

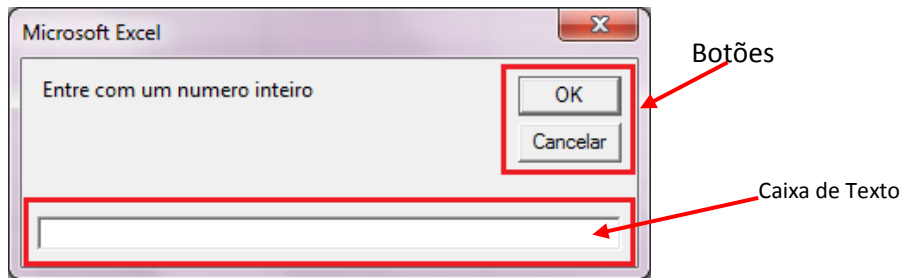
AULA 02 – InputBox e MsgBox

No desenvolvimento de programas VBA é possível utilizar outro mecanismo de entrada e saída de dados

Comando InputBox()

O Comando InputBox é uma função do Excel que permite ao usuário introduzir dados no programa, ou seja, representa um mecanismo de ENTRADA DE DADOS. Ao ser executado, o Comando InputBox, exibe uma janela com uma caixa de texto para a inserção de dados.

Ao ser executado, este comando exibe uma Caixa de Diálogo que possui uma região onde o usuário deverá digitar um dado solicitado e clicar em um botão. Estão disponíveis dois botões: botão OK e botão Cancelar. Se o usuário escolher o botão Ok e tiver algum dado digitado, este será atribuído a uma variável. Se ele clicar um Cancelar será armazenado o espaço em branco (" ") na variável. Observe a figura abaixo:



Assim, o usuário recebe uma mensagem com uma questão e a janela aguarda que ele digite alguma dado, que neste caso, será uma String.

No caso do dado ser um numero inteiro ou real, é boa prática fazer a conversão deste dado antes de armazená-lo numa variável.

Funções que realizam a conversão de tipos de dados:

- `CInt()` – dados inteiros.
- `CSng()` – dados reais.
- `CDbl()` – dados com dupla precisão.
- `CStr()` – dados texto (String).

Sintaxe:

```
Variável = InputBox (mensagem, título, valor_default, xpos, ypos)
```

Os argumentos são:

- **Mensagem Texto** da mensagem a ser inserido na Caixa de Entrada (máximo de 1024 caracteres)
- **Título** Conteúdo da barra de título da janela (opcional)
- **valor_default** Valor inicial a ser considerado default na Caixa de entrada (opcional)
- **xpos e ypos** Coordenadas da "InputBox" relativamente aos bordos esquerdo e superior da janela (opcionais)

Exemplo de codificação do comando Inputbox:

1. Considere a leitura de um dado inteiro e que este dado seja armazenado na célula A1.

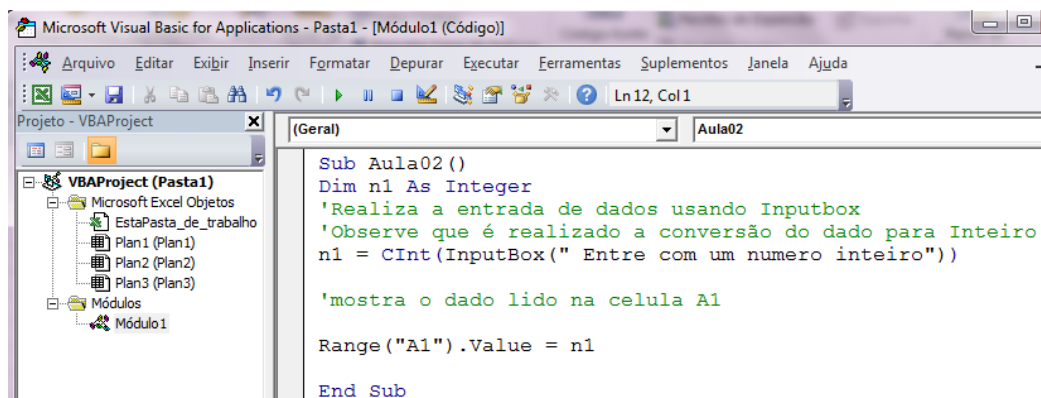
Dados de Entrada (numero inteiro):

n1 que é do tipo numérico inteiro

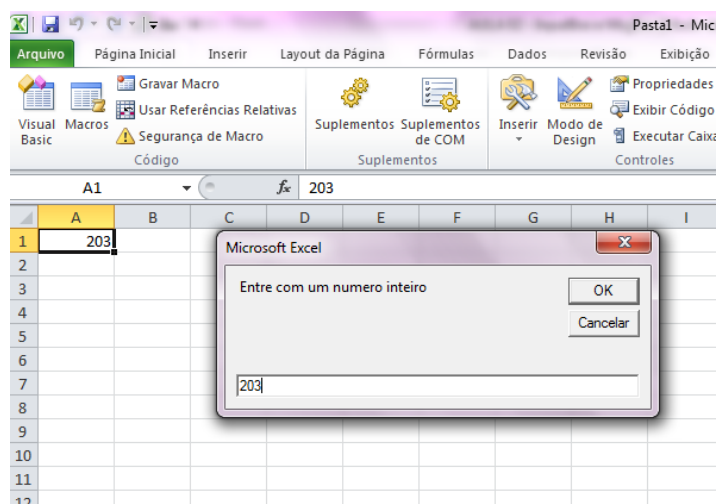
Dados de Saída(Resultados): (o que é pedido ou solicitado):

A saída de dados é mostrar o dado na planilha na célula A1.

Assim o código em VBA para este problema é:

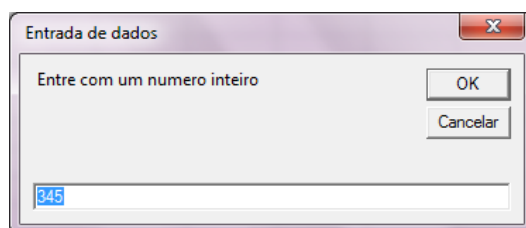


Ao executar a macro , vai gerar uma janela solicitando a entrada de dados.



Ao clicar o botão OK, a macro vai inserir o número 203 na célula A1.

A seguir o comando inserindo o titulo da Mensagem:



Observe a o titulo da janela “ Entrada de dados” e observe o valor *default* que é mostrado ao abrir a janela, no valor de 345.

Comando MsgBox

Este comando permite apresentar uma Caixa de Mensagem que representa a Saída de Dados ou resultado de um programa em VBA. Neste caso, o comando *MsgBox* gera uma pequena janela contendo uma mensagem com pelo menos um botão de comando e em algumas situações, com um pequeno ícone ilustrativo de acordo com o tipo da mensagem mostrada.

Quando o usuário seleciona e clica um botão, gera como resultado um número inteiro que representa a escolha realizada pelo usuário.

Sintaxe:

```
variável = MsgBox(mensagem [, botão] [, titulo] [, ajuda ,contexto])
```

Os itens entre colchetes “[]” são opcionais.

Abaixo é mostrada uma tabela com as variáveis que devem ser utilizadas para os ícones e botões

Constante de VBA	Valor	Descrição
VbOKOnly	0	Exibe somente o botão de OK.
VbOKCancel	1	Exibe os botões OK e Cancel.
VbAbortRetryIgnore	2	Exibe os botões Abort, Retry, e Ignore.
VbYesNoCancel	3	Exibe os botões Yes, No, e Cancel .
VbYesNo	4	Exibe os botões Yes e No.
VbRetryCancel	5	Exibe os botões Retry e Cancel.
VbCritical	16	Exibe o ícone de Critical Message.
VbQuestion	32	Exibe o ícone de Warning Query.
VbExclamation	48	Exibe o ícone de Warning Message.
VbInformation	64	Exibe o ícone de Information Message.
VbDefaultButton1	0	O primeiro botão é o seleccionado por defeito.
VbDefaultButton2	256	O segundo botão é o seleccionado por defeito.
VbDefaultButton3	512	O terceiro botão é o seleccionado por defeito.
VbDefaultButton4	768	O quarto botão é o seleccionado por defeito.
VbApplicationModal	0	Application modal – o utilizador só depois de responder à MsgBox é que poderá dar continuidade ao trabalho na aplicação corrente.
VbSystemModal	4096	System modal - – o utilizador só depois de responder à MsgBox é que poderá dar continuidade ao trabalho em qualquer aplicação em curso no sistema.

Como resultado a execução do comando MsgBox, é possível saber qual botão o usuário selecionou com a mensagem. Observe a figura abaixo.

Observe que para cada tipo de botão ou mesmo ícone existe um número relacionado. Assim, se você quiser os botões OK e Cancel (representados pelo numero 1) e um ícone do tipo Question (representado pelo número 32), deverá inserir: 1 + 32

A próxima tabela mostra o número resultante depois que a caixa de mensagem aparece e o usuário clica em um dado botão.

Se for um botão Ok ele vai retornar o número 1; se for um botão Yes (sim) ele vai retornar o numero 6.

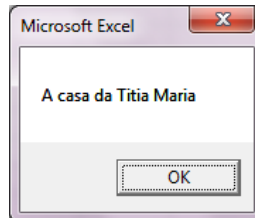
CONSTANTE	VALOR	DESCRIÇÃO
vbOK	1	OK
vbCancel	2	Cancelar
vbAbort	3	Abortar
vbRetry	4	Repetir
vbIgnore	5	Ignorar
vbYes	6	Sim
vbNo	7	Não

Exemplo de utilização do comando MsgBox

1. Considere a codificação abaixo:

```
Sub Aula03()  
MsgBox " A casa da Titia Maria "  
End Sub
```

Ao ser executado a janela mostrada será:

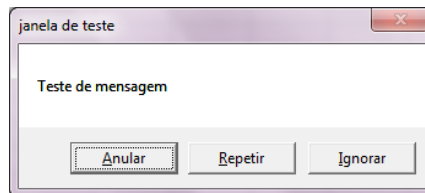


2. Considere o Seguinte comando:

```
Sub Aula04()  
MsgBox " Teste de mensagem ", vbAbortRetryIgnore, "janela de teste"  
End Sub
```

Observe que o será mostrado os botões Anular, Repetir e Ignorar.

Ao ser executado a janela mostrada será:

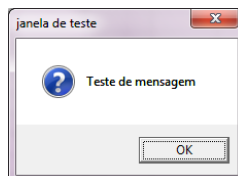


3. Considere o Seguinte comando:

```
Sub Aula05()  
MsgBox " Teste de mensagem ", 32, "janela de teste"  
End Sub
```

Observe que o será mostrado o ícone *Question*.

Ao ser executado a janela mostrada será:



4. Considere o Seguinte comando:

```

Sub Aula06()
Dim msg As String, titulo As String
Dim botao As Integer, resultado As Integer

msg = "Voce realmente deseja mostrar " & vbCrLf & "o nome da planilha?"
botao = vbYesNo + vbQuestion + vbDefaultButton2
titulo = "Mostrando o nome da planilha"

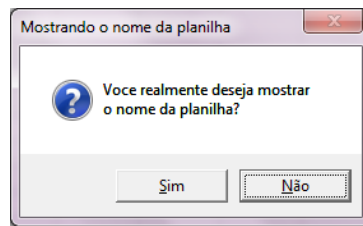
resultado = MsgBox(msg, botao, titulo)
End Sub

```

Observe que o será mostrado os botões Sim ou Não, o ícone *Question*, e será selecionado o segundo botão.

Observe o operador **&** (concatenação) que concatena uma *String* e um numero.

Ao ser executado a janela mostrada será:



Observação:

- O comando *MsgBox* pode ser mostrado sozinho numa linha e neste caso os elementos podem estar entre aspas e separados por virgula

```
MsgBox " Teste de mensagem ", 32, "janela de teste"
```

- Quando o comando *MsgBox* precisa retornar um valor, os seus elementos deverão estar entre parêntesis.

```
resultado = MsgBox(msg, botao, titulo)
```

Exemplo de um programa em VBA

Assim considere o seguinte exemplo: *Suponha sejam dados 02 notas e se deseja calcular a soma destas notas, a sua média aritmética e a sua média ponderada com pesos 3 e 7.*

Para a solução deste pequeno problema é necessário inicialmente identificar os dados de entrada:

Dados de Entrada (as duas notas do aluno):

n1 que é do tipo numérico real

n2 que é do tipo numérico real

Dados de Saída(Resultados): (o que é pedido ou solicitado):

- Soma das notas
- O Calculo da média aritmética
- O Calculo da média ponderada

Processamento: As operações que foram solicitadas:

Soma= $n1 + n2$

Media_a= $(n1 + n2) / 2$

Media_p= $n1 * 0,3 + n2 * 0,7$

Para o exemplo dado acima teremos:

Para a declaração das variáveis de entrada de dados::

```
Dim n1 As Double, n2 As Double
```

Para representar as variáveis de saída de dados:

```
Dim soma As Double, media_a As Double, media_p As Double
```

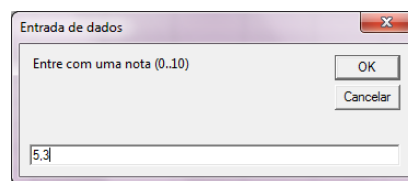
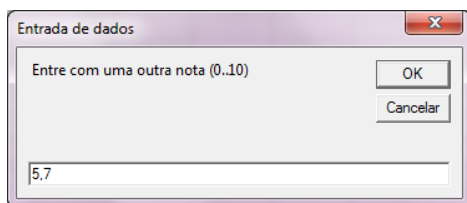
Para representar as mensagens de saída de dados, utilizamos variáveis do tipo String

```
'Declaração de string que serão usados para mostrar os dados no MsgBox
Dim s1 As String, s2 As String, s3 As String
```

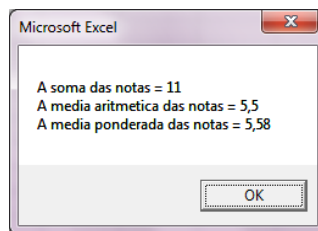
Para representar a leitura de dados, vamos obter os dados usando o comando **Inputbox** e converter os dados lidos **para Double**. Observe que o comando Inputbox realiza a leitura de dados na forma de String, ou seja, o dado digitado para ser convertido para real e armazenado em um variável. Observe a utilização do comando `CDbl()` que converte a expressão entre parênteses para Double, ou seja, um número quebrado

```
'leitura de dados usando o comando Inputbox
n1 = CDbl(InputBox(" Entre com uma nota (0..10) ", "Entrada de dados "))
n2 = CDbl(InputBox(" Entre com uma outra nota (0..10) ", "Entrada de dados "))
```

Observe abaixo que as janelas mostradas podem aceitar tanto dados reais usando virgula ou ponto



Como resultado será mostrada a seguinte mensagem:



Assim, o programa completo em VBA:

```
Sub Aula07()
' declaração de variáveis de Entrada de Dados
Dim n1 As Double, n2 As Double
'Declaração de saída de dados
Dim soma As Double, media_a As Double, media_p As Double

'Declaração de string que serão usados para mostrar os dados no MsgBox
Dim s1 As String, s2 As String, s3 As String

'leitura de dados usando o comando Inputbox
n1 = CDbl(InputBox(" Entre com uma nota (0..10) ", "Entrada de dados "))
n2 = CDbl(InputBox(" Entre com uma outra nota (0..10) ", "Entrada de dados "))

'Operações a serem realizadas
soma = n1 + n2
media_a = (n1 + n2) / 2
media_p = n1 * 0.3 + n2 * 0.7

' Temos que mostra os dados concatenados com as mensagens
s1 = " A soma das notas = " & soma
s2 = " A media aritmetica das notas = " & media_a
s3 = " A media ponderada das notas = " & media_p

' Mostrar somente as mensagem usando o comando MsgBox
' o comando vbCrLf pula uma linha dentro do comando MsgBox
MsgBox s1 & vbCrLf & s2 & vbCrLf & s3

End Sub
```

Observamos que o comando `vbCrLf` utilizado numa string (cadeia de caracteres) faz com que exista uma quebra de linha