

Macrocomandos (Macros)

O que é uma macro? São programas que executam tarefas específicas, automatizando-as.

Quando uma macro é activada, executa uma sequência de instruções.

Tipos de macros:

- Macros de comandos
- Macros de funções

Macros de comandos

Repetição de tarefas

- É frequentemente necessário executar a mesma tarefa, que podem ser células de um intervalo, folhas de cálculo de um livro ou diferentes livros de uma aplicação.
- Embora não seja possível ao gravador de macros gravar ciclos, consegue gravar a tarefa principal de modo a ser possível a sua repetição.

Macros de comandos Tarefas simples

No exemplo que segue é criada uma macro que muda a cor das células com valores negativos.

Passo 1: Criar a macro

Ferramentas / Macro / Gravar nova macro/ Nome

Passo 2: Formatar os números

Formatar / Células/ Padrões/

Passo 3: Terminar a gravação

Utilizar o botão 'Parar' na barra de ferramentas 'Terminar gravação'.

Passo 4: Executar a macro

Ferramentas / Macro / Macros / Executar

Macros de comandos

Visualizar o conjunto de instruções que constituem a macro (código VBA): Ferramentas / Macros/ Macro/ Editar

Sub Pintar()

' Macro gravada em 04/6/2000

With Selection.Interior

.ColorIndex = 6

.Pattern = xlSolid

End With

End Sub

Macros de comandos

Automatize as seguintes tarefas:

- Inserir o seu nome, turma, alinhados á esquerda,e o número da página, alinhado á direita, no rodapé da folha de cálculo. Atribua-lhe uma tecla de atalho.
- Gravar ficheiro actual na disquete. Atribua-lhe um botão com uma imagem elucidativa.
- 3. Configurar a página para impressão com os seguintes parâmetros:

Margem Ed^a=Margem Dt^a =2cm ; Cabeçalho =Rodapé=3 cm

Tamanho do papel = A4, Orientação=Horizontal; Folha sem grelha;

Atribua-lhe um botão com a descrição: Configuração de página.

Macros de Funções

Funções definidas pelo utilizador

Existem cálculos bastante específicos que não podem ser executados por nenhuma das funções predefinidas. Quando ocorre esta situação é necessário que o utilizador crie a sua própria função.

Para definir uma função é necessário:

- Nome da função
- Argumentos
- Fórmulas

Macros de Funções

Funções definidas pelo utilizador

Para criar uma função é necessário:

Passo1: utilizar o editor de texto do Visual Basic.

Ferramentas/Macro/Editor do Visual Basic

Passo2: inserir módulo

Insert/Module

Passo3: iniciar a função com a palavra Function e terminar com a palavra End Function.

Exemplo do cálculo do valor do Iva

Function IVA (Valor, Taxa)
IVA = Valor* Taxa
End Function

Macros de Funções

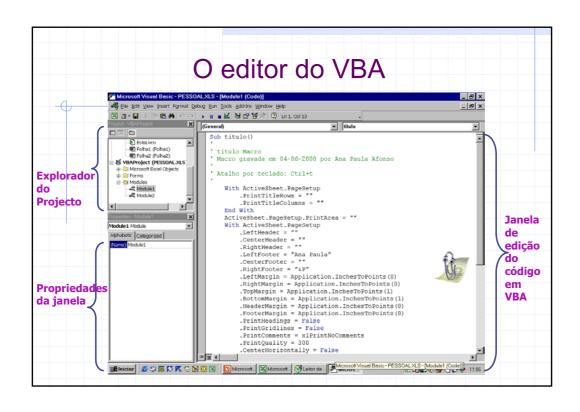
Funções definidas pelo utilizador

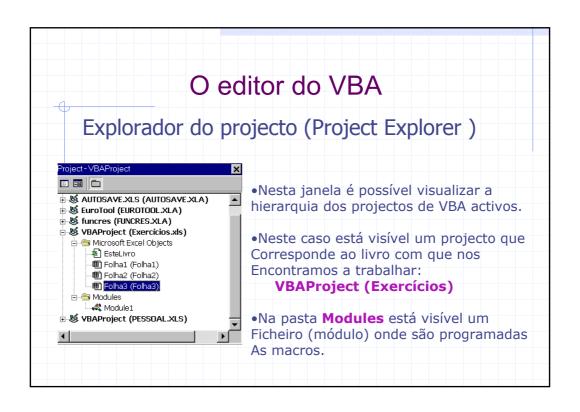
Crie funções que permitam efectuar as seguintes operações:

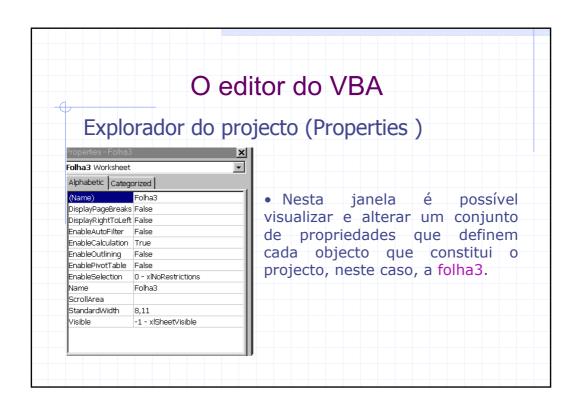
- 1. Calcular a classificação final á disciplina de INF II, em que a fórmula aplicada é a seguinte:
 - a. Classificação= 80% * PAC + 20% * TRAB
- Repetir a operação mas para uma situação em que são desconhecidas as percentagens atribuídas às PAC e ao TRAB (Classificação= X% * PAC + Y%* TRAB)
- 3. Calcular o valor comercial de um artigo com IVA
- 4. Calcular o valor comercial de um artigo com desconto

Resumo

- Activar o gravador de macros
- Desactivar o gravador de macros
- Operações sobre macros:
 - Executar
 - Editar
 - Alterar
 - Eliminar
 - Atribuir uma tecla de atalho ou um botão







Colecções de Objectos e Objectos

O que são Objectos?

São elementos caracterizados por um conjunto de propriedades e que apresentam um determinado comportamento. P.ex.: uma folha de cálculo é um objecto, tem um nome, um conjunto de linhas e colunas, uma grelha que pode ser desactivada, pode ser protegida contra escrita, a altura das linhas e a largura das colunas pode ser modificada,...

Objectos: Propriedades e Métodos

Propriedades

Constituem o conjunto de características que o definem. P.ex.: nome, cor,dimensão, valor contido, ... (Luisa Domingues). As propriedades determinam a aparência e o comportamento dos objectos.

Métodos

São acções que os objectos podem executar. Cada objecto pode ter associados vários métodos.

As acções desencadeadas pelos métodos podem alterar as propriedades dos objectos.

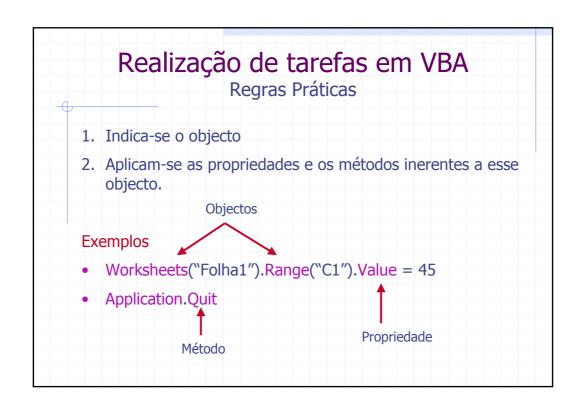
P.ex.: Fechar um livro de Excel

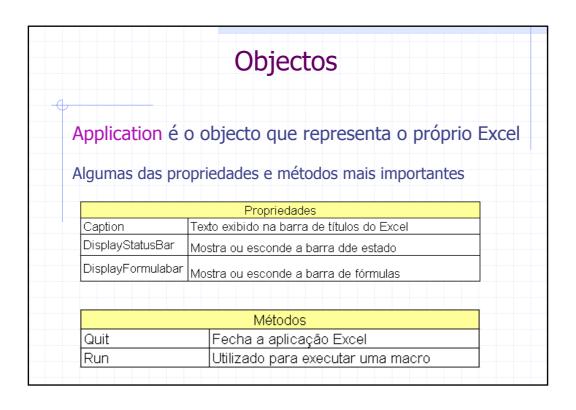
Objectos: Colecções

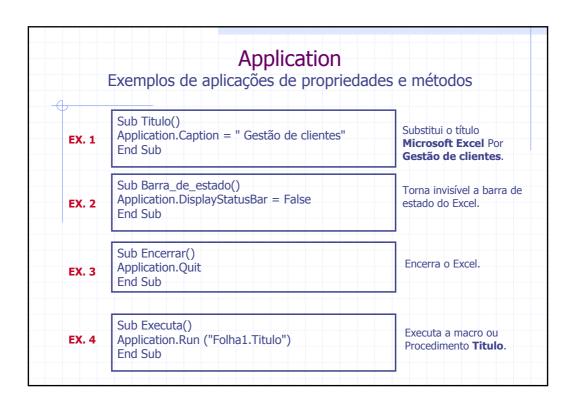
- ♦ As colecções são conjuntos de objectos relacionados. Cada objecto dentro de uma colecção é um elemento dessa colecção.
- ◆ Uma colecção é também um objecto, com as suas propriedades e métodos.

Por exemplo uma colecção que agrupa todas as folhas de cálculo de um determinado livro é um objecto que existe em Excel, denominado Worksheets. Possui várias propriedades p.ex.: Count, que devolve o número de elementos dessa colecção (José António Carriço).

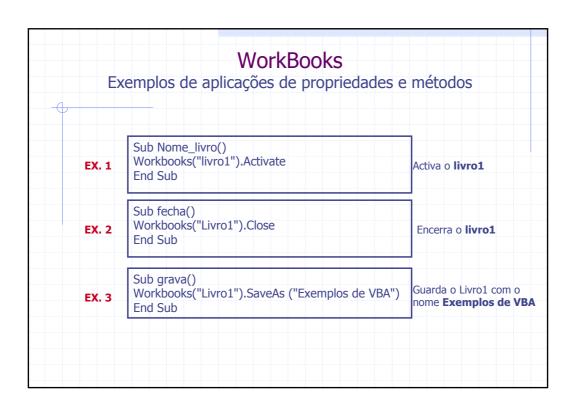
Hierarquia dos principais Objectos em Excel Application Workbooks Worksheets Range O objecto Application contém a colecção Workbooks. O Objecto Workbooks contém entre outros o objecto Worksheets, que por sua vez contém entre outros o objecto Range.



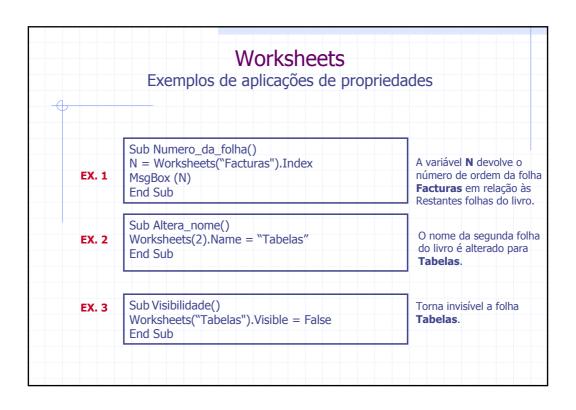




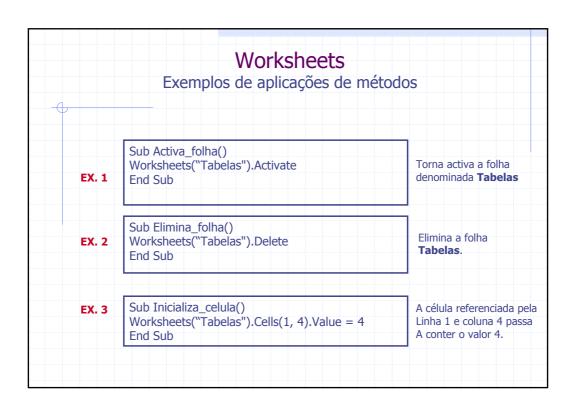
	Objectos
Vorkbooks repr	esenta um ficheiro de Excel
Alguns	dos métodos mais importantes
	Métodos
Activate	Activa um documento aberto
Close	Fecha um documento
Protect	Protege um documento contra alterações
Save	Grava o livro



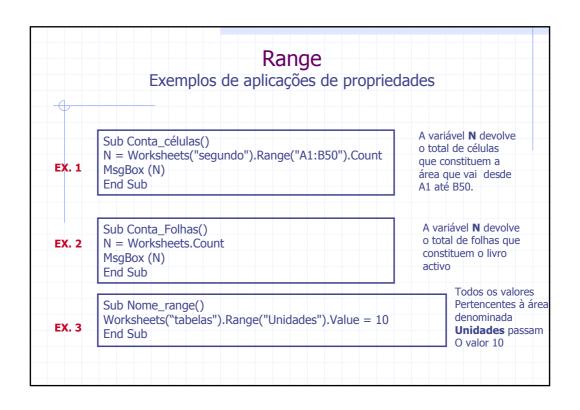
	Objectos
Vorksh	eets é uma folha do objecto Workbooks
	Algumas das propriedades mais importantes
Index	Propriedades
Index Name	

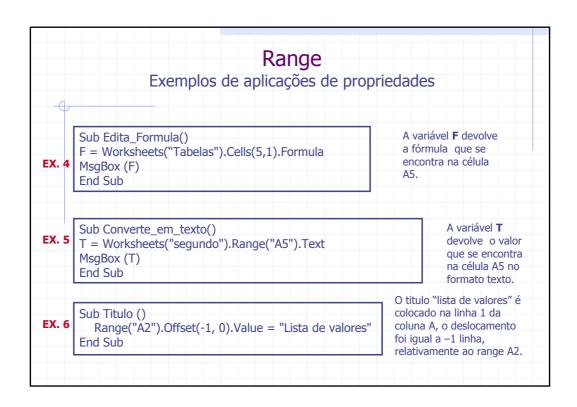


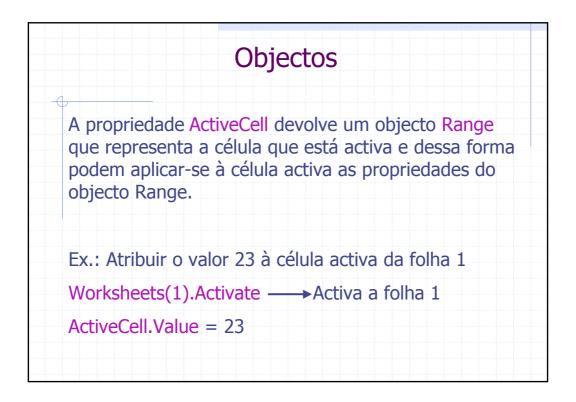
	Worksheets
	Alguns dos métodos mais importantes
	Métodos
Activate	Activa um determinada Worksheet
Delete	Apaga uma Worksheet
Cells	Referencia uma célula através dos seus índices numéricos (linha,coluna)

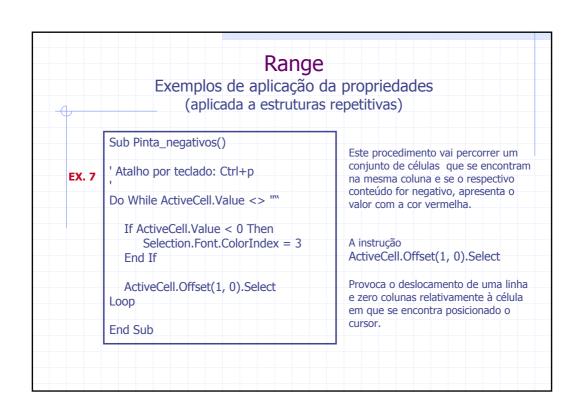


	Objectos
	objecto utilizado para representar uma ou as de uma Worksheet
Al	gumas das propriedades mais importantes
	Propriedades
Count	Conta o número de células num range
Name	Nome de um range
Value	Conteúdo de uma célula ou conjunto de células
Formula	Edita a fórmula contida num range como um conjunto de caracteres
Text	Devolve o conteúdo de uma célula em formato texto
Offset	Provoca o deslocamento de um determinado número de linhas e de colunas, tendo como base um range.

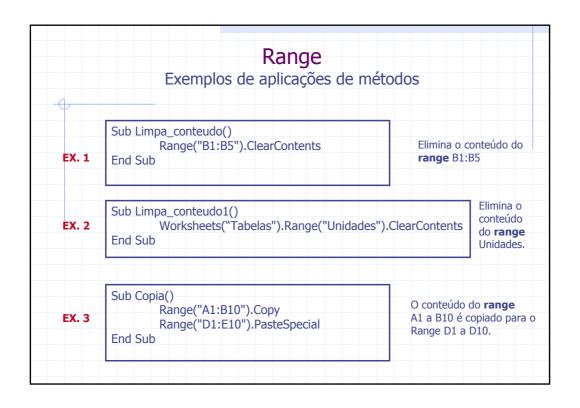


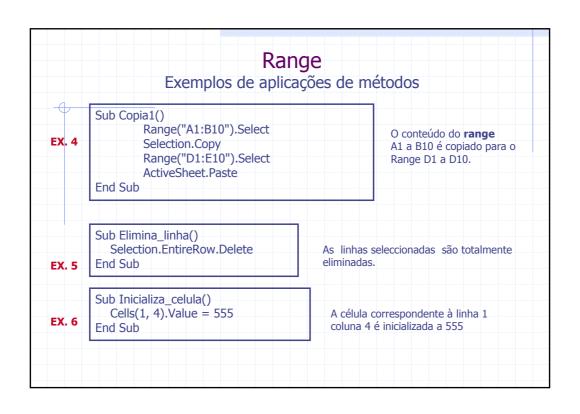






Objectos				
Range - Alguns dos métodos mais importantes				
	Métodos			
ClearContents	Apaga o conteúdo (fórmulas e valores) de uma célula deixando os formatos			
Сору	Copia o conteúdo de um range para outro de igual dimensão			
EntireRow	Faz referência à(s) linha(s) inidicadas num range			
Select	Selecciona um range			
Cells	Referencia uma célula através dos seus índices numéricos.			





Controlos Personalizados (Botões de comando)

Os botões de comando respondem a um único evento: o clique. O qual deverá permitir a execução de um macrocomando (procedimento).

Existem diferentes tipos de controlo incluindo, Caixa de verificação, Botão de opção, Botão, etc.

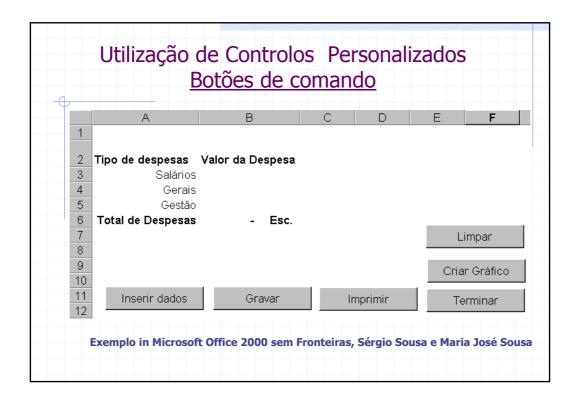
Barra de Ferramentas de Formulários

As ferramentas que podem ser utilizadas para desenhar os controlos encontram-se na barra de ferramentas Formulários:

• Ver /Barras de Ferramentas/ Formulários



Criar Controlos • Seleccionar o controlo e desenhá-lo no local pretendido. Formatação de controlos Cada controlo tem as suas propriedades e para fazer a sua alteração é necessário seguir os passos seguintes: Passo1: Seleccionar o controlo Passo2: Formatar/Controlo/Formatar controlo Passo3: Indicar/Alterar as propriedades pretendidas na caixa de diálogo que surge no ecrã.



Utilização de Controlos Personalizados Botões de comando

Passo1: Criar o quadro de despesas na folha de cálculo

Passo2: Criar os botões de comando

Ver/Barras de Ferramentas/ Formulários/Botão

Passo3: Atribuir os Procedimentos aos botões de Comando

- Construir os seguintes procedimentos:
 - Inserir / Gravar / Imprimir / Limpar / Criar Gráfico / Terminar

Procedimentos (VBA) Exemplos

1. Inserção de Dados

Sub inserir()

Dim salario, gerais, gestao As Long

salario = InputBox("Introduza a despesa Salário", "Introdução de dados") gerais = InputBox("Introduza a despesa Gerais", "Introdução de dados") gestao = InputBox("Introduza a despesa Gestão", "Introdução de dados")

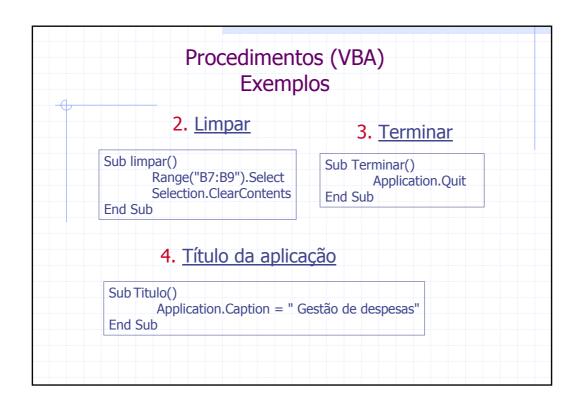
Application. Cells(7, 2) = salario

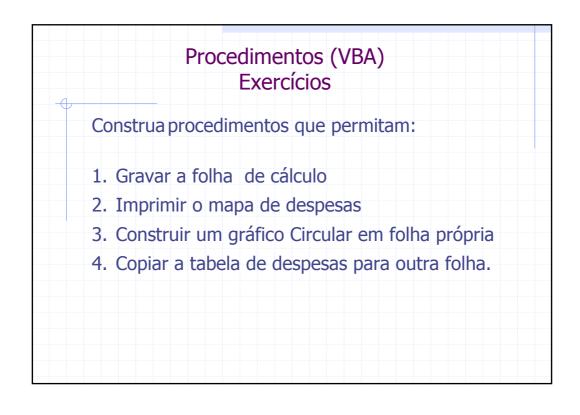
Application.Cells(8, 2) = gerais

Application.Cells(9, 2) = gestão

End Sub

Exemplo in Microsoft Office 2000 sem Fronteiras, Sérgio Sousa e Maria José Sousa





Bibliografia

- Microsoft Office 2000 sem fronteiras, Sérgio Sousa, Maria José Sousa
- ◆ Manual de VBA, Luísa Domingues (ISCTE)
- Microsoft Excel 2000 VBA Fundamentals, Reed Jacobson
- ♦ O Guia Prático do Excel 2002, Ana Paula Afonso