

## Excel – Macros

Il existe deux types de macros :

- Les macros enregistrées
- Les macros programmées

Les macros enregistrées sont plus abordables que les macros programmées mais offrent beaucoup moins de possibilités. Ce document présente donc uniquement le concept des macros programmées.

Pour pouvoir travailler avec les macros, il faut activer l'onglet « Développeur » sur le ruban. Pour cela, rendez-vous dans le menu « Fichier », puis « Options ». Choisissez la rubrique « Personnaliser le ruban ». Activez ensuite l'onglet « Développeur ».

## Contrôles

L'onglet « Développeur » vous permet d'insérer des contrôles (boutons, listes déroulante...).

Chacun des contrôles peut être associé aux données présentes sur Excel (exemple : liste déroulante) ou alors associé à une macro (exemple : bouton).

## **Interface Visual Basic**

L'onglet « Développeur » vous permet d'accéder à l'interface Visual Basic.

Dans la colonne de gauche on trouve la fenêtre du projet. Cette dernière contient les différents objets qui peuvent utilisés. Il peut s'agir d'objet Excel comme d'objets propres à Visual Basic.

Par défaut, vous disposez de l'objet Feuil1 qui représente la feuille Excel disponible dans votre classeur. L'objet ThisWorlbook représente le classeur Excel en lui-même.

En double-cliquant sur l'objet Feuil1, vous faites apparaître les différents morceaux de code en rapport avec l'objet. Dans la fenêtre ouverte il y a deux listes déroulantes. Dans celle de gauche, vous devez sélectionner « Worksheet ». Dans celle de droite, il y a les différents événements qui peuvent survenir sur l'objet. Vous pouvez par exemple choisir l'événement « Activate ». Dans l'interface, la procédure « Worksheet\_Activate » apparaît alors. On reconnaît qu'il s'agit d'une procédure grâce au mot-clé « Sub ». La fin de la procédure est repérée par « End Sub ». Il est maintenant possible d'écrire du code Visual Basic dans cette procédure afin qu'il soit exécuté lorsque la feuille est sélectionnée (au niveau des onglets).



## Langage Visual Basic

En Visual Basic on retrouve les structures conditionnelles et les boucles comme dans n'importe quel langage de programmation.

Structure conditionnelle

```
IF [TEST] THEN [INSTRUCTION]
```

```
IF [TEST] THEN
[INSTRUCTIONS]
```

END IF

IF [TEST] THEN [INSTRUCTION] ELSE [INSTRUCTION]

IF [TEST] THEN [INSTRUCTIONS]

ELSE

[INSTRUCTIONS]

END IF

Les tests peuvent utiliser les symboles =, >, <, <> (différent) et les connecteurs AND, OR, ...

Boucle « pour »

FOR i=1 TO 100 [INSTRUCTIONS]

NEXT La variable i va prendre les valeurs : 1, 2, 3, 4, ..., 99, 100

FOR i=0 TO 50 STEP 2 [INSTRUCTIONS]

NEXT La variable i va prendre les valeurs : 0, 2, 4, 6, ..., 48, 50 Boucle « tant que »

WHILE [TEST] [INSTRUCTIONS] WEND

Les cellules

Il est intéressant de pouvoir obtenir le contenu de certaines cellules. Vous pouvez utiliser l'instruction suivante pour manipuler une ou plusieurs cellules :

RANGE("A1").SELECT : sélectionne la cellule A1 RANGE("A1:C5").SELECT : sélectionne la plage qui va de la cellule A1 à la cellule C5 RANGE("B1").CLEAR : efface le contenu de la cellule B1 RANGE("C1").DELETE : supprime la cellule C1 et décale les autres cellules RANGE("A1").VALUE : représente la valeur de la cellule A1.

On peut ainsi récupérer le contenu d'une cellule dans une variable Visual Basic : a = RANGE("A1").VALUE Il est aussi possible de modifier le contenu d'une cellule : RANGE("A1").VALUE = 45