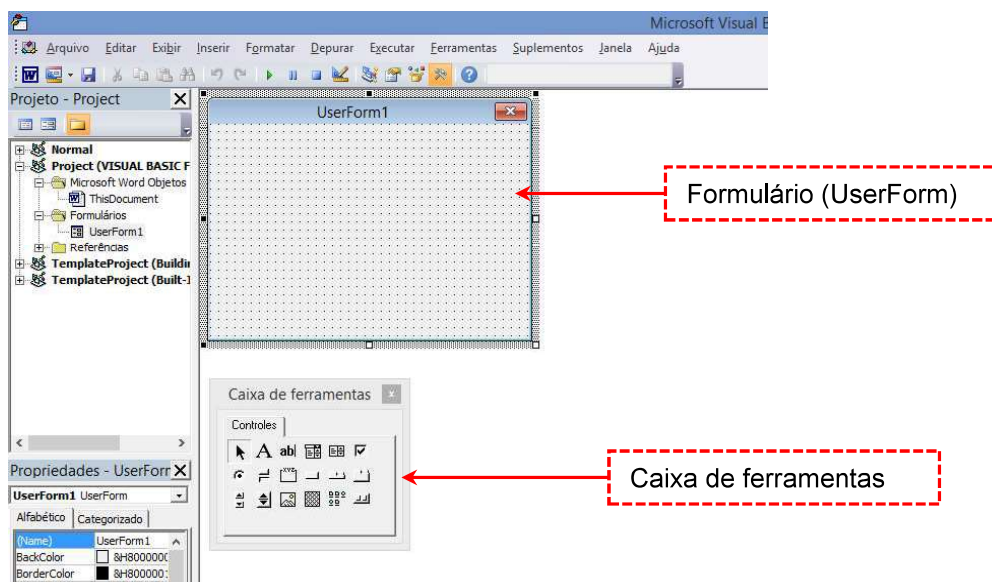
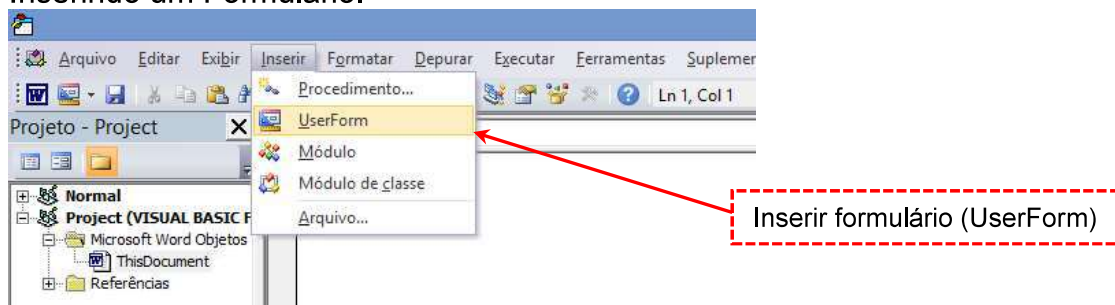
2.3-Criando Formulários

Como neste capítulo estamos falando de interface, não poderíamos prosseguir sem comentar sobre a forma mais interessante de se criar interfaces no VBA que é por meio de formulários.

Um formulário funciona como caixas de mensagem com controles para coletar dados do usuário, enviar dados, mostrar dados, mostrar opções, etc.

Como vimos no Capítulo I, quando inserimos um formulário no ambiente de edição do VBA a caixa de ferramentas é automaticamente exibida. Basta clicar no controle desejado e soltá-lo sobre o formulário para criarmos a interface desejada. Na janela Propriedades podemos alterar o formulário e os controles nele inseridos para deixarmos com o melhor visual possível.

Inserindo um Formulário.



Como foi dito no início desta apostila, o primeiro e o segundo capítulos são apenas uma breve introdução ao VBA. Poderíamos até descrever detalhadamente todas as outras ferramentas do editor, poderíamos nos aprofundar no escopo de variáveis e constantes, nos métodos e propriedades, nas estruturas de programação (estrutura condicional IF, estrutura SELECT CASE, controles de LOOPING, etc), mas acho que é bem mais útil descrever as bases do funcionamento e dar os sedimentos para o leitor ter condições suficientes para seguir sozinho. Além do mais, nos capítulos seguintes apresentarei vários exemplos de macros e soluções que os usuários do Word poderão implementá-los em seu dia a dia e também estarei mostrando passo a passo como criar 10 Projetos utilizando uma linguagem simples, sem os termos técnicos, de maneira que aqueles que são leigos no assunto possam compreender toda lógica envolvida e partir daí se aventurar a criar suas próprias aplicações.

CAPÍTULO - III

MACROS

3.-O QUE É MACRO?

Uma macro nada mais é que uma sequência de comandos e funções armazenados em um módulo de VBA e que pode ser usada sempre que você precisar executar a tarefa, portanto, macros são excelentes recursos para realizar e automatizar tarefas repetitivas e longas.

Muitos tem uma certa "repulsa" ao falarmos em macros, pois envolve a programação VBA (Visual Basic for Applications), porém, o que muita gente não sabe é que o Word possui um gravador de macros para automatizar esta programação, de maneira que não é necessário conhecer a linguagem VBA para criar uma macro no Word, pois, ao utilizar este mecanismo ele grava a sequência de todas as ações enquanto o gravador estiver ligado.

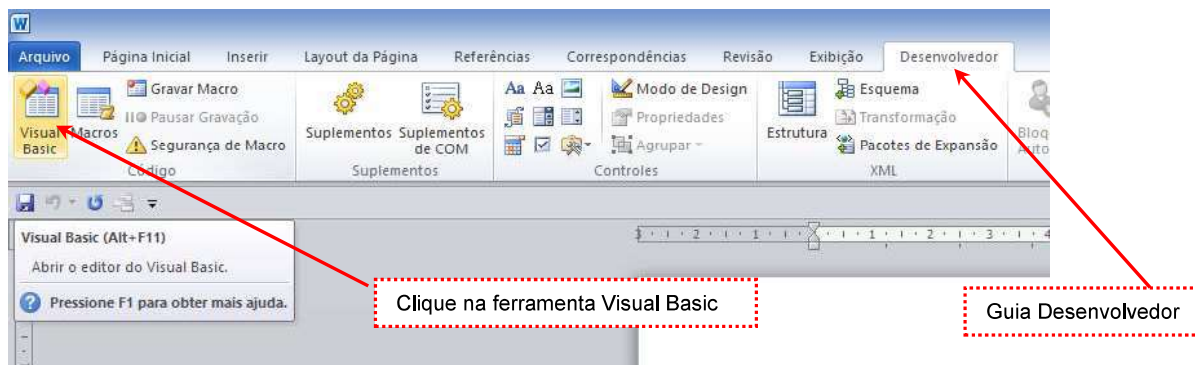
Apesar do gravador de macros ser uma mão na roda para quem quer automatizar tarefas repetitivas, com o tempo torna-se imprescindível aprender e dominar a linguagem VBA para criar macros mais sofisticadas sem a necessidade de gravá-las passo a passo.

A edição a partir dos códigos é feita através do próprio Editor do Visual Basic contido no programa e esta ferramenta que utilizaremos para criar as 15 (quinze) macros que apresentaremos neste capítulo.

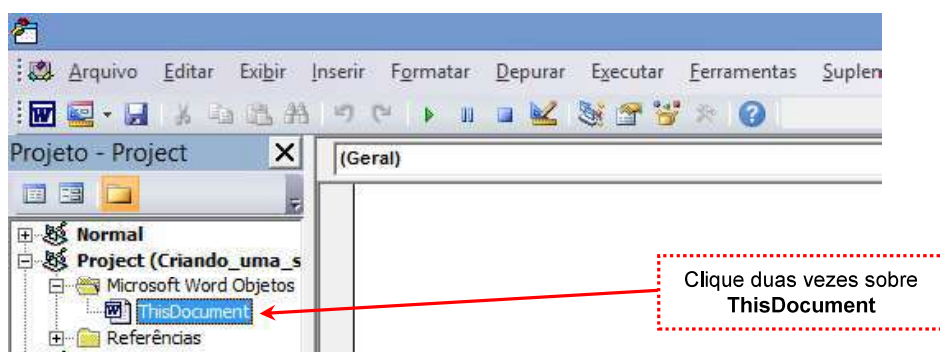
Mas cuidado, preste muita atenção ao nomear uma macro, pois, se você der a uma nova macro o mesmo nome que uma interna do Word, as ações da nova macro substituirão a macro interna. Para exibir uma lista de macros internas, acione a guia **Desenvolvedor** e no grupo **Código** clique em **Macros**. Na lista **Macros em**, clique em **Comandos do Word**.

3.1-Criando uma saudação ao usuário

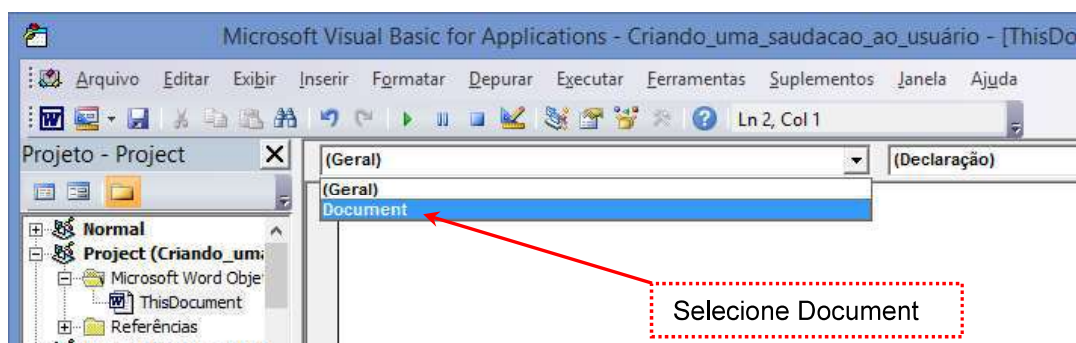
Abra um documento Word existente ou um novo documento. Abra o Editor do Visual Basic (VBE) através da **Guia Desenvolvedor** e no **Grupo Código** clique em **Visual Basic** ou simplesmente através do atalho **Alt + F11**.



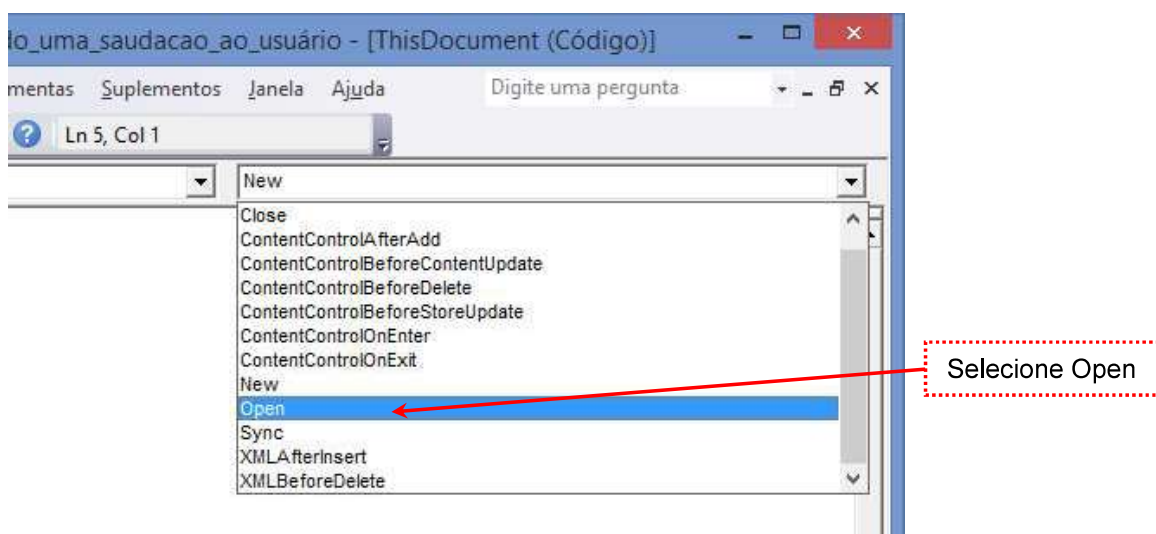
No ambiente do VBA clique duas vezes sobre **"ThisDocument"** que fica na janela **Projeto –Project**.



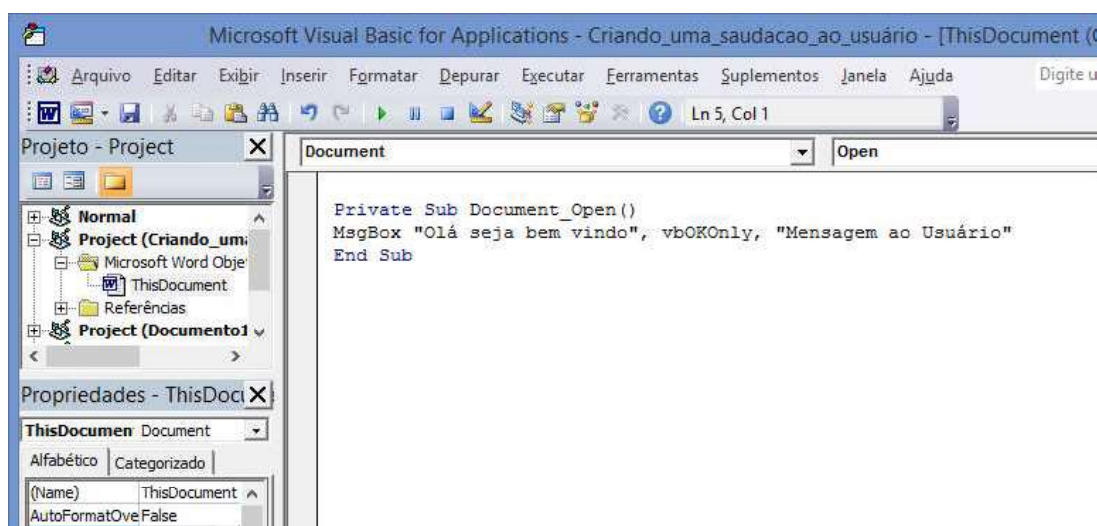
Na **Janela de Código** selecione o objeto **Document**:



Em procedimento selecione **Open**:



Veja como deve ficar na imagem abaixo:

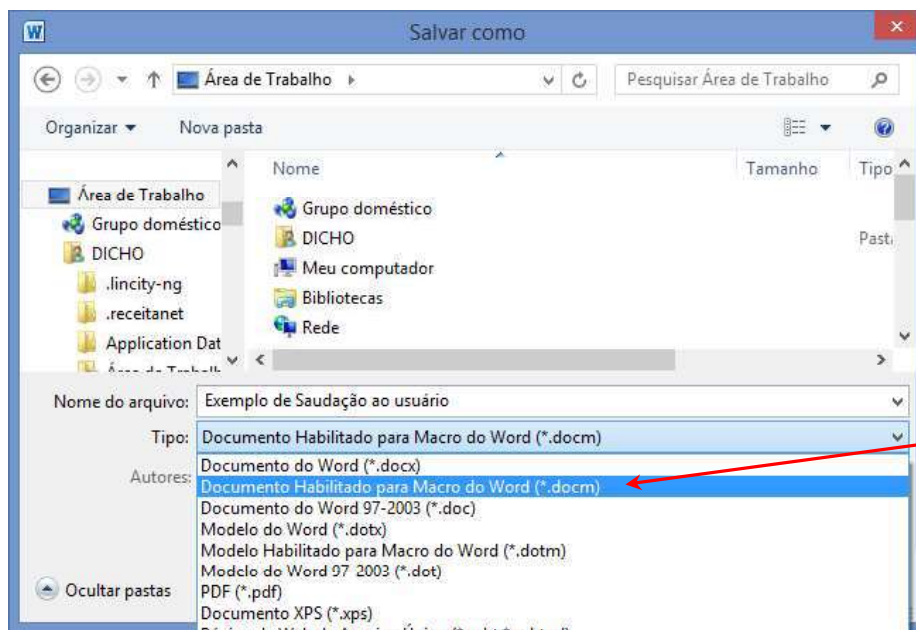


Explicando o Código:

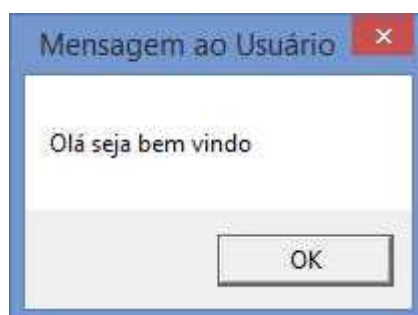
```
Private Sub Document_Open()  
MsgBox "Olá seja bem vindo", vbOKOnly, "Mensagem ao Usuário"  
End Sub
```

Observe que para este exemplo utilizamos o procedimento **Open**, isso significa que toda vez que o documento for aberto será exibida uma mensagem de Boas vindas ao usuário.

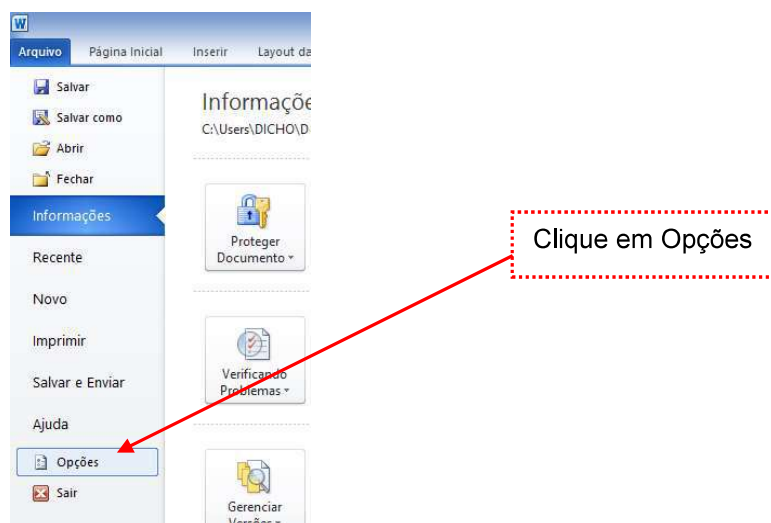
Feche o Editor do Visual Basic e salve o documento como **Documento Habilitado para Macro do Word**.



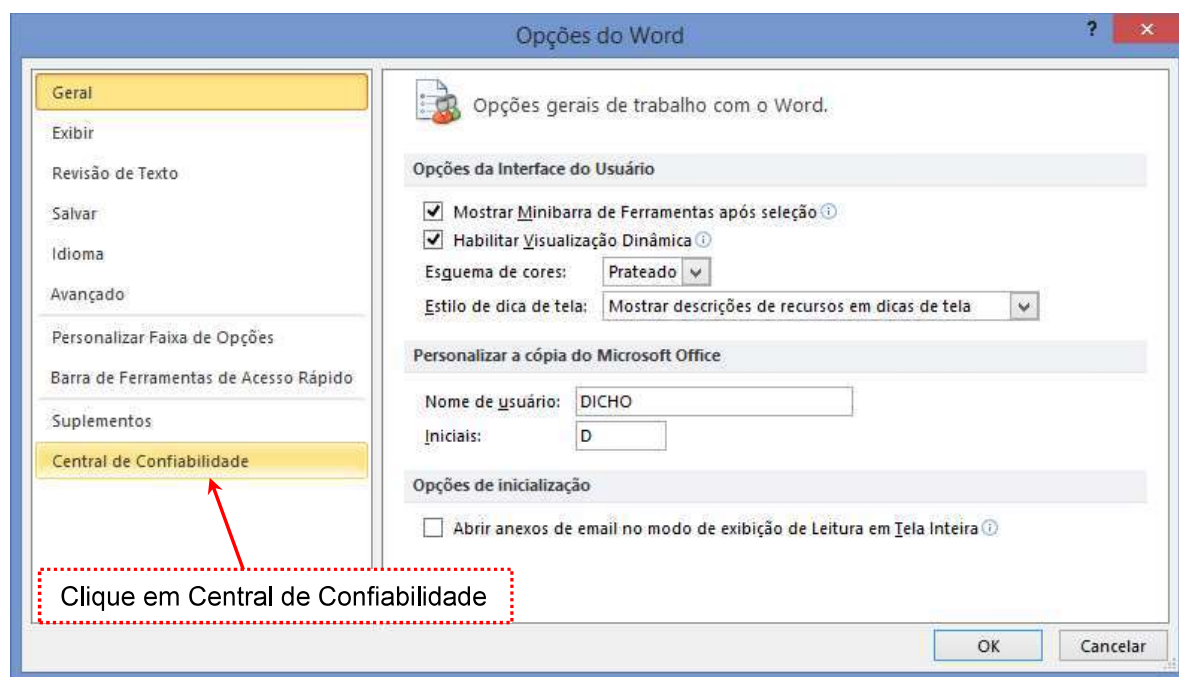
Feche o documento e em seguida abra-o novamente para ver o resultado:



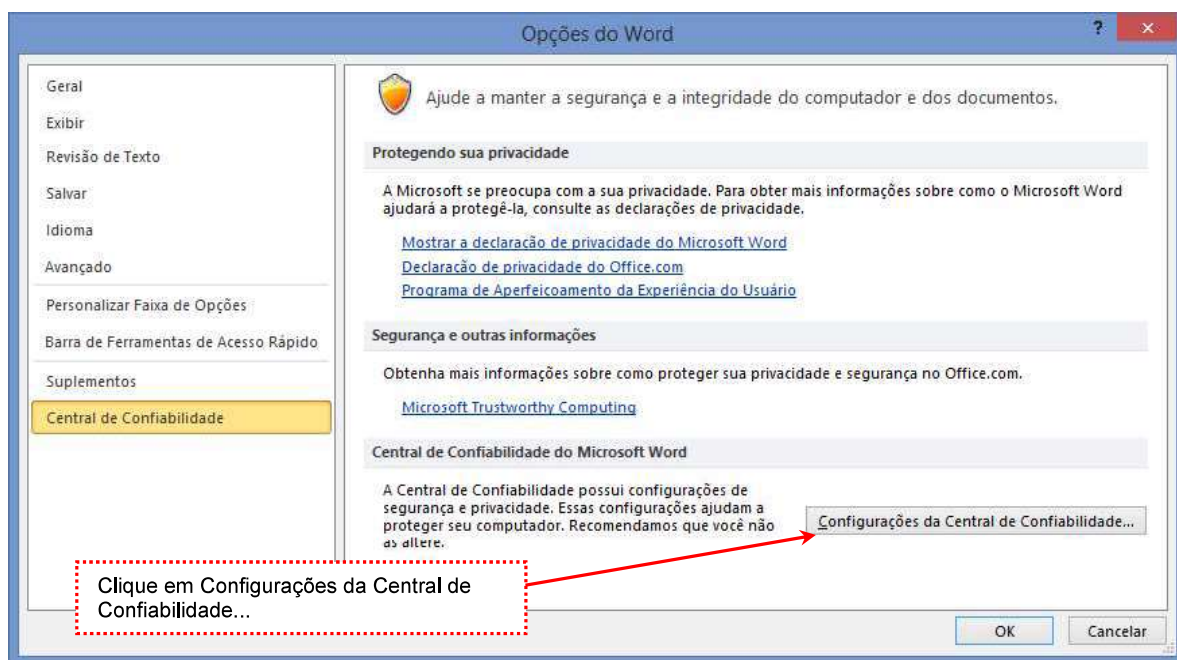
Observação importante: Para que uma macro possa ser executada é necessário alterar a configuração de Macro do Word através da Guia **Arquivo** >> **Opções** >> **Central de Confiabilidade**. Como mostra as imagens abaixo:



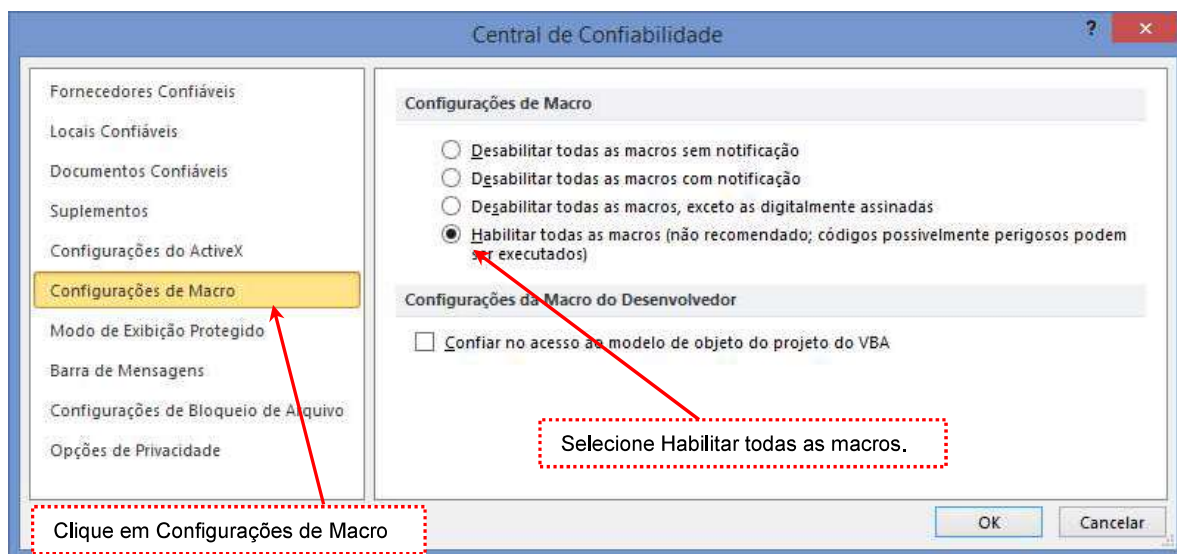
Na janela de Opções do Word clique em **Central de Confiabilidade**:



Clique em **Configurações da Central de Confiabilidade...**

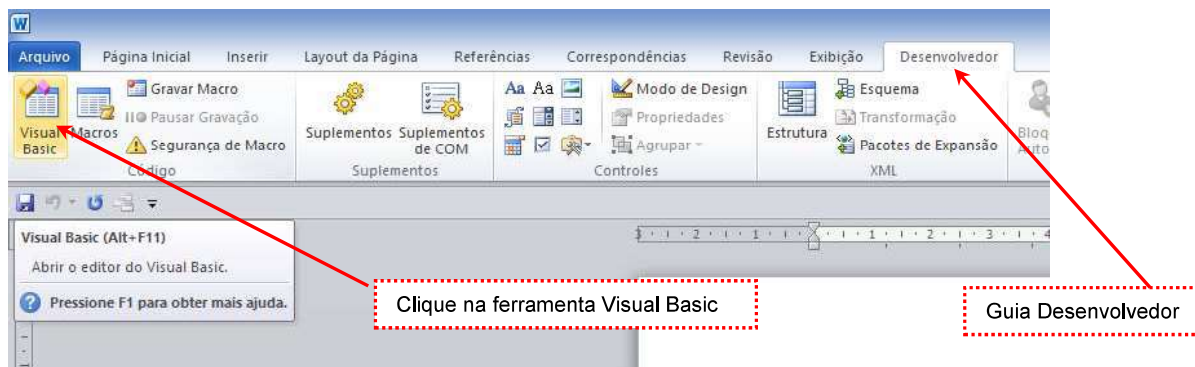


Em Central de Confiabilidade clique em **Configurações de Macro** e em Configurações de Macro selecione a opção **Habilitar todas as macros**.

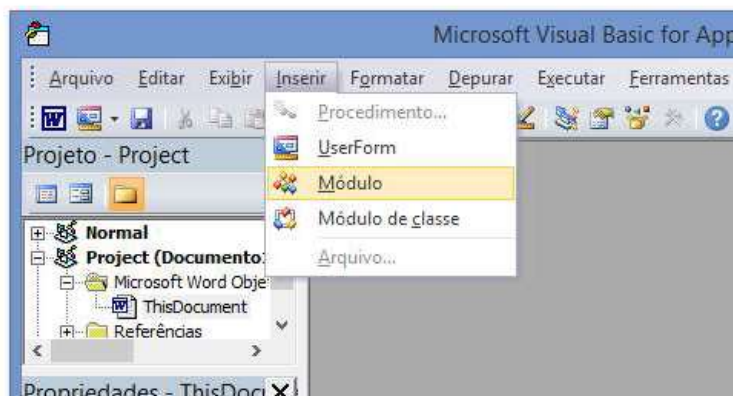


3.2-Exibindo uma saudação ao usuário de acordo com o horário

Abra um documento Word existente ou um novo documento. Abra o Editor do Visual Basic através da **Guia Desenvolvedor** e no **Grupo Código** clique em **Visual Basic** ou simplesmente através do atalho **Alt + F11**.



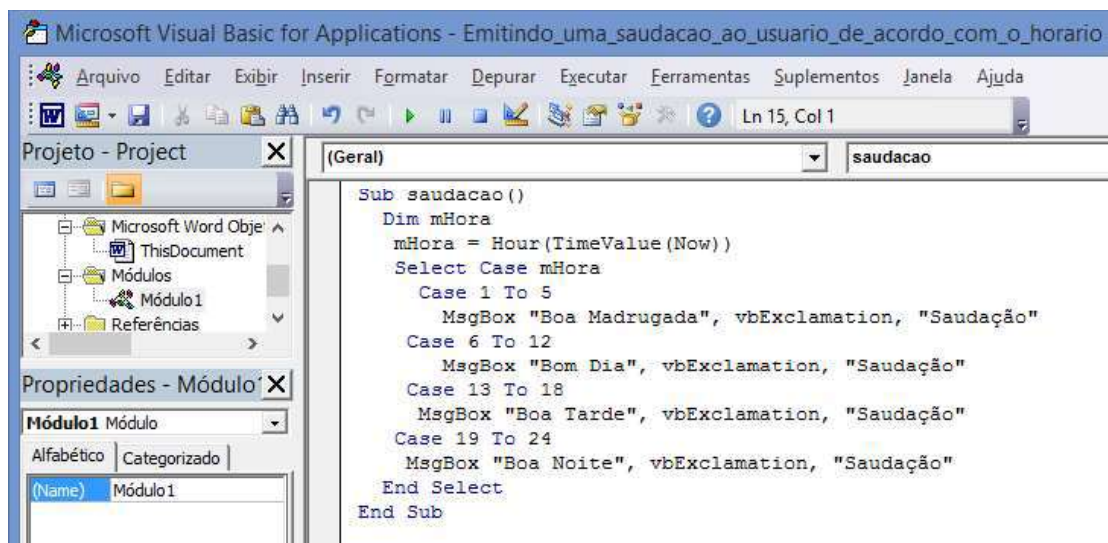
No ambiente de programação do VBA clique no menu **Inserir >> Módulo** para abrir a janela de edição dos códigos (**Módulo1**).



Digite o código abaixo:

```
Sub saudacao()  
    Dim mHora  
    mHora = Hour(TimeValue(Now))  
    Select Case mHora  
        Case 1 To 5  
            MsgBox "Boa Madrugada", vbExclamation, "Saudação"  
        Case 6 To 12  
            MsgBox "Bom Dia", vbExclamation, "Saudação"  
        Case 13 To 18  
            MsgBox "Boa Tarde", vbExclamation, "Saudação"  
        Case 19 To 24  
            MsgBox "Boa Noite", vbExclamation, "Saudação"  
    End Select  
End Sub
```

Observe a imagem:



Explicando o Código:

As linhas de código acima cria uma macro chamada **“saudacao”** cuja rotina emitirá em uma caixa de mensagem, uma saudação ao usuário, de acordo com o horário.

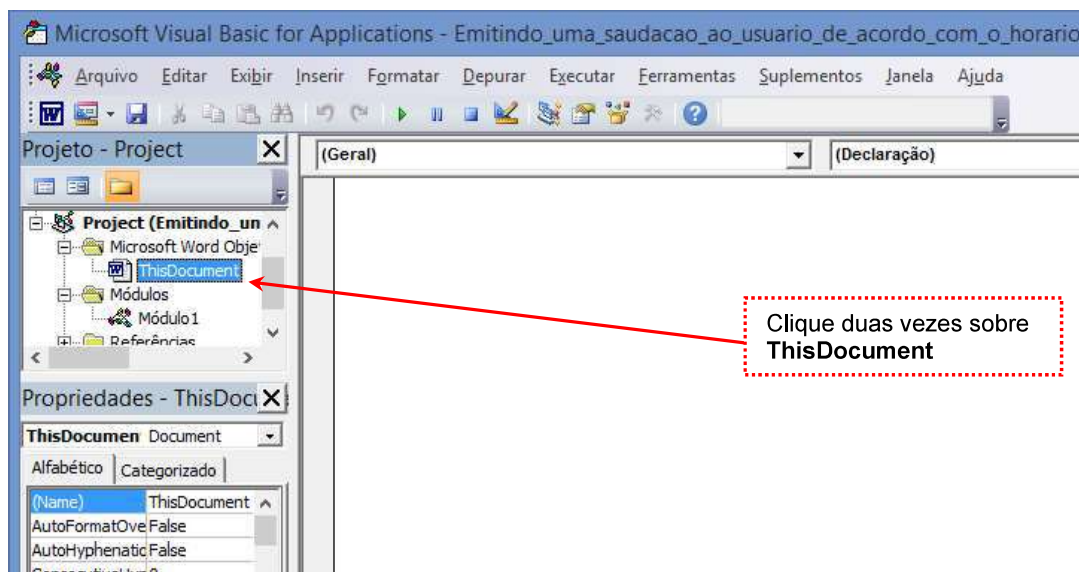
Primeiramente declaramos a variável (aqui neste exemplo demos o nome de mHora);

Utilizamos a função Hour que retorna como parâmetro uma hora, e a função TimeValue juntamente com a propriedade Now que retorna a hora atual do sistema.

A estrutura Select Case testa um valor dentre os vários possíveis e determina qual Grupo de código será chamado:

- Se o documento for aberto no horário das 01:00 h às 05:00 h será exibido a mensagem: “Boa Madrugada”.
- Se o documento for aberto no horário das 06:00 h às 12:00 h será exibido a mensagem: “Bom Dia”.
- Se o documento for aberto no horário das 13:00 h às 18:00 h será exibido a mensagem: “Boa Tarde”.
- Se o documento for aberto no horário das 19:00 h às 24:00 h será exibido a mensagem: “Boa Noite”

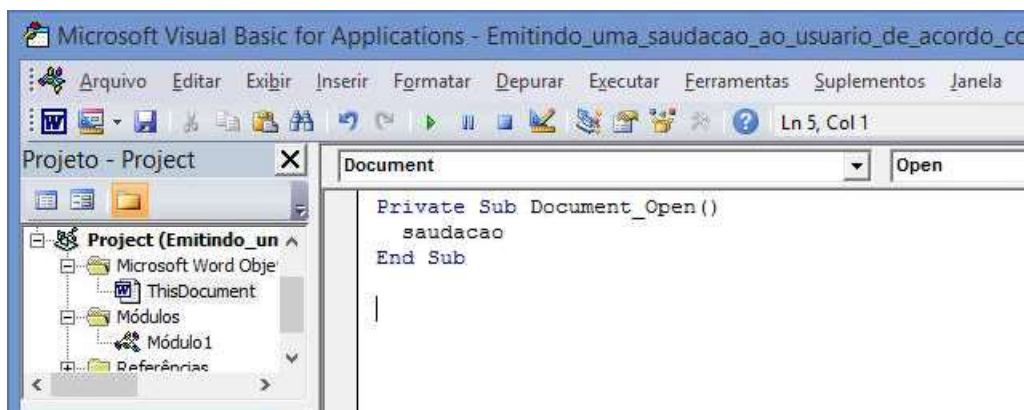
Agora clique duas vezes sobre **“ThisDocument”** que fica na janela Projeto – Project.



Insira o seguinte código:

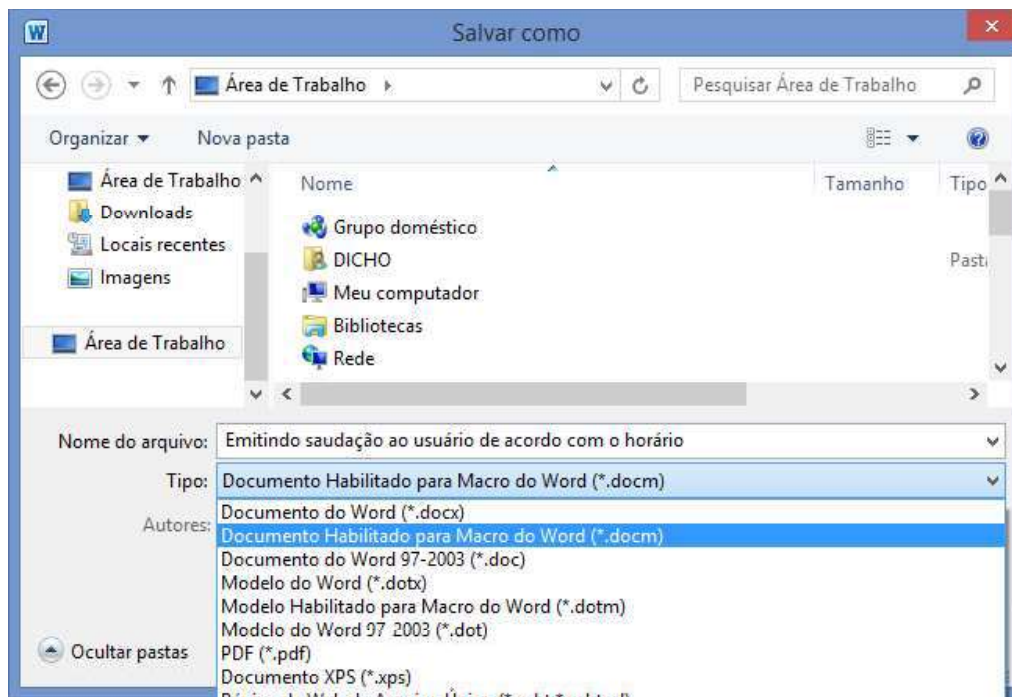
```
Private Sub Document_Open()  
    saudacao  
End Sub
```

Veja a imagem abaixo:



Estas linhas de código fará com que a macro **“saudacao”** seja executada todas as vezes em que o documento for aberto.

Feche o Editor do Visual Basic e salve o documento como **Documento Habilitado para Macro do Word**.

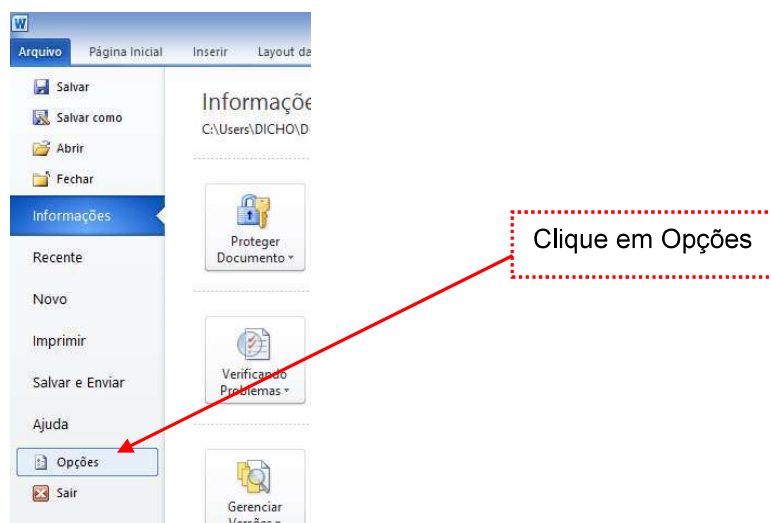


Feche o documento e abra-o novamente. Será exibido uma mensagem ao usuário de acordo com o horário em que o documento está sendo aberto

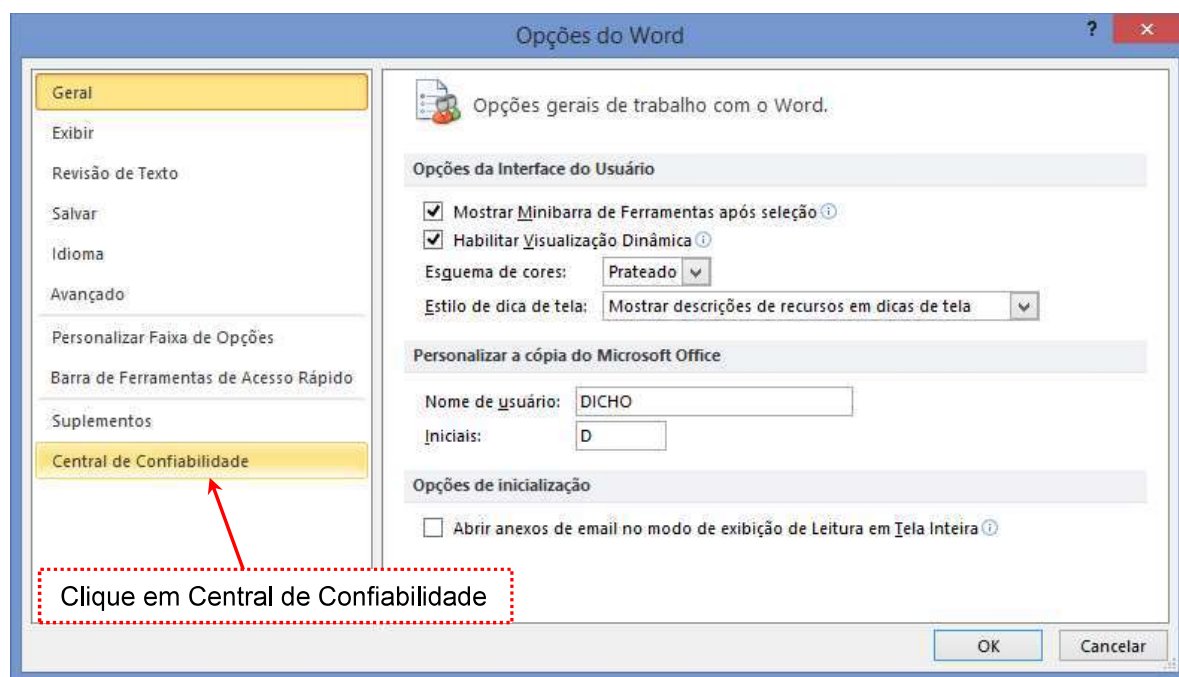
Veja o resultado:



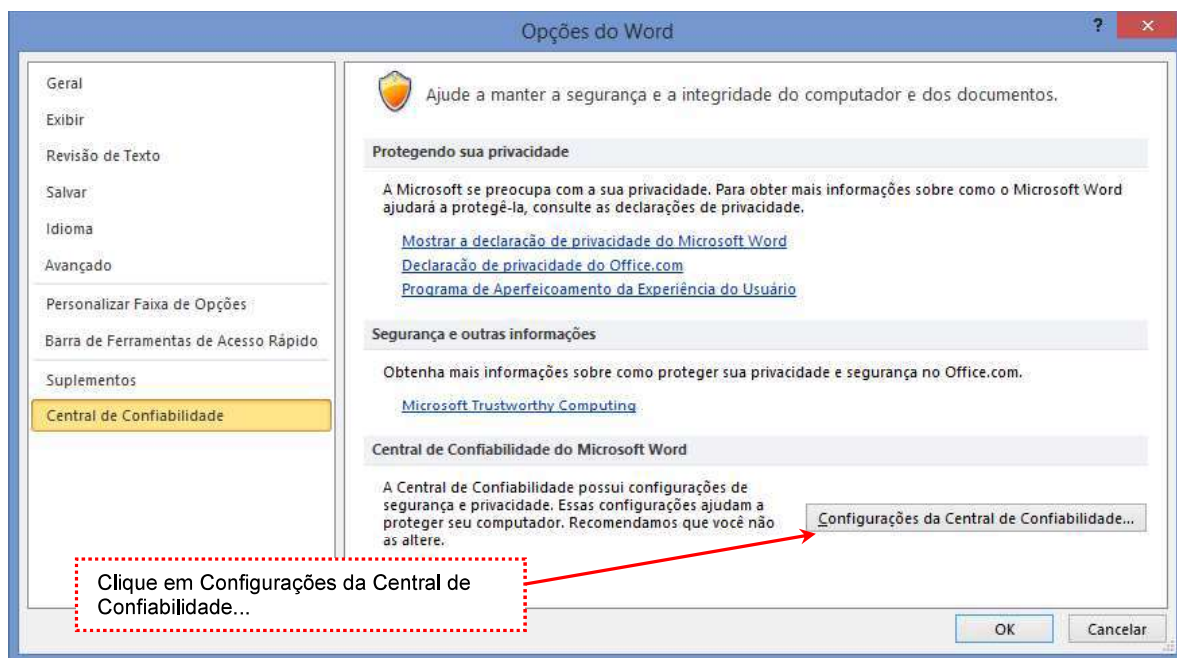
Observação importante: Para que uma macro possa ser executada é necessário alterar a configuração de Macro do Word através da Guia **Arquivo** >> **Opções** >> **Central de Confiabilidade**. Como mostra as imagens abaixo:



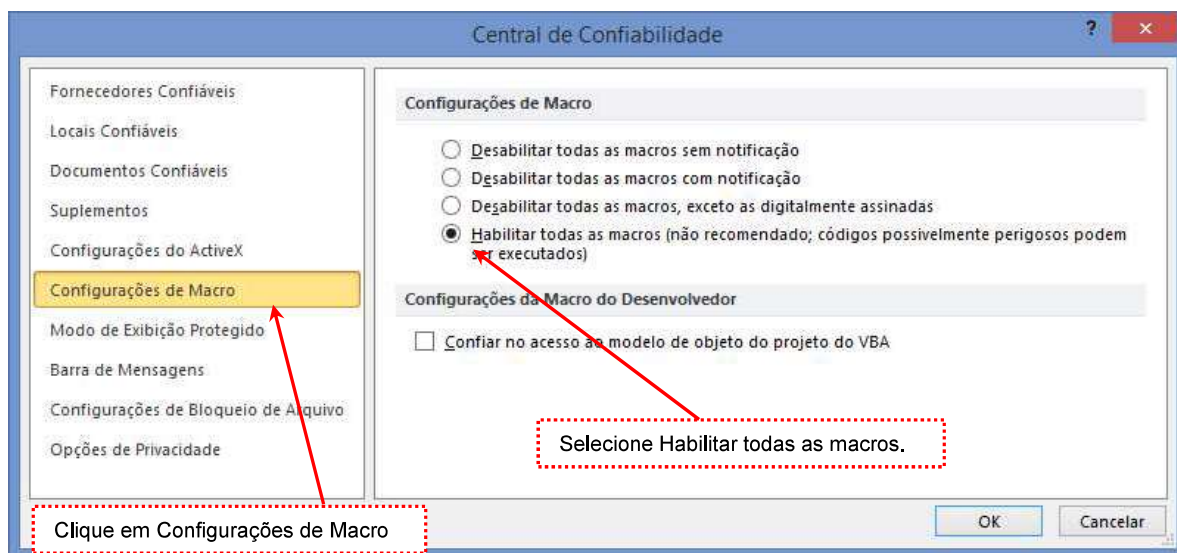
Na janela de Opções do Word clique em **Central de Confiabilidade**:



Clique em **Configurações da Central de Confiabilidade...**



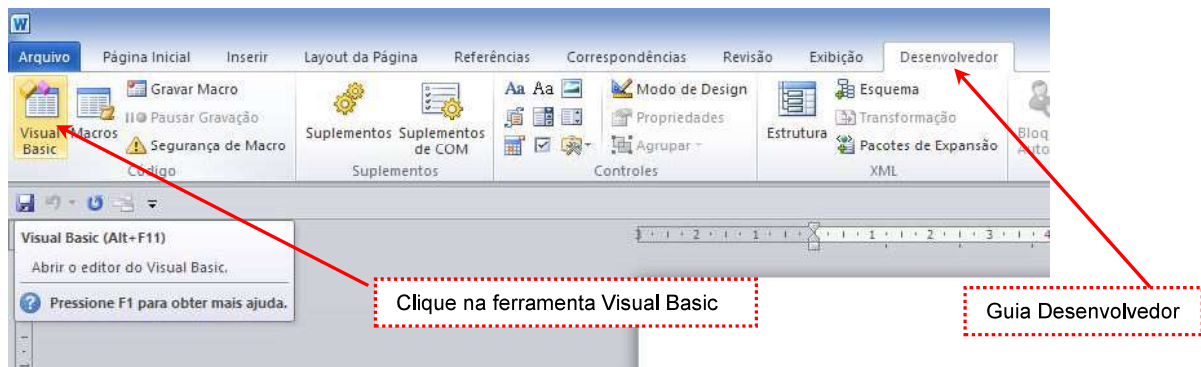
Em Central de Confiabilidade clique em **Configurações de Macro** e em Configurações de Macro selecione a opção **Habilitar todas as macros**.



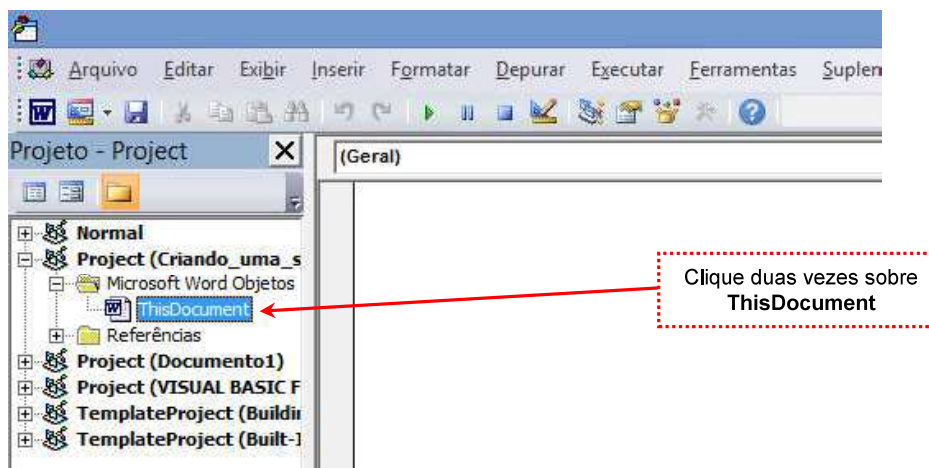
3.3-Saudação – interagindo com usuário

Outra forma simples de saudação ao usuário:

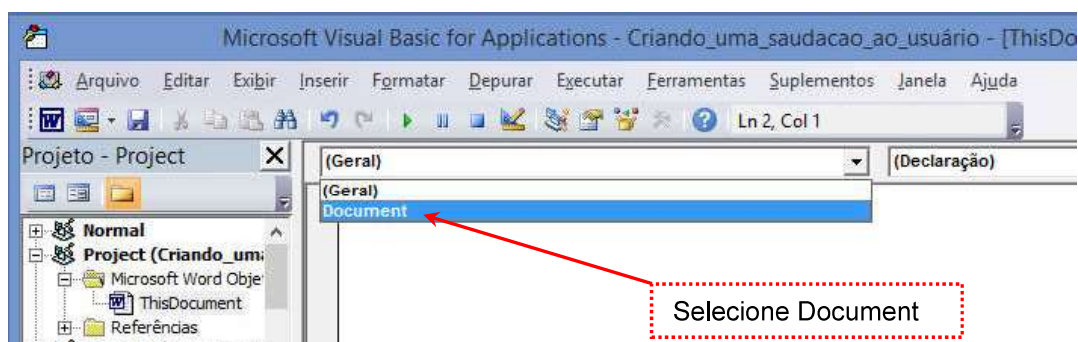
Abra um documento Word existente ou um novo documento. Abra o Editor do Visual Basic (VBE) através da **Guia Desenvolvedor** e no **Grupo Código** clique em **Visual Basic** ou simplesmente através do atalho **Alt + F11**.



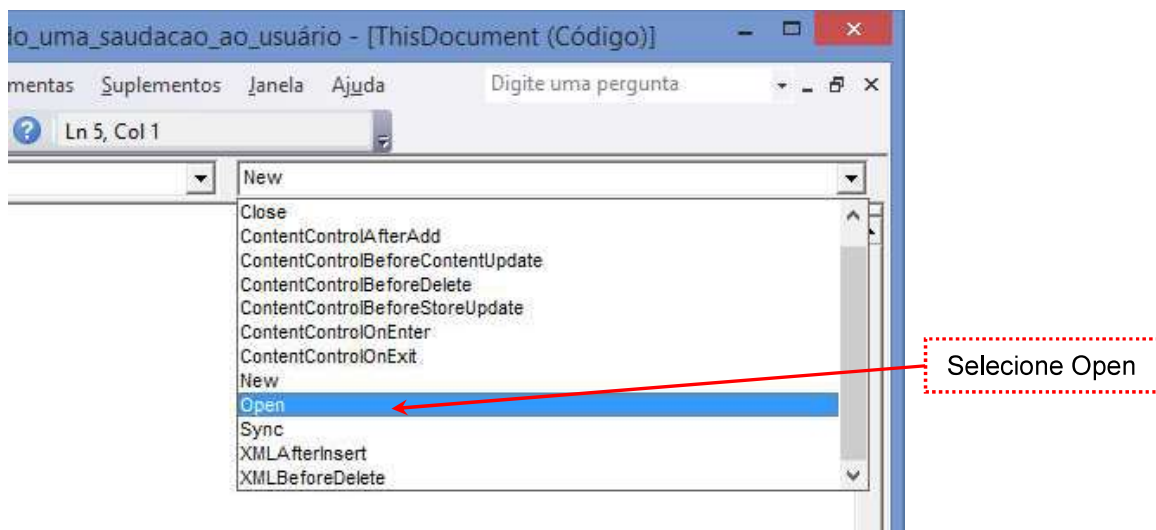
No ambiente do VBA clique duas vezes sobre **"ThisDocument"** que fica na janela **Projeto –Project**.



Na **Janela de Código** selecione o objeto **Document**:



Em procedimento selecione **Open**:



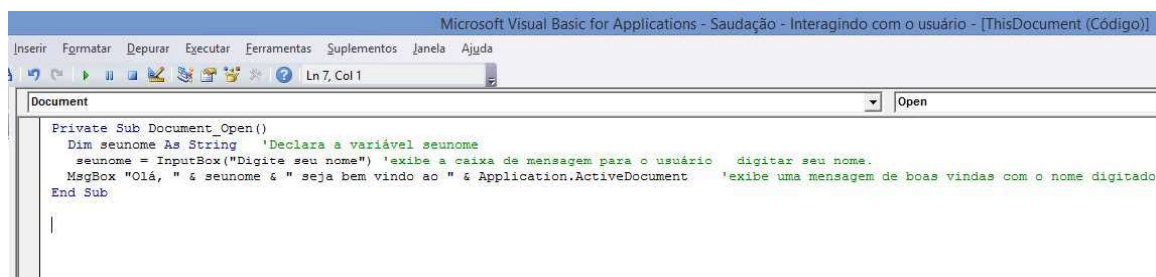
Digite o código abaixo:

```
Private Sub Document_Open()  
Dim seunome As String 'Declara a variável seunome  
seunome = InputBox("Digite seu nome") 'exibe a caixa de mensagem para o usuário digitar seu nome.  
MsgBox "Olá, " & seunome & " seja bem vindo ao " & Application.ActiveDocument 'exibe uma  
mensagem de boas vindas com o nome digitado InputBox exibindo também o nome do documento ativo.  
End Sub
```

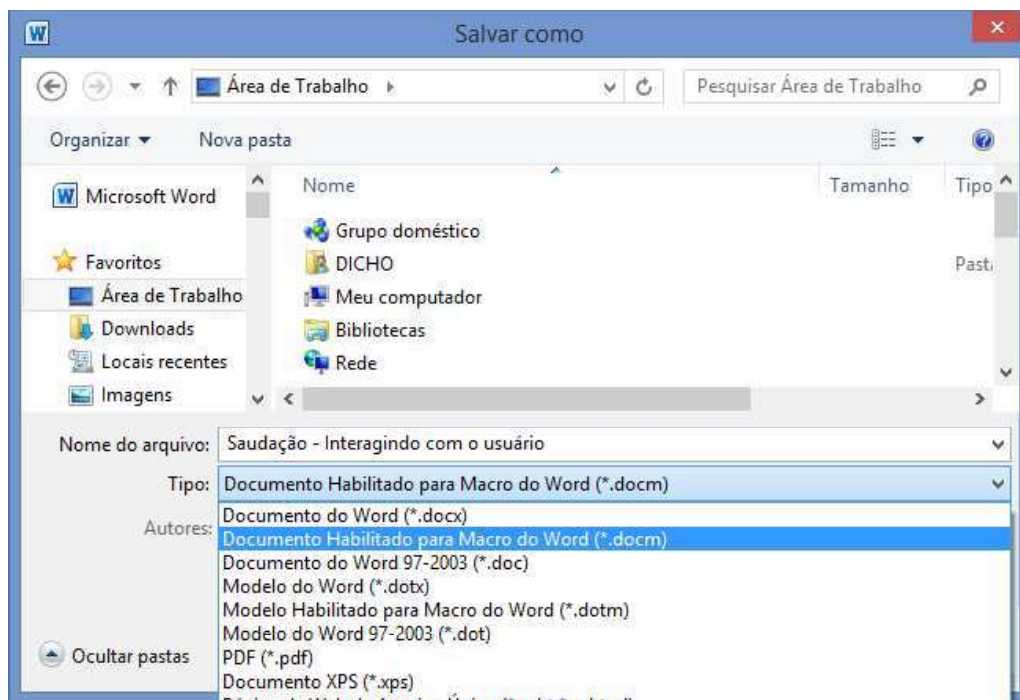
Explicando o Código:

O valor da variável **seunome** é igual ao nome que foi digitado na InputBox. Ao clicar no Botão OK será exibida uma caixa de mensagem de boas vindas com o nome do usuário e o nome do Documento ativo.

OBS: As frases em verde são apenas comentários que foram inseridos para melhor entender o código. Não deve ser tirado o ' (apóstrofo) porque se for tirado deixará de ser um comentário e o editor entenderá como linha de código o que gerará erros na sua execução.

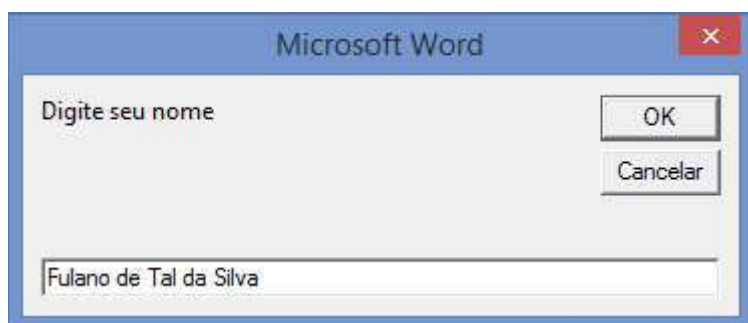


Salve o documento como **Documento Habilitado para Macro do Word**.

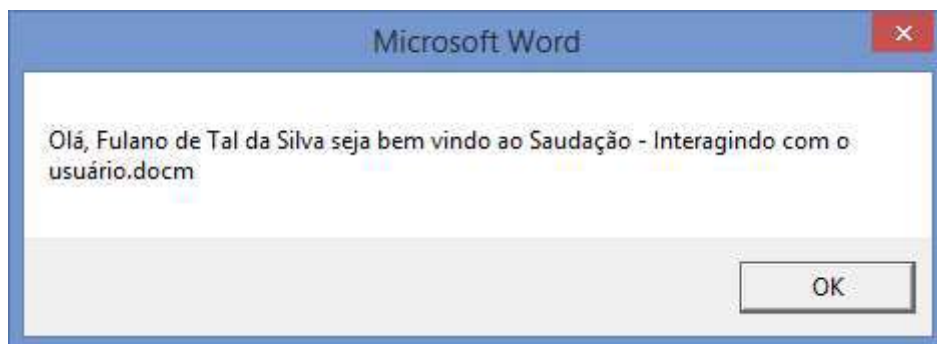


Feche o documento e em seguida abra-o novamente para ver o resultado.

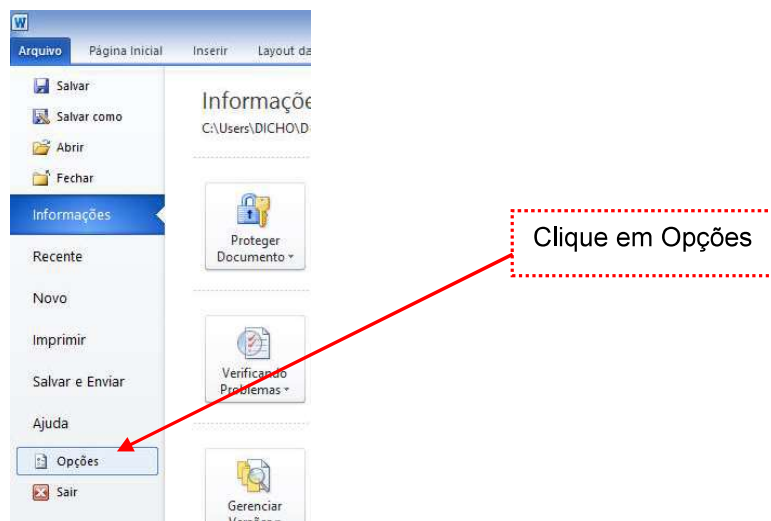
Será exibida uma caixa de mensagem para que o usuário insira o seu nome.



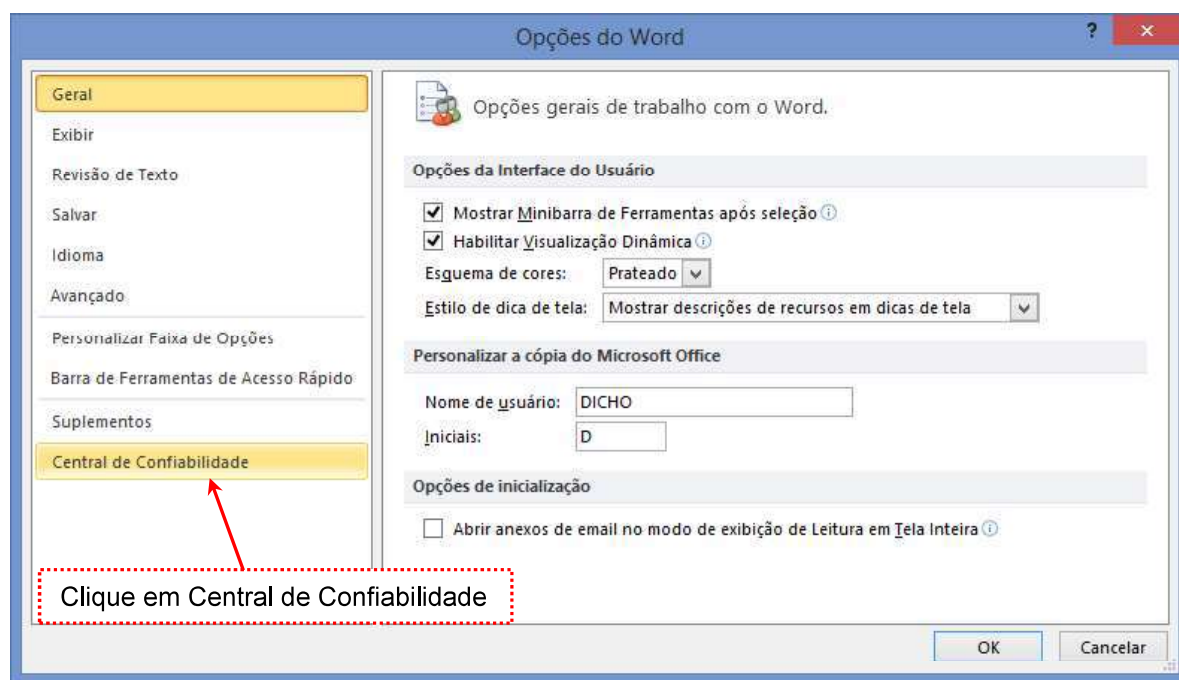
Após inserir o nome e clicar no botão OK será exibida outra caixa de mensagem com o nome do usuário e o nome do documento.



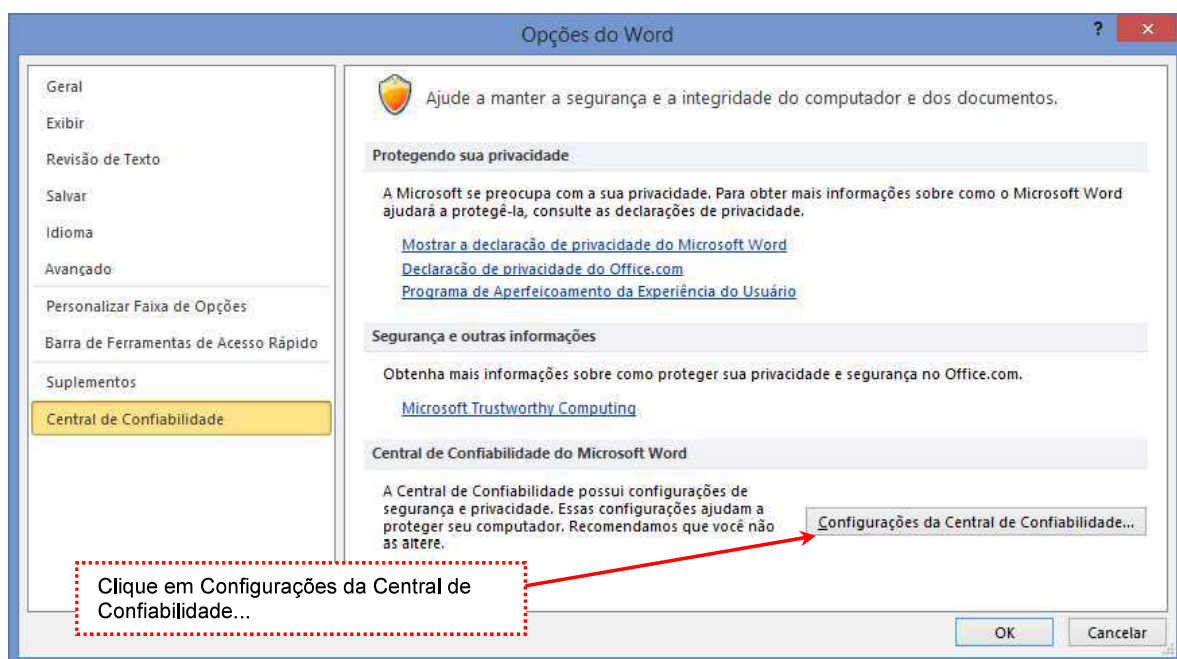
Observação importante: Para que uma macro possa ser executada é necessário alterar a configuração de Macro do Word através da Guia **Arquivo** >> **Opções** >> **Central de Confiabilidade**. Como mostra as imagens abaixo:



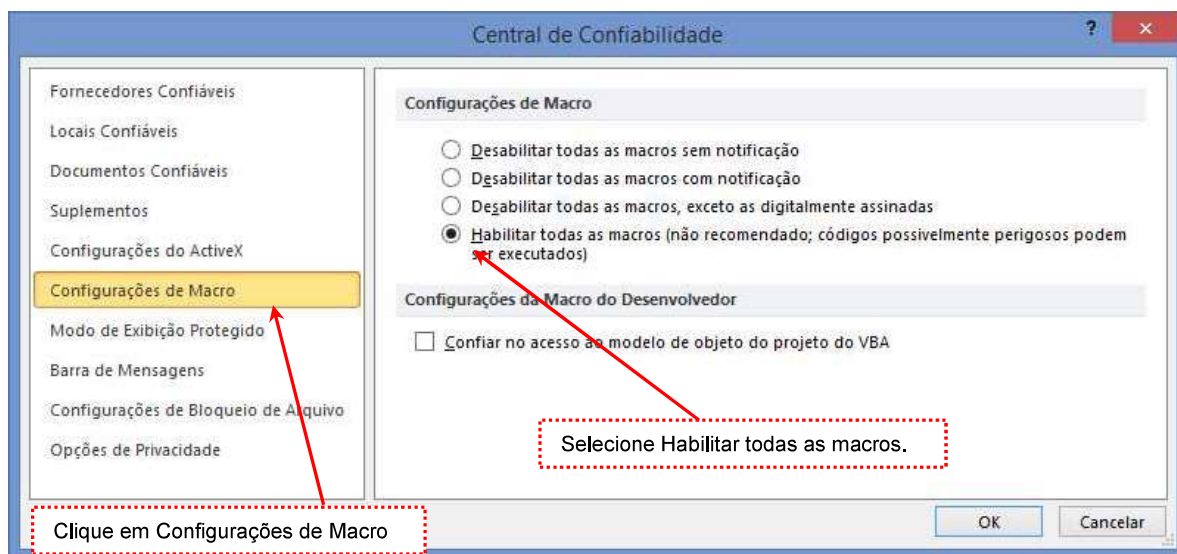
Na janela de Opções do Word clique em **Central de Confiabilidade**:



Clique em **Configurações da Central de Confiabilidade...**



Em Central de Confiabilidade clique em **Configurações de Macro** e em Configurações de Macro selecione a opção **Habilitar todas as macros**.

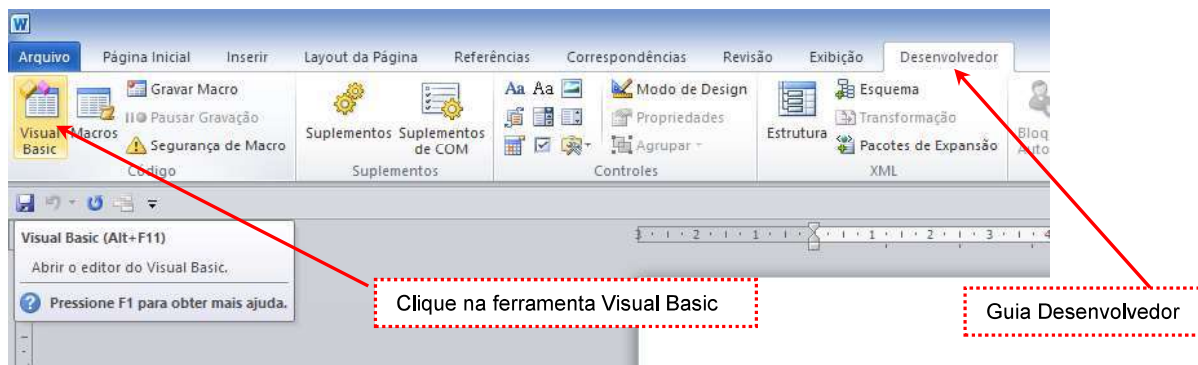


3.4-Outro exemplo utilizando a função INPUTBOX

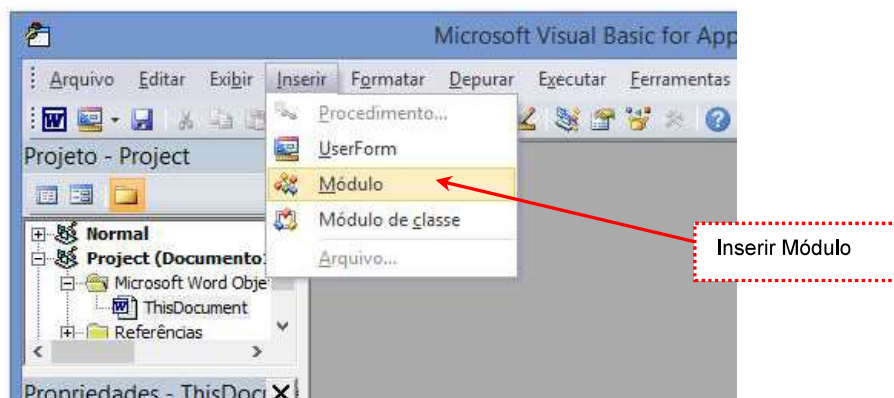
Outro exemplo utilizando a função INPUTBOX para calcular valores e apresentar o resultado em uma caixa de mensagem.

Neste exemplo usaremos duas InputBox, para realizar a soma de valores.

Abra um documento Word existente ou um novo documento. Abra o Editor do Visual Basic através da **Guia Desenvolvedor** e no **Grupo Código** clique em **Visual Basic** ou simplesmente através do atalho **Alt + F11**.



No ambiente de programação do VBA clique no menu **Inserir >> Módulo** para abrir a janela de edição dos códigos (**Módulo1**) e crie a seguinte macro com o nome calculavalores:

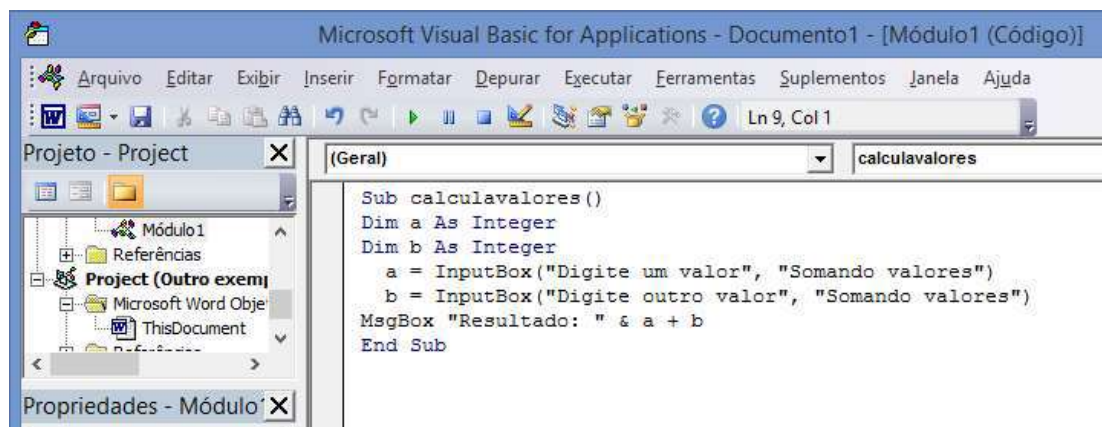


Na janela de Código digite:

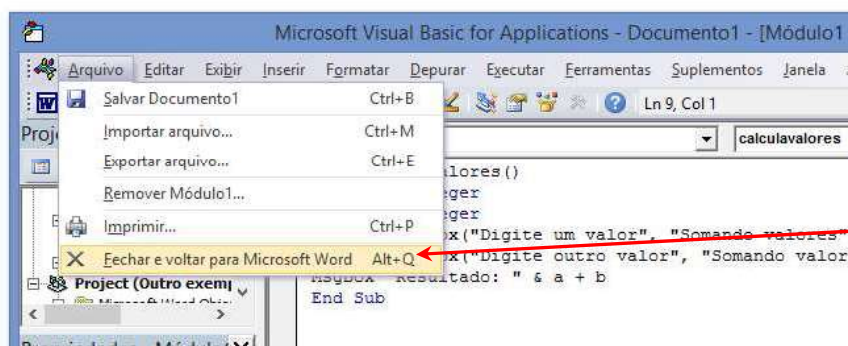
```
Sub calculavalores()  
Dim a As Integer  
Dim b As Integer  
a = InputBox("Digite um valor", "Somando valores")  
b = InputBox("Digite outro valor", "Somando valores")  
MsgBox "Resultado= " & a + b  
End Sub
```


Explicando o Código:

Primeiramente declaramos duas variáveis “a” e “b” como tipo Integer. O valor da variável “a” é igual ao valor digitado na primeira InputBox e o valor da variável “b” é igual ao valor digitado na segunda InputBox. Será exibida uma caixa de mensagem com a soma dos valores de “a” e “b”.



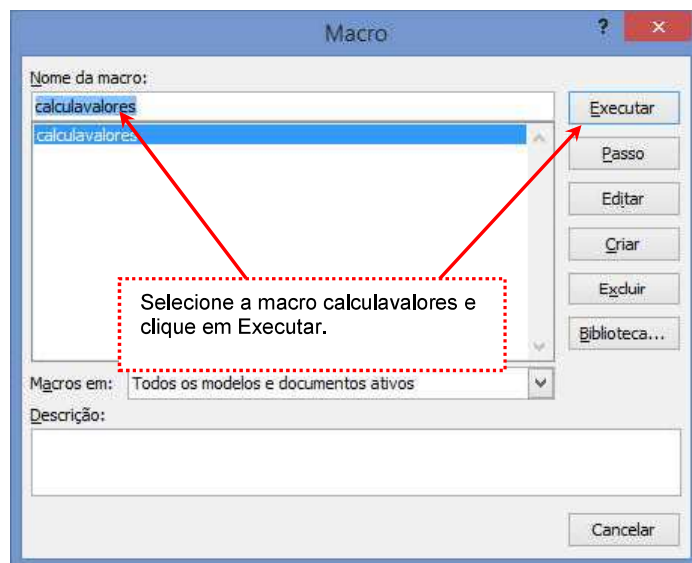
Agora feche o Editor do Visual Basic para voltar ao Word. Clique no Menu **Arquivo** e em seguida em **Fechar e voltar para Microsoft Word**.



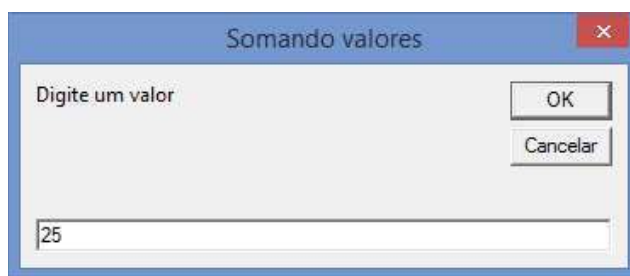
Acione a Guia **Desenvolvedor** e no Grupo **Código** clique em **Macros** ou se preferir use a combinação de teclas (teclas de atalho) **Alt + F8**.



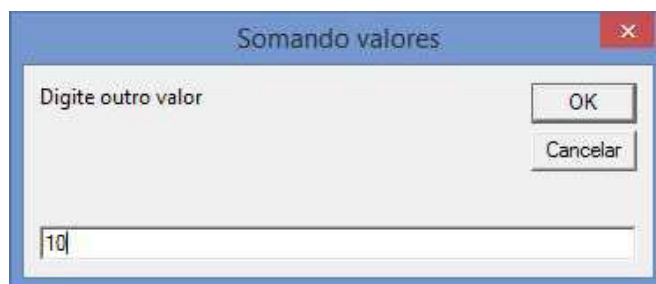
Selecione a macro **calculavalores** e clique em **Executar**.



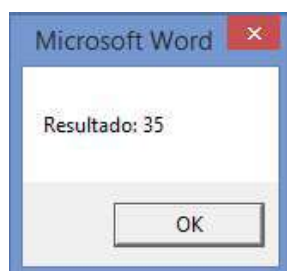
Surgirá uma caixa de mensagem para que seja inserido o primeiro valor.



Insira um valor e clique no botão OK. Surgirá uma nova caixa de mensagem para que seja inserido o segundo valor.



Insira o segundo valor e clique no botão OK. O resultado da soma será exibida em uma nova caixa de mensagem.

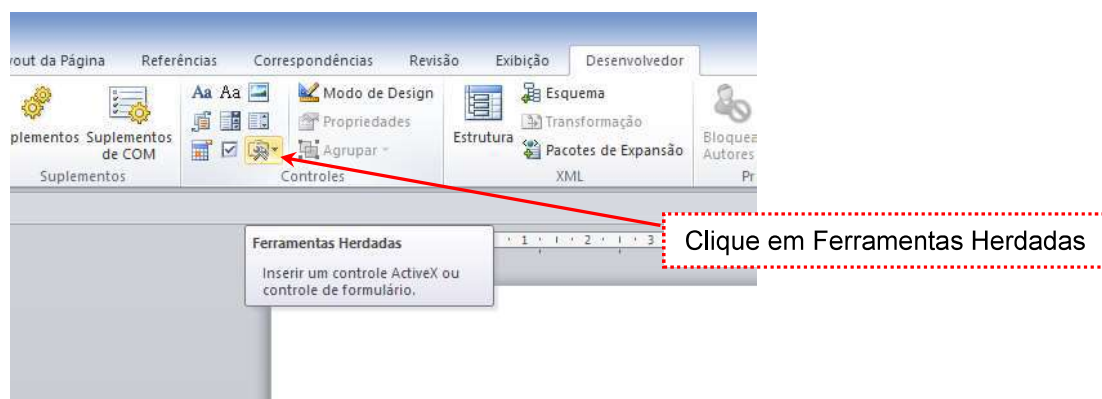


Podemos ainda associar essa macro a um botão usando um **Controle ActiveX** ou adicioná-la a **Faixa de Opções** ou ainda inseri-la na **Barra de Ferramentas de Acesso Rápido**.

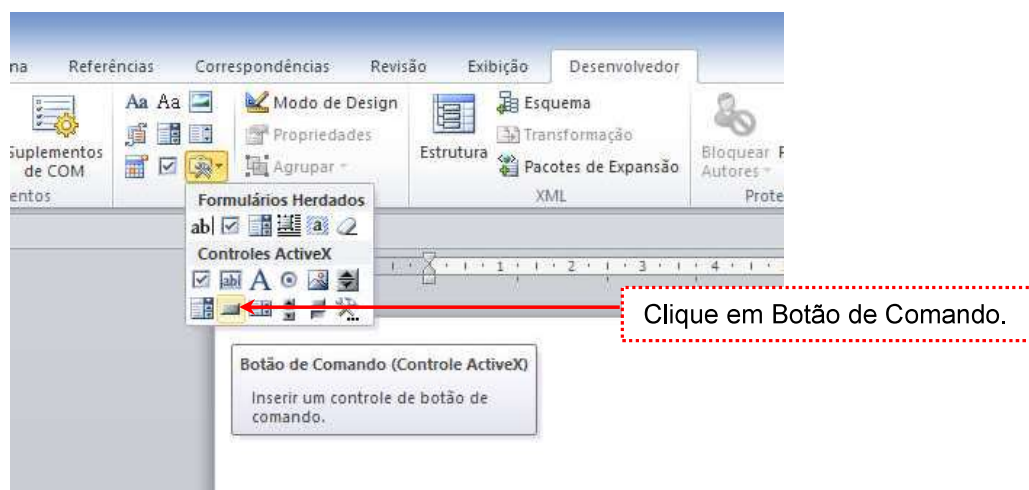
Mostraremos aqui como fazer as três opções.

1ª Opção: usando Controle ActiveX

Acione a Guia **Desenvolvedor** e no Grupo **Controles** clique em **Ferramentas Herdadas**.



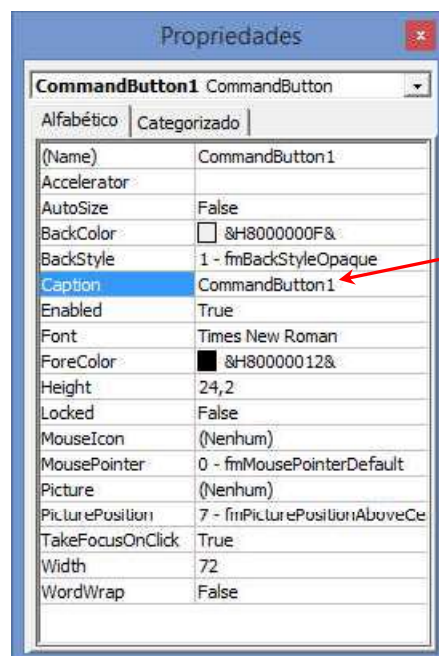
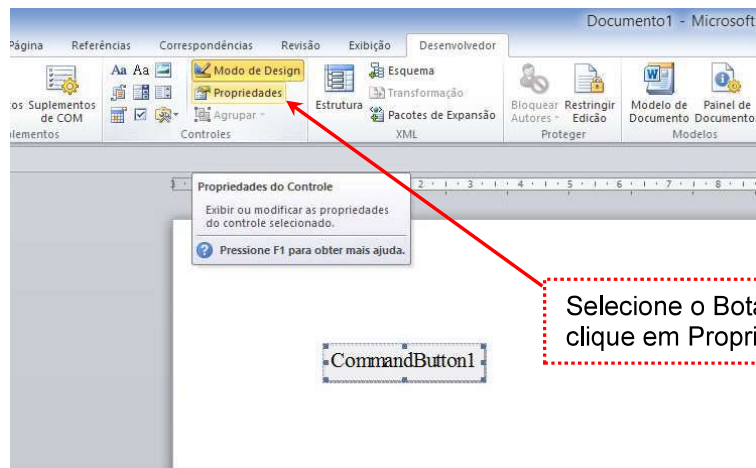
Em Controles ActiveX clique em Botão de Comando.

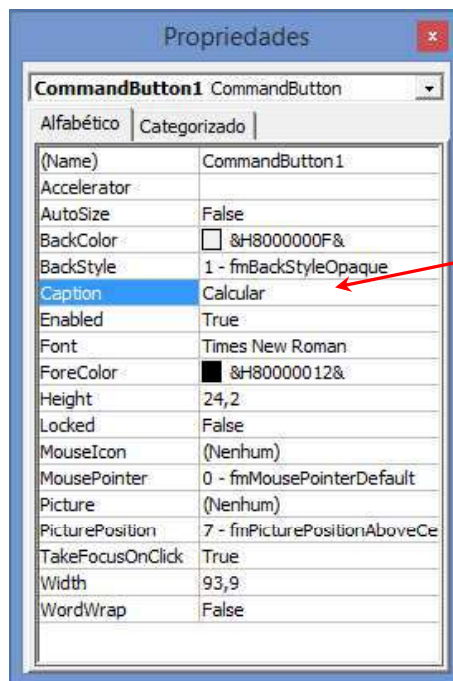


Por padrão o Botão de Comando possui sua **Propriedade Caption** como **CommandButton1** e caso seja inserido um novo botão ele será CommandButton2 e assim sucessivamente. Vamos alterar essa propriedade para “**Calcular**”.

CommandButton1

Selecione o Botão de Comando que acabamos de inserir no documento e no Grupo Controles clique em **Propriedades**.

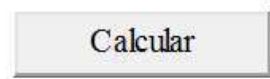




Propriedade Caption alterada para Calcular.

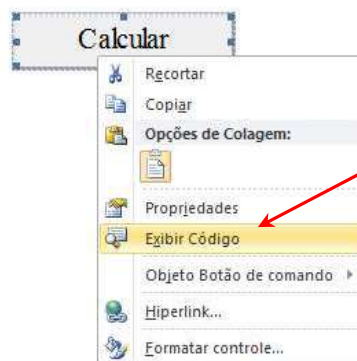
Você ainda pode, se quiser, alterar outras propriedades do botão como a sua cor através do propriedade BackColor; a fonte, estilo da fonte e tamanho através da propriedade Font, a cor da fonte através da propriedade ForeColor, etc.

Veja como ficou o nosso botão:



Agora é necessário colocar o código por trás do botão que acionará a macro **calculavalores**.

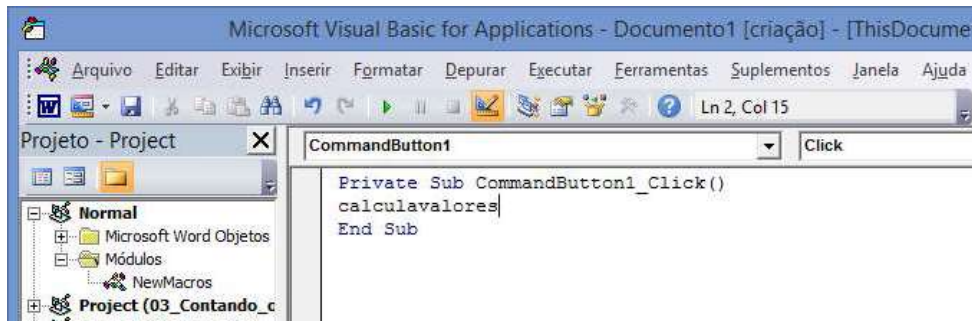
Clique duas vezes sobre o botão para abrir o Editor do Visual Basic ou selecione o botão e clique com o botão direito do mouse e selecione **Exibir Código**.



Clique com o botão direito do mouse e clique em Exibir Código.

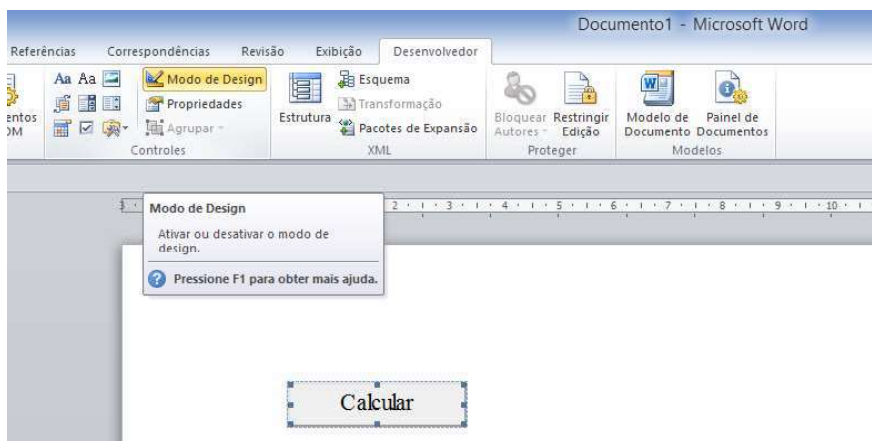
No evento click do botão digite o nome de sua macro, neste caso é “calculavalores”. Ficará assim:

```
Private Sub CommandButton1_Click()  
calculavalores  
End Sub
```



Feche o Editor do Visual Basic para voltar ao Word.

No Word selecione o botão de comando e clique em Modo de Design para desativar o Modo Design.



Tendo feito isso é só clicar no botão Calcular para que sejam exibidas as caixas de mensagens solicitando que sejam inseridos os valores para realizar o cálculo.