

VBA

Introduction

VBA (Visual Basic for Applications) est un langage de programmation intégré aux logiciels Microsoft Office.

Il permet d'automatiser des tâches répétitives et de créer des applications personnalisées au sein d'Excel, Word ou Access.

Cette procédure guide la création, l'exécution et la sauvegarde d'une macro VBA simple.

Objectif

- Créer, modifier et exécuter des macros VBA dans Excel (ou Word, Access, etc.).

Prérequis

- Microsoft Office installé
- Fichier Excel (ou autre) prêt
- Connaissance basique du besoin (exemple : automatiser une tâche répétitive)

Activer l'onglet Développeur

- Ouvrir Excel.
- Aller dans Fichier > Options > Personnaliser le ruban.
- Cocher la case Développeur dans les onglets principaux.
- Valider avec OK.

Créer une nouvelle macro simple

- Cliquer sur Développeur > Visual Basic ou Développeur > Macros.
- Dans l'éditeur VBA (VBE) :
 - Insertion > Module.
 - Écrire ou coller votre code VBA dans le module ouvert.

Exemple de Macro très simple :

```
Sub Bonjour()  
    MsgBox "Bonjour, bienvenue sur Excel!"  
End Sub
```

Exécuter une macro

- Retourner sur Excel.
- Aller dans Développeur > Macros.
- Sélectionner votre macro (Bonjour par exemple).
- Cliquer sur Exécuter.

Enregistrer un classeur avec macro

- Fichier > Enregistrer sous.
- Type de fichier : Classeur Excel avec macros (*.xlsm).

VBA

Éléments supplémentaires utiles

Élément	Description
Sub / End Sub	Définissent le début et la fin d'une macro.
Dim	Déclare une variable (ex : Dim i As Integer).
If...Then...Else	Logique conditionnelle.
For...Next ou Do...Loop	Boucles pour répéter des actions.
MsgBox	Affiche une boîte de message.
Range, Cells	Accéder aux cellules ou plages de cellules Excel.

Remarques importantes

- Sécurité : Par défaut, Excel bloque les macros pour éviter les virus → Il faut autoriser les macros dans Fichier > Options > Centre de gestion de la confidentialité.
- Sauvegardes : Toujours faire une copie de votre fichier avant de tester un nouveau code.
- Débogage : Utiliser F8 pour exécuter ligne par ligne et voir où se produit une erreur.

Exemple Complet : Boucle For pour remplir une colonne

```
Sub RemplirColonne()  
    Dim i As Integer  
    For i = 1 To 10  
        Cells(i, 1).Value = "Ligne " & i  
    Next i  
End Sub
```

Ce script remplit les 10 premières cellules de la colonne A avec "Ligne 1", "Ligne 2", etc.

Fiche Résumé - Actions Rapides

Action	Où cliquer
Activer Développeur	Fichier > Options > Personnaliser Ruban
Ouvrir l'éditeur VBA	Développeur > Visual Basic
Insérer un Module	VBE > Insertion > Module
Exécuter une macro	Développeur > Macros

Bonus : Conseils

- Toujours commenter (' commentaire) votre code pour comprendre plus tard.
- Décomposer votre code en petites Sub (procédures) pour faciliter la maintenance.
- Penser à enregistrer régulièrement pendant le développement.