

Chapitre 4

Structures de contrôle

Durée : 1 heure 55

Mots-clés

condition, choix, alternative, branchement conditionnel, compteur, itération, incrémenter, décrémenter, sortie, boucle, test

Objectifs

Maîtriser les structures de décision afin de tester des conditions puis effectuer des actions différentes selon le résultat obtenu. Maîtriser les instructions d'itération qui, associées aux instructions conditionnelles, permettent d'écrire du code Visual Basic pour la prise de décision et la répétition des actions. Vous retrouverez ces structures dans la suite du livre. Dans ce chapitre, nous nous limitons à l'emploi des boîtes de dialogue déjà rencontrées.

Prérequis

Pour valider les prérequis nécessaires, avant d'aborder le TP, répondez aux questions ci-après (certaines questions peuvent nécessiter plusieurs réponses) :

1. Les structures suivantes sont des structures de décision :
 - a. If ... Then ... Else ... End If
 - b. Do ... Loop
 - c. Select Case ... Case ... End Select
2. Résultat = IIf(7 / 2 > 3, IIf(2.8 * 3.3 < 11, "X", "Y"), "Z")
La variable Résultat contient la valeur :
 - a. X
 - b. Y
 - c. Z

3. Les instructions suivantes appartenant chacune à une structure de contrôle différente sont correctes :
 - a. Case If N1 > N2
 - b. Case A, B, C
 - c. Case 1 to 10
 - d. Case Nombre, Is > 50
4. Le mot-clé ElseIf :
 - a. peut apparaître de suite après une clause Else.
 - b. est facultatif.
 - c. peut être utilisé plusieurs fois dans un bloc If.
5. Répétition d'instructions tant qu'une condition a la valeur True :
 - a. For Each ... Next
 - b. For ... Next
 - c. Do ... Loop
 - d. While ... Wend
 - e. With ... End With
6. Utilisation d'un compteur pour exécuter des instructions un certain nombre de fois :
 - a. For Each ... Next
 - b. For ... Next
 - c. Do ... Loop
 - d. While ... Wend
 - e. With ... End With
7. Répéter un groupe d'instructions pour chaque élément d'un tableau ou d'une collection :
 - a. For Each ... Next
 - b. For ... Next
 - c. Do ... Loop
 - d. While ... Wend
 - e. With ... End With

8. Répéter un groupe d'instructions le nombre de fois indiqué :
 - a. For Each ... Next
 - b. For ... Next
 - c. Do ... Loop
 - d. While ... Wend
 - e. With ... End With
9. Exécuter une série d'instructions appliquées à un seul objet ou à un type défini par l'utilisateur :
 - a. For Each ... Next
 - b. For ... Next
 - c. Do ... Loop
 - d. While ... Wend
 - e. With ... End With

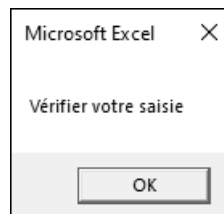
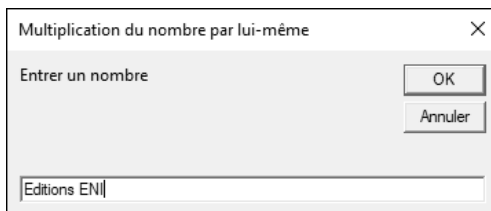
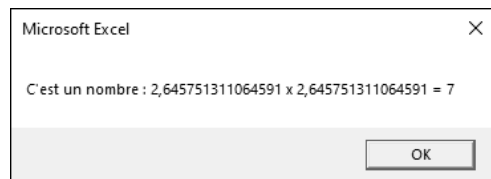
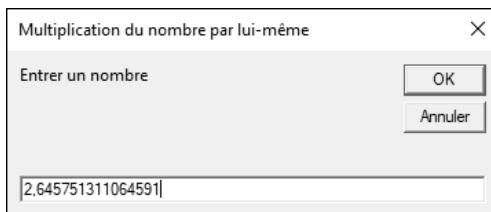
Corrigé p. 271

Énoncé 4.1 Vérifier que la saisie est numérique

Exercice 1

Durée estimative : 10 minutes

Créez la procédure **ContrôleSaisie** qui propose une boîte de saisie et vérifiez qu'il s'agit d'un nombre. Si c'est le cas, effectuez l'opération qui consiste à multiplier le nombre par lui-même, sinon affichez un message. Exemple :



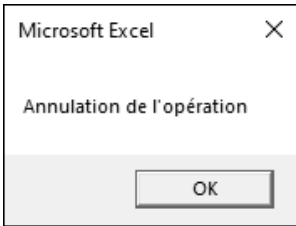
Indice

Utilisez la fonction `VBA.IsNumeric` pour tester la saisie.

Exercice 2

Durée estimative : 5 minutes

Complétez la procédure précédente qui vérifie que l'utilisateur a cliqué sur le bouton **OK**. Dans le cas contraire, affichez un message. Exemple :

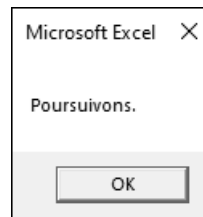
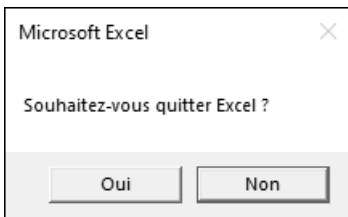
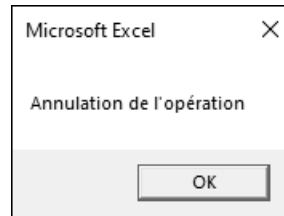
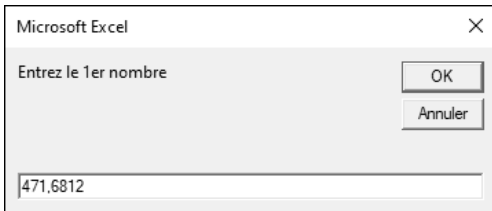


Corrigé p. 271

Énoncé 4.2 Dire au revoir

Durée estimative : 5 minutes

Complétez la procédure **AuRevoir**. Celle-ci propose une boîte de dialogue demandant si l'on souhaite ou non quitter Excel. Elle affiche le message "A bientôt!" si l'utilisateur clique sur le bouton **Oui**, et le message "Poursuivons" sinon. Exemple :



```
Sub AuRevoir()  
    Dim strMessage As String  
    Dim intStyle As Integer  
    Dim intChoix As Integer  
    strMessage = "Souhaitez-vous quitter Excel ?"  
    intStyle = vbYesNo + vbDefaultButton2  
    ... = MsgBox(strMessage, intStyle)  
    If ... = vbYes Then  
        MsgBox "A bientôt."  
        ...  
        MsgBox "Poursuivons."  
        ...  
End Sub
```

Indice

Voici un extrait de l'aide VBA en ligne. L'aide complète se trouve à l'adresse suivante : <https://docs.microsoft.com/fr-fr/office/vba/language/reference/user-interface-help/msgbox-function>

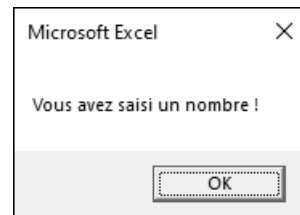
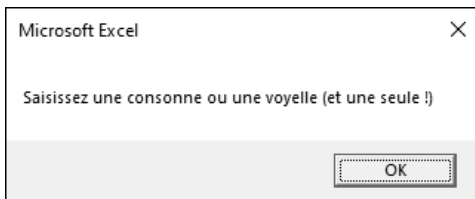
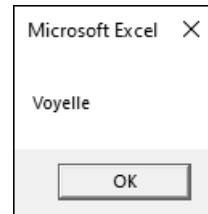
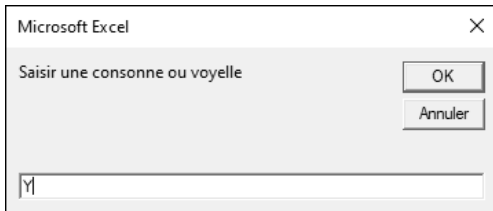
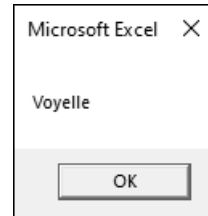
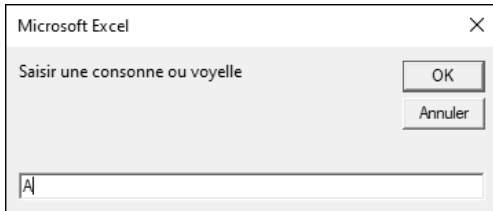
Constante	Valeur	Description
vbOKOnly	0	Bouton OK uniquement (par défaut)
vbOKCancel	1	Boutons OK et Annuler
vbAbortRetryIgnore	2	Bouton Abandonner, Réessayer et Ignorer
vbYesNoCancel	3	Boutons Oui, Non et Annuler .
vbYesNo	4	Boutons Oui et Non

Corrigé p. 272

Énoncé 4.3 Contrôler la saisie d'une consonne ou voyelle

Durée estimative : 10 minutes

Créez la procédure **ConsonneVoyelle** qui demande de saisir une voyelle ou une consonne. Affichez un message selon la saisie : "Voyelle", "Consonne" ou un message approprié en cas d'erreur. Exemple :



Indice

Pour tester la lettre en majuscule, on effectuera la transformation suivante :

```
varChoix = VBA.UCase(varChoix)
```

Corrigé p. 272

Énoncé 4.4 Afficher un message selon l'âge et le genre

Durée estimative : 15 minutes

Créez la procédure **VotreAge** qui demande le sexe et l'âge de l'utilisateur. Les messages affichés seront différents selon l'âge et selon qu'il s'agisse d'un homme ou d'une femme. Prenez également en considération l'abandon de la saisie et la saisie de lettres pour l'âge.

Pour les hommes :

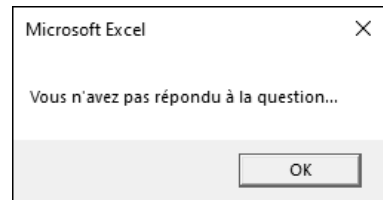
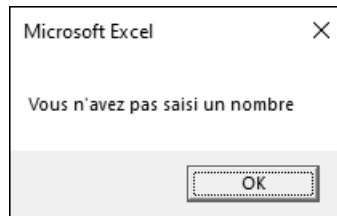
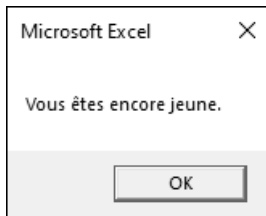
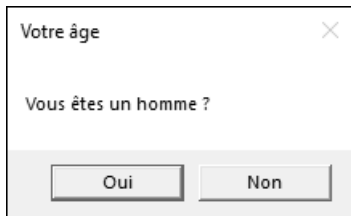
0 - 17 ans : "Gamin"

18 - 30 ans : "Vous êtes jeune"

31 - 50 ans : "Vous êtes encore jeune"

> 50 ans : "Vous commencez à vieillir"

Pour les femmes, un seul message : "On ne demande pas son âge à une femme". Exemple :



Indices

Déclaration des variables :

```
Dim strHomme As String * 1
Dim strFemme As String * 1
Dim intChoixSexe As Integer
Dim dblAge As Double
```

Pour afficher la première boîte de dialogue :

```
intChoixSexe = MsgBox("Vous êtes un homme ?", vbYesNo, "Votre âge")
```
