

Exemplos Práticos de VBA

1 - Range("cell")

1.

```
Sub t1()  
    Range("A1").Value = "XXX"  
End Sub
```

2.

```
Range("B2").Value = Range("B1").Value  
Range("B3").Value = Range("B2").Value + 7  
Range("B5").Value = 2 * Range("B3").Value + 1
```

3.

```
MsgBox Range("A1").Value  
MsgBox Range("B1").Value + 1  
MsgBox "A célula B5 contém o valor " & Range("B5").Value
```

4. Exercícios

- colocar em C3 a soma dos valores de C1 e C2;
- aumentar 3 ao valor existente na célula C4;
- trocar os valores das células C6 e C7;

2 - ActiveCell

1.

```
Sub t2()  
  MsgBox ActiveCell.Value  
  ActiveCell.Value = 789  
  MsgBox ActiveCell.Value  
End Sub
```

2.

```
Range("K5").Activate
```

3.

```
MsgBox ActiveCell.Address  
Range("K9").Activate  
MsgBox ActiveCell.Address
```

4. Exercícios

- copiar o valor da célula activa para a célula F5;
- colocar na célula activa o dobro do valor que está na célula F9;
- acrescentar 3 ao valor da célula activa;
- colocar na célula activa a junção (&) dos textos existentes nas células A1 e A2;

3 - Offset

1.

```
Sub t3()  
    ActiveCell.Offset(1, 0) = "Abaixo"  
    ActiveCell.Offset(-1, 0) = "Acima"  
    ActiveCell.Offset(0, 1) = "'=>"  
    ActiveCell.Offset(0, -1) = "<="
```

End Sub

2.

```
ActiveCell.Offset(1, 0).Activate
```

3.

```
experimental até estoirar...  
ActiveCell.Offset(-1, 0).Activate
```

4. Exercícios

- escrever X em todas as células à volta da célula activa;
- passar a célula activa para a célula do lado;
- mostrar (com MsgBox) o conteúdo da célula que está acima da célula activa;
- mostrar o endereço da célula que está acima da célula activa;
- preencher a célula activa e as 2 células abaixo dela com números de 1 a 3;

4 - Variáveis

1.

```
Sub t4()  
  
    Dim i  
  
    i = 1  
    MsgBox i  
  
    i = 7  
    Range("M1") = i  
  
    i = 2 * i  
    MsgBox i  
  
End Sub
```

2.

```
Dim i, j  
  
i = Range("C1").Value + 1  
j = i + Range("C2")  
MsgBox "O valor de j fica a ser " & j
```

3.

```
Dim s  
  
s = "F5"  
Range(s).Activate  
Range(s).Value = "eu sou a " & s
```

5 - InputBox

1.

```
Sub t5()  
    InputBox "Escreva na caixinha"  
End Sub
```

2.

```
Dim n  
n = InputBox("Escreva um número na caixinha")  
MsgBox "Escreveu " & n
```

3.

```
Dim valor  
valor = InputBox("Escreva na caixinha")  
ActiveCell.Value = valor
```

4. Exercícios

- ler um número (InputBox) e escrever esse número na célula F9;
- ler dois números (vai aparecer duas vezes a InputBox) e escrever a soma desses dois números na célula activa;

Grupo II

1 - If - Else

- ```
'indica se a nota dada na célula activa dá Aprovado ou Reprovado
Sub g21_1()
 Dim n
 n = ActiveCell.Value
 If n >= 10 Then
 MsgBox "Aprovado"
 Else
 MsgBox "Reprovado"
 End If
End Sub

'outra versão...
Sub g21_1a()
 Dim n as Integer, s as String
 n = ActiveCell.Value
 If n >= 10 Then
 s = "Aprovado"
 Else
 s = "Reprovado"
 End If
 MsgBox s
End Sub
```
- ```
'lê um número; diz se é par ou ímpar;
Sub g21_2()
  Dim n
  n = InputBox("Diga um número")
  If n Mod 2 = 0 Then
    MsgBox "É par."
  Else
    MsgBox "É ímpar"
  End If
End Sub
```
- ```
'ler dois números; escrever nas células B1, B2 por ordem crescente
Sub g21_3()
 Dim a As Integer, b As Integer
 a = InputBox("Diga um número:")
 b = InputBox("Diga outro número:")

 If a <= b Then
 Range("B1").Value = a
 Range("B2").Value = b
 Else
 Range("B1").Value = b
 Range("B2").Value = a
 End If
End Sub
```

#### 4. Exercícios

- ler um número; dizer se é igual ou diferente do número que está na célula activa;
- ler um número; se for ímpar escrever na célula B1; se for par escrever na célula B2;
- um serviço custa 100€ por hora nas primeiras 5 horas e 500€ daí para a frente; dado na célula B1 o número de horas gastas, calcular o custo em B2;

2 - If

**1.**

```
'nova versão...
Sub g22_2()
 Dim s
 s = "Aprovado"
 If ActiveCell.Value < 10 Then
 s = "Reprovado"
 End If
 MsgBox s
End Sub
```

**2.**

```
'nova versão
Sub g22_3()
 Dim a As Integer, b As Integer, c As Integer
 a = InputBox("Diga um número:")
 b = InputBox("Diga outro número:")
 If a > b Then
 'trocar os valores de a e b
 c = a
 a = b
 b = c
 End If
 Range("B1").Value = a
 Range("B2").Value = b
End Sub
```

**3. Exercícios**

- ler um número; inseri-lo na célula activa (se for maior do que o que lá está);
- mostrar (numa janela MsgBox) o primeiro número par maior ou igual a B1;  
Exemplo: se B1 contém 12 mostrar 12; se B1 contém 13 mostrar 14;
- ordenar (por ordem crescente) os números contidos em D1 e D2;
- ler uma palavra (através de InputBox); se a palavra for "Euros" converter o valor de B1 para Euros; se for "Contos" converter o valor de B1 para Contos;

## 3 - If - ElseIf

### 1.

```
'indica se a nota dá Aprovado, Exame ou Oral
Sub g23_1()
 Dim nota
 nota = ActiveCell.Value
 If nota >= 10 Then
 MsgBox "Aprovado"
 Else
 If nota >= 8 Then
 MsgBox "Oral"
 Else
 MsgBox "Exame"
 End If
 End If
End Sub
```

### 2.

```
'outra versão...
Sub g23_2()

 Dim nota
 nota = ActiveCell.Value
 If nota >= 10 Then
 MsgBox "Aprovado"
 ElseIf nota >= 8 Then
 MsgBox "Oral"
 Else
 MsgBox "Exame"
 End If

End Sub
```

### 3. Exercícios

- nas células D1 e D2 está o nome de duas equipas; ao lado o resultado de um jogo; faça uma subrotina que mostre (MsgBox) o nome da equipa vencedora (se houver);
- nas células B1 e B3 são dados dois números; na célula B2 é dado uma sinal de operação ("+", "-", "\*" ou "/"); faça uma subrotina que calcule escreva o resultado da operação em B5; (se for uma divisão por 0 não deve fazer);
- ler uma palavra (através de InputBox); se a palavra lida for "troca" deve trocar o conteúdo das células B1 e B2; se a palavra lida for "B1" deve somar 1 ao valor da célula B1; se a palavra lida for "B2" deve somar 1 à célula B2; se a palavra lida for "mais" deve aumentar 1 à célula que tiver o menor valor (B1 ou B2) e nada fazer se B1 e B2 forem iguais;

## 4 - Operadores lógicos

### 1.

```
'é dada a nota do trabalho na célula activa e a do teste na célula
'ao lado da célula activa; o teste tem nota mínima 7;
'indicar Aprovado/Reprovado, 2 células ao lado da célula activa;
Sub g24_1()

 Dim nota1 As Integer, nota2 As Integer
 Dim nota_final As Integer

 nota1 = ActiveCell.Value
 nota2 = ActiveCell.Offset(0, 1)
 nota_final = (nota1 + nota2) / 2

 If nota2 >= 7 And nota_final >= 10 Then
 ActiveCell.Offset(0, 2) = "Aprovado"
 Else
 ActiveCell.Offset(0, 2) = "Reprovado"
 End If

End Sub
```

### 2.

```
'acertar no conteúdo de K1 ou K2
Sub g24_2()
 Dim s
 s = InputBox("Diga qualquer coisa")
 If s = Range("K1") Or s = Range("K2") Then
 MsgBox "Acertou"
 End If
End Sub
```

### 3. Exercícios

- modificar a subrotina do ponto 2: em vez de Or usar And e Not;
- idem, usando só ifs (nem And, nem Or);

## 5 - Outros Exercícios

### 1.

Faça uma subrotina que coloque a célula activa com fundo Vermelho caso esta contenha um número negativo (ou sem fundo no caso contrário).

Grave uma Macro para ver como é que se faz para pôr uma célula com fundo Vermelho e como é que se faz para tirar a cor de fundo.

### 2.

Há várias formas de ver se uma célula está vazia. Por exemplo:

```
Sub teste()
 MsgBox ActiveCell.Value = ""
End Sub
```

Faça uma subrotina que elimine a célula activa (ie faça o equivalente à operação de menu "Edit|Delete") caso esta se encontre vazia. Grave uma Macro para ver como é que se elimina uma célula.

### 3.

A função IsNumeric() permite ver se determinado valor é um número.

Por exemplo:

```
Sub teste()
 MsgBox IsNumeric(ActiveCell.Value)
End Sub
```

Faça uma subrotina que insira o "número seguinte" na célula activa: se a célula de cima estiver vazia insere o número 1; se a célula de cima contiver um número insere o número seguinte; senão, manda uma mensagem (MsgBox) a dizer que não pode fazer nada.

## **Grupo III**

### **1 - For**

**1.**

```
Sub g31_1()
 Dim i As Integer
 For i = 1 To 3
 MsgBox "um piano."
 Next
End Sub
```

**2.**

```
Sub g31_2()
 Dim i As Integer
 For i = 5 To 10 Step 2
 MsgBox Range("A1").Value
 Next
 MsgBox "Fim."
End Sub
```

**3.**

```
'lê 5 números; asiciona cada um deles a A1;
Sub g31_3()
 Range("A1").Value = 0
 Dim i As Integer, n As Integer
 For i = 1 To 5
 n = InputBox("Diga um número")
 Range("A1").Value = Range("A1").Value + n
 Next
End Sub
```

**4.**

```
'lê 5 números; põe o maior deles em A1;
Sub g31_4()
 Range("A1").Value = 0
 Dim i As Integer, n As Integer
 For i = 1 To 5
 n = InputBox("Diga um número")
 If n > Range("A1").Value Then
 Range("A1").Value = n
 End If
 Next
End Sub
```

### **5. Exercícios**

- repetir 3 vezes: ler um número (com InputBox); mostrar o mesmo número (com MsgBox);
- repetir 3 vezes: ler um número; mostrar o mesmo número se for > 10;
- ler 10 números; indicar na célula B1 quantos são > 10  
(ou seja, repetir 10 vezes: ler um número; somar 1 à célula B1 se o número for >=10);

## 2 - Indexação

**1.**

```
'mostra os números de 7 a 9
Sub g32_1()
 Dim i As Integer
 For i = 7 To 9
 MsgBox i
 Next
End Sub
```

**2.**

```
'mostra o conteúdo das 3 células abaixo da célula activa
Sub g32_2()
 Dim i As Integer
 For i = 0 To 2
 MsgBox ActiveCell.Offset(i, 0)
 Next
End Sub
```

**3.**

```
'preenche, com números dados, as 3 células abaixo de K3;
sub g32_3()
 Dim i As Integer
 Dim x
 For i = 0 To 2
 x = InputBox("Diga qualquer coisa")
 Range("K3").Offset(i, 0) = x
 Next
End Sub
```

**4.**

```
'o mesmo
Sub g32_3a()
 Dim i As Integer, s As String
 Dim x
 For i = 0 To 2
 s = "Para " & ActiveCell.Offset(i, 0).Address & "vai ?"
 x = InputBox(s)
 ActiveCell.Offset(i, 0) = x
 Next
End Sub
```

**5.**

```
'preenche com números a partir de 1
Sub g32_4()
 Dim i As Integer
 For i = 0 To 9
 ActiveCell.Offset(i, 0) = i + 1
 Next
End Sub
```

**6.**

```
'idem, de 2 em 2
sub g32_5()
 Dim i As Integer
 For i = 1 To 10
 ActiveCell.Offset(i - 1, 0) = 2 * i - 1
 Next
End Sub
```

**7.**

```
'o mesmo
Sub g32_5a()
 Dim i As Integer
 For i = 1 To 21 Step 2
 ActiveCell.Offset((i - 1) / 2, 0) = i
 Next
End Sub
```

**8.**

```
'preenche com números, a partir de 1 até ao número dado;
Sub g32_6()
 Dim i As Integer, n As Integer
 n = InputBox("Números até ?")
 For i = 1 To n
 ActiveCell.Offset(i - 1, 0) = i
 Next
End Sub
```

**9.**

```
Sub g32_7()
 Dim inicial As Integer, final As Integer
 inicial = InputBox("Numero inicial ?")
 final = InputBox("Numero final ?")
 Dim i As Integer
 For i = inicial To final
 ActiveCell.Offset(i - inicial, 0) = i
 Next
End Sub
```

## **10. Exercícios**

- preencher as 5 células acima da célula activa com números de 10 a 15;
- ler um número; escrever esse número nas 5 células abaixo da célula activa;
- idem, mas escrever só nas células onde esteja um número menor;
- escrever X ao lado de cada célula do grupo A1:A0 que contenha um número  $\geq 10$ ;

**3 - Range(" : "); Select/Selection; Count;**

**1.**

```
'preenche todas as células do grupo indicado com 77
Sub g33_1()
 Range("B5:C8").Value = 77
End Sub
```

**2.**

```
Sub g33_2()
 MsgBox Selection.Address
 Range("B5:C8").Select
 MsgBox Selection.Address
End Sub
```

**3.**

```
'repete o valor da célula activa em todas as células seleccionadas
Sub g33_3()
 Selection.Value = ActiveCell.Value
End Sub
```

**4.**

```
'propriedade count: indica o número de células de um grupo
Sub g33_5()
 Dim n
 n = Selection.Count
 MsgBox "estão seleccionadas " & n & " células"
End Sub
```

4 – Cells(i)

**1.**

```
Sub g34_1a()
 ActiveCell.Cells(1).Value = "um"
 ActiveCell.Cells(2).Value = "dois"
 ActiveCell.Cells(4).Value = "quatro"
End Sub
```

```
Sub g34_1b()
 ActiveCell.Cells(1, 1).Value = "(1,1)"
 ActiveCell.Cells(1, 2).Value = "(1,2)"
 ActiveCell.Cells(2, 1).Value = "(2,1)"
 ActiveCell.Cells(2, 2).Value = "(2,2)"
 ActiveCell.Cells(3, 3).Value = "(3,3)"
End Sub
```

**2.**

```
'activa a célula cujo número é dado
Sub g34_3()
 Range("B1:C4").Select
 Dim n As Integer
 n = InputBox("Activar a célula ?")
 Selection.Cells(n).Activate
End Sub
```

**3.**

```
'preenche as células seleccionas com números a partir de 1
Sub g34_3xxx()
 Dim i As Integer
 For i = 1 To Selection.Count
 Selection.Cells(i) = i
 Next
End Sub
```

**4.**

```
'aumentar 1 a cada uma das as células seleccionadas
Sub g34_4()
 Dim i As Integer
 For i = 1 To Selection.Count
 Selection.Cells(i).Value = Selection.Cells(i).Value + 1
 Next
End Sub
```

**5.**

```
'indicar a soma das células seleccionadas
Sub g34_5()
 Dim i As Integer, n As Integer
 Dim soma
 soma = 0
 n = Selection.Count
 For i = 1 To n
 soma = soma + Selection.Cells(i)
 Next i
 MsgBox "Soma = " & soma
End Sub
```

**6. Exercícios**

- indicar o endereço da primeira e da última célula seleccionada;
- ler um número; inserir em todas as células seleccionadas (menos a primeira e a última);
- preencher as células seleccionadas com número pares a partir de 2;
- ler um número; somar a cada uma das células seleccionadas;
- indicar quantas das células seleccionadas têm um valor maior que 10;
- ler um número; indicar se e quantas vezes aparece nas células seleccionadas;

## **Grupo IV**

### **1 - Set; For Each**

**1.**

```
Sub g41_1()
 Dim c As Range

 Set c = Range("A2")
 MsgBox c.Address

 Set c = Range("B5")
 MsgBox c.Address

 Set c = ActiveCell
 MsgBox c.Address
End Sub
```

**2.**

```
Sub g41_2()

 Dim c As Range

 For Each c In Range("A1:B2")
 MsgBox c.Address
 Next

End Sub
```

**3.**

```
Sub g41_3()

 Dim c As Range
 Dim soma As Integer

 For Each c In Selection
 soma = soma + c.Value
 Next
 MsgBox soma

End Sub
```

#### **4. Exercícios**

- adaptar g41\_3 para calcular a média;
- aumentar 1 a todas as células seleccionadas
- aumentar 1 a todas as células seleccionadas que contenham um número < 20;
- aumentar 1 a todas as células seleccionadas contendo um valor < que a média;

### **2 - Funções**

**1.**

Experimente a seguinte função em fórmulas na folha de cálculo.

P.ex: =g42\_iva( 100; 17%)  
=g42\_iva( A1; 17%)

```
Function g42_iva(valor, taxa)
 g42_iva = valor * (1 + taxa)
End Function
```

## 2.

Experimente a seguinte função em fórmulas na folha de cálculo.

P.ex: = g42\_soma ( A1:B2 )

```
Function g42_soma(rng As Range)

 Dim c As Range

 g42_soma = 0
 For Each c In rng
 If IsNumeric(c) Then
 g42_soma = g42_soma + c.Value
 End If
 Next

End Function
```

## 3. Exercícios

- faça uma função **MinhaCat** que junte o texto de um grupo de células; por exemplo:  
= MinhaCat (C1:C3) daria o mesmo que =A1 & A2 & A3
- faça uma função **MeuCountIf** que conte as células com valor igual ao indicado no segundo argumento; por exemplo:  
=MeuCountIf(A1:B2, K1) daria o mesmo que =COUNTIF(A1:B2, K1)

## 3 - WorkSheetFunction

### 1.

```
Sub g43_1()

 Dim x
 x = ActiveCell.Value
 x = Application.WorksheetFunction.Round(x, 0)
 MsgBox x

End Sub
```

### 2.

```
Sub g43_2()

 Dim c As Range
 Dim x

 Set c = Range("A1:B2")
 x = Application.WorksheetFunction.Sum(c)
 MsgBox x

End Sub
```

### 3. Exercícios

- arredondar para duas casas decimais todos os números das células seleccionadas;
- suponha que está seleccionada uma tabela do tipo pauta;  
fazer uma macro que dado um número de aluno mostre o respectivo nome  
(os números são indicados na coluna 1 da tabela e os nomes na coluna 2)

## 4 - Outros exercícios

### 1. Acrescentar funções do grupo "countif"

- a) Fazer uma função AVERAGEIF que funcione, basicamente como um SUMIF / COUNTIF.
- b) Fazer uma função COUNTIF\_OP que receba o termo de comparação num argumento e o valor noutro. Por Exemplo: =COUNTIF\_OP( A1:B2; ">"; K1 )  
teria o efeito de contar o número de células com valor superior ao valor existente em K1;

### 2. Pauta

Suponha que temos uma pauta. Queremos seleccionar a coluna das notas e fazer uma macro que ponha as notas negativas com fundo cinzento e as notas acima de 16 com fundo azul.

### 3. Detectar repetições

Temos uma coluna de dados, na qual queremos assinalar as repetições. O plano é o seguinte: seleccionamos os dados; fazemos uma macro que põe um X ao lado de cada nome repetido.

## **Grupo V**

### **1 - While**

**1.**

```
Sub g51_1()
 Dim x As Integer
 Do
 x = InputBox("Diga um número: ")
 Loop While x <> 7
End Sub
```

**2.**

```
Sub g51_2()
 Dim x
 x = InputBox("Diga um número: ")
 Do While Not IsNumeric(x)
 x = InputBox("Eu disse, UM NÚMERO !!!")
 Loop
End Sub
```

**3.**

```
Sub g51_3()
 Dim x
 x = ""
 Do
 x = InputBox("Diga um número (ou CANCEL)")
 Loop While x <> 7 And x <> ""
End Sub
```

**4.**

Preencha as células A2:A10 com números.

O programa seguinte lê um número e posiciona a célula activa no número lido.

```
Sub g51_4()
 Dim num As Integer
 num = InputBox("Diga um número: ")

 Dim c As Range
 Set c = Range("A2")
 Do While c.Value <> num
 Set c = c.Offset(1, 0)
 Loop
 c.Activate
End Sub
```

## 2 - Procura

**1.**

O programa anterior tem um problema: se o número não existir nunca mais pára (ou pára só quando atingir a última célula).

Altere a condição para

```
while c.value <> x and c.Value <> ""
```

O que acontece, agora, se o número não existir ?

Faça com que apareça uma mensagem (MsgBox) se o número não existir.

**2.**

Suponha que temos uma coluna de números de A2 para baixo.

Faça um programa que posicione a célula activa na primeira célula vazia abaixo de A2.

**3.**

Suponha que tem uma pauta.

Faça um rotina que peça um número e nome (com InputBox) e acrescente no fim da Pauta.

**4.**

Para a mesma Pauta: suponha que os números de aluno estão por ordem crescente. Faça um programa que peça um numero (InputBox) e posicione a célula activa no primeiro número maior (ou igual) ao número dado.

**5.**

Para a mesma Pauta: suponha que os números de aluno estão por ordem crescente. Faça um programa que peça um numero e nome de um aluno e: se o número já existir altere o nome do aluno; senão insira o novo aluno na pauta, por ordem.

## Grupo VI

### 1 - Operações com Range

1. Descreva cada um das seguintes propriedades/métodos do objecto Range. Em caso de dúvida faça uma subrotina para experimentar.

Value, Text, Formula, Address  
Activate, Select  
Clear, ClearContents, ClearFormat  
Count  
Cells, Offset

2. A operação Copy permite copiar um grupo de células para outro. Grave uma macro com uma operação de cópia para ver um exemplo.

Um forma expedita de usar o Copy é indicar, de imediato, o destino da cópia. Por Exemplo:  
Range("A1:A3").Copy Range("B1")  
copia as células A1:A3 para a célula B1.  
Faça uma macro que copie a célula de cima para a célula activa.  
Qual o efeito se a célula copiada tiver uma fórmula ?

3. A propriedade Resize(N,M), aplicada a um Range, dá como resultado outro Range com a mesma origem e dimensão N x M. P. Exemplo

Range ("A1").Resize (2, 2)      o mesmo que Range ("A1:B2")  
Range ("A1:K7").Resize (1, 2)    o mesmo que Range ("A1:B1")  
ActiveCell.Resize (10, 10)      10x10 células a partir da célula activa

Suponha que está selecciona apenas uma célula.  
Faça uma Macro que estender a selecção a todas células em volta dessa.  
Ex: se a célula activa for K5, o resultado será a selecção de J4:L6.

4. A macro anterior pode dar problemas se a célula active estiver na primeira linha ou na primeira coluna. A propriedade .Row indica o número da linha de uma célula; identicamente .Column indica o número da coluna.

Altere a macro do exercício anterior para evitar os problemas na 1ª linha / coluna.

5. Aplicando .Rows.Count a um Range dá o número de linhas. Identicamente .Columns.Count dá o número de colunas. Por exemplo:

Range ("A1:K3").Rows.Count      dá 3  
Range ("A1:K3").Columns.Count    dá 11

Suponha que está seleccionado um grupo de células qualquer.  
Faça uma macro que deixe seleccionada apenas a primeira coluna das células anteriormente seleccionadas.

Por exemplo:

B2:K7      passaria a      B2:B7

## 2 - Exercícios Variados

### 1.

Grave uma macro que ponha uma célula com fundo vermelho.  
Dará qualquer coisa do género:

```
With Selection.Interior
 .ColorIndex = 3
 .Pattern = xlSolid
End With
```

O with serve apenas para não ter que escrever Selection.Interior até ao End With.  
Ou seja, a parte que mais interessa pode-se escrever:

```
Selection.Interior.ColorIndex 3
```

Um das formas de pôr cor é usar as cores configuradas (numeradas de 1 a ...).  
Faça uma subrotina que produza uma tabela de cores, ou seja: uma coluna com números de 1 a 25; ao lado uma célula com a cor correspondente.

### 2.

Temos uma folha com uma Pauta.  
A Pauta tem os cabeçalhos na linha 1 (Número, Nome, Turma, Nota Trabalho, Nota Teste, Nota Final) e os dados imediatamente a seguir.

Pretende-se fazer uma macro ponha fundo cinzento linha sim linha não.  
Ou seja, ficaria A2:F2 a fundo cinzento; A3:F3 a fundo branco; A4:F4 a fundo cinzento; e assim sucessivamente até ao fim da pauta.

### 3.

Faça duas Macros, uma que ordene a pauta por número, outra que ordene a pauta por nome do aluno. (Grave uma macro para ver como é que se faz a ordenação em VB).

Inclua em ambas a chamada à macro anterior para voltar a pôr as linhas a cinzento alternadamente.