

- Aide-mémoire de Visual Basic -

Commentaires

introduits par une apostrophe (') ou **Rem**. Ce qui suit sur la ligne est ignoré.

Variables et constantes

Déclaration (optionnelle, type *Variant* par défaut) avec **Dim** *nomVar* [**As** *type*] ou **Dim** *nomVar* suivi d'un suffixe. Le nom des variables est insensible à la casse.

Spécification de la portée à la déclaration en utilisant **Public** ou **Private** au lieu de **Dim**

Déclaration d'une constante personnalisée : **Const** *nomConst* [**As** *type*] = *expression*

Nombreuses constantes prédéfinies ex. *vbCrLf* ⇔ nouvelle ligne, *vbTab* ⇔ tabulation, ...

Tableaux

Déclaration avec **Dim|Private|Public** *nomTab* ([*indiceMin1 To*] *indiceMax1* , [*indiceMin2 To*] *indiceMax2*,...)

L'indice minimum est 0 (par défaut) ou 1 (avec l'instruction **Option Base 1** au niveau module)

Tableaux dynamiques : déclaration avec **Dim** *nomTab*() [**As** *type*], redimensionnement avec **ReDim** [**Preserve**] *nomTab* (*indices*) et utilisation de **UBound**() pour connaître l'indice maximum.

Chaînes

Constantes entourées de guillemets doubles (") et concaténation par l'opérateur &.

Affichage dans la fenêtre d'exécution par **Debug.Print**, dans une boîte de message par **MsgBox** ().

Fonctions : longueur **Len** (), sous-ch. **Left** (), **Right** () ou **Mid** (), position de sous-ch. **InStr** (), en maj. **UCase** ()

Opérateurs

= (égalité), <> (différence), **And** (ET logique), **Or** (OU logique), **Not** (NON logique), **Mod** (modulo)

Structures de contrôle

Test		Sélection	
If <i>condition</i> Then <i>instructions</i> 'Sur une seule ligne If <i>condition1</i> Then <i>instructions1</i> [Elseif <i>condition2</i> Then <i>instructions2</i> [Elseif <i>condition3</i> Then <i>instructions3</i> ... [Else <i>instructionsElse</i> End If		Select Case <i>expressionTestée</i> Case <i>listeValeurs1</i> <i>instructions1</i> Case <i>listeValeurs2</i> <i>instructions2</i> ... [Case Else <i>instructionsElse</i> End Select	
Boucles FOR		Boucles WHILE et UNTIL	
For <i>compteur</i> =début to fin [step pas] <i>Instructions</i> Next <i>compteur</i> For Each <i>element</i> In groupe <i>instructions</i> Next <i>element</i>		Do While <i>condition</i> <i>Instructions</i> Loop	Do Until <i>condition</i> <i>Instructions</i> Loop
		Do <i>Instructions</i> Loop While <i>condition</i>	Do <i>Instructions</i> Loop Until <i>condition</i>

Fonctions

Déclaration : **Sub** *nomProcédure* ([*arguments*]) ou **Function** *nomFonction* ([*arguments*]) [**As** *type*]

Arguments (syntaxe d'un argument) : [**Optional**] [**ByVal** | **ByRef**] *nomArg* [**As** *type*] [= *valDéfaut*]

Appel d'une procédure **Sub** : **Call** *nomProcédure* (*arg1*, *arg2*, ...) ou *nomProcédure* *arg1*, *arg2*, ...

Appel d'une fonction : - sans utiliser le résultat : *nomFunction* *arg1*, *arg2*, ... ;
 - en utilisant le résultat : *nomVar* = *nomFunction* (*arg1*, *arg2*, ...)

Utilisation des noms des arguments : [**Call**|*nomVar* =] *nomMacro* [(|*nomArg2* := *valArg2*, *nomArg1* := *valArg1*,... [|])