

Programmation Impérative

Structures conditionnelles

Contenu du cours

- Introduction
- Composantes des expressions conditionnelles
 - Opérateurs relationnels
 - Opérateurs logiques
- Structure If Then
- Structure If Then Else
- Structure If Then ... Elself ... Else ...
- Structure de sélection: Select Case

Introduction

- Comment rédiger des **expressions conditionnelles** ?
- Comment utiliser une instruction **If...Then** ?
- Comment court-circuiter une instruction **If...Then** ?
- Comment utiliser une instruction **Select Case** pour faire une sélection ?

Introduction

- Nous avons déjà introduits les instructions de base comme par exemple, l'affectation (=), des opérateurs divers (ex: arithmétiques), la lecture et l'écriture permettant de réaliser des programmes dans lesquels l'exécution était séquentielle :
 - Les instructions s'y exécutaient dans l'ordre où elles étaient écrites.
- Durant l'exécution d'un programme il est possible d'avoir à effectuer **des choix** dans le traitement à réaliser.
 - Par exemple, nous pourrions avoir à déterminer le pourcentage de la remise en fonction de l'ancienneté du client et du montant de la facture.

Structure conditionnelle (de choix)

Exemple

- Écrire un programme qui lit une variable X entrée au clavier et qui affiche si la valeur de cette variable est positive ou négative.
 - Lire (X)
 - Si $X > 0$ alors écrire `valeur positive`
 - sinon écrire `valeur négative ou nulle`
- Si la condition ($X > 0$) exprimée après le mot **si** est vraie, on exécute l'instruction après le mot **alors** ; si la condition est fausse, on exécute l'instruction après le mot **sinon**.

Composantes des expressions conditionnelles

- Une **expression conditionnelle** est une **partie d'une instruction** de programmation complète qui pose une question, dont la réponse est Vraie (**True**) ou fausse (**False**), à propos d'une **variable**, d'une **propriété** ou d'une autre donnée du programme.
 - Une expression conditionnelle **simple** consiste en la comparaison de deux expressions de même type
 - Ex: $X > 100$, $\text{Prix} = 10$, $\text{Caractere} = \text{``M``}$, $\text{Val} < 50.5$

Composantes des expressions conditionnelles: Opérateurs relationnels

Opérateur	Signification numérique	Signification caractère
=	Égal à	Égal à
<>	Différent de	Différent de
<	Inférieur à	Placé avant dans l'ordre alphabétique
>	Supérieur à	Placé après dans l'ordre alphabétique
<=	Inférieur ou égal à	Placé avant dans l'ordre alphabétique ou égal
>=	Supérieur ou égal à	Placé après dans l'ordre alphabétique ou égal

Composantes des expressions conditionnelles: Opérateurs logiques

- Une expression conditionnelle **complexe** est formée de plusieurs conditions simples reliées entre elles par des opérateurs logiques **And, Or, Xor, Not**
- Exemples:
 - `(Val > 50.5) And (Moyenne = 74)`
 - `Caractere > ``A`` And Caractere < ``D```

Composantes des expressions conditionnelles: Opérateurs logiques...

Opérateur	Signification
And	Si les deux expressions conditionnelles sont True , le résultat est True .
Or	Si l'une des expressions est True , le résultat est True .
Not	Si l'expression conditionnelle est False (/TRUE), le résultat est True (/ False).
Xor	Si l'une, et seulement l'une, des expressions conditionnelles est True , le résultat est True . Sinon, le résultat est False .

Composantes des expressions conditionnelles: Opérateurs logiques ...

- Les expressions qui peuvent être évaluées comme **True** ou **False** sont appelées des **expressions booléennes**, et le résultat True ou False peut être attribué à une **variable** ou une **propriété booléenne**.
 - Dim Flag1, Flag2, Flag3 As Boolean
 - Flag1 = True
 - Flag2 = False
 - Flag3 = (Flag2 And Flag1)

Remarque

- Lorsqu'un programme évalue une expression mêlant des types différents (des opérateurs de type différent), il évalue d'abord :
 1. Les opérateurs Mathématiques
 2. Les opérateurs relationnels
 3. Les opérateurs logiques
- If $A+B > C$ or Flag

Structure If ...Then

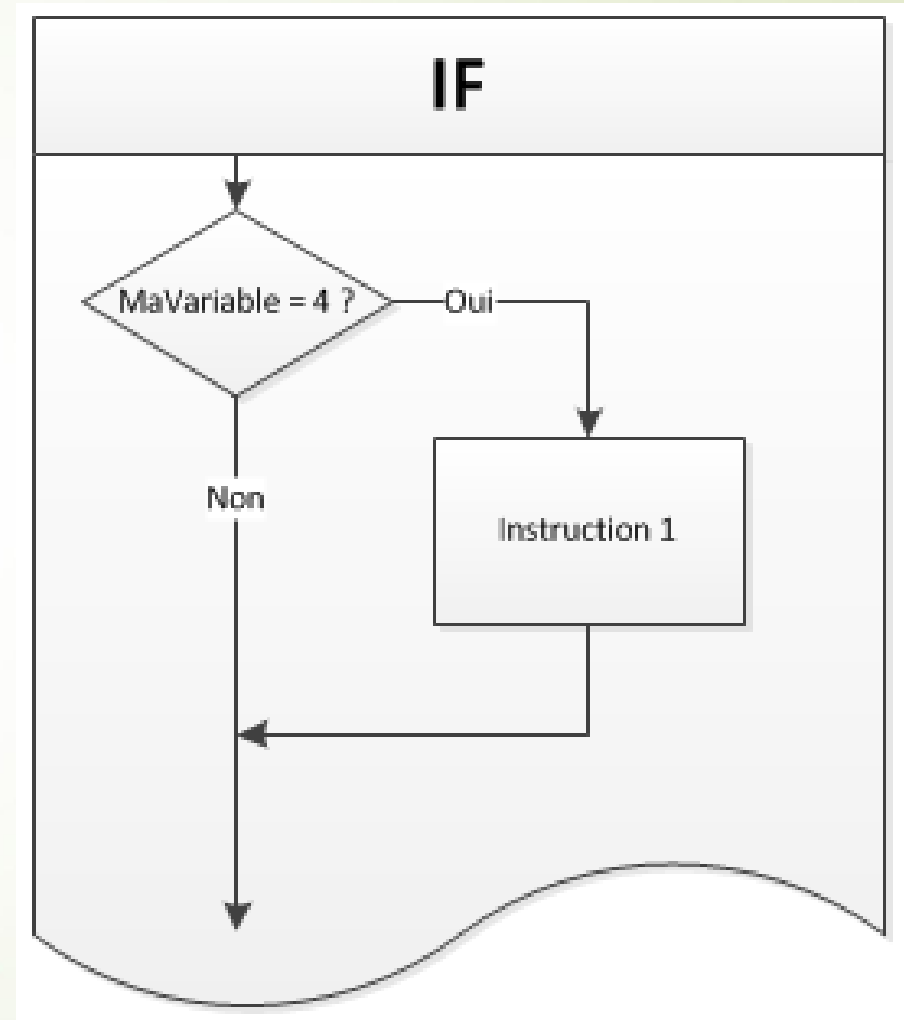
- Les structures de décision **If...Then** introduisent des éléments logiques dans les programmes.
- On utilise une structure de décision **If...Then** pour évaluer une condition dans le programme et définir une action en fonction du résultat.
 - **If** *Condition* **Then** *Instruction* où condition est une expression conditionnelle et instruction est une instruction de programmation
 - **If** Moyenne >= 94 **Then** Cote = ``A+``

Structure If ...Then

If mavariable = 4 Then

Instruction 1

End If



Structure If ...Then ... Else

If condition Then instruction

Else instruction ` Cette partie est optionnelle

End If

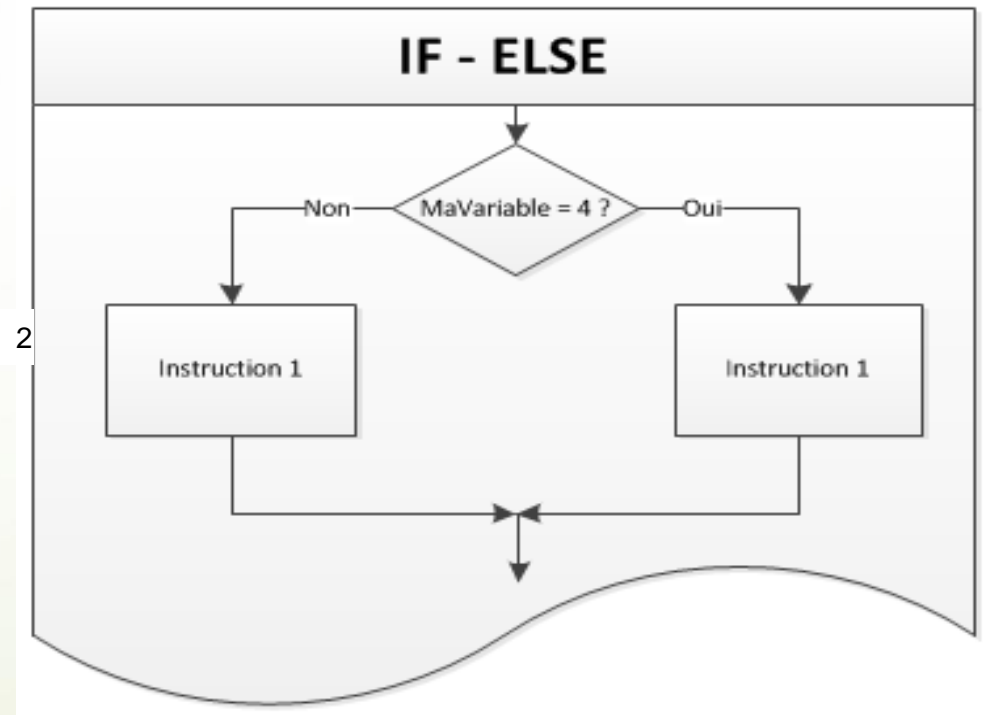
If mavariable = 4 Then

Instruction 1

Else

Instruction 2

End If



Structure conditionnelles imbriquées:

Structure If ...Then ... Elself Else

- Visual Basic permet aussi la création de structures de décision **If...Then** pouvant contenir à son tour **plusieurs** expressions conditionnelles.
- Ce bloc d'instructions peut contenir plusieurs lignes de code et contient les mots clés **Elself**, **Else** et **End If**.

Structure conditionnelles imbriquées:

Structure If ...Then ... Elself Else (syntaxe)

```
If condition1 Then instruction1  
Elself condition2 Then instruction2  
[ Elself conditionN Then instructionN ]  
Else instruction  
End If
```

- ➡ Si aucune des expressions conditionnelles (condition1, ..., conditionN) n'est **True** alors ce sont les instructions sous le mot **Else** qui sont exécutées.

Structure conditionnelles imbriquées: Structure If ...Then ... Elself Else

If mavariable = 4 Then

Instruction 1

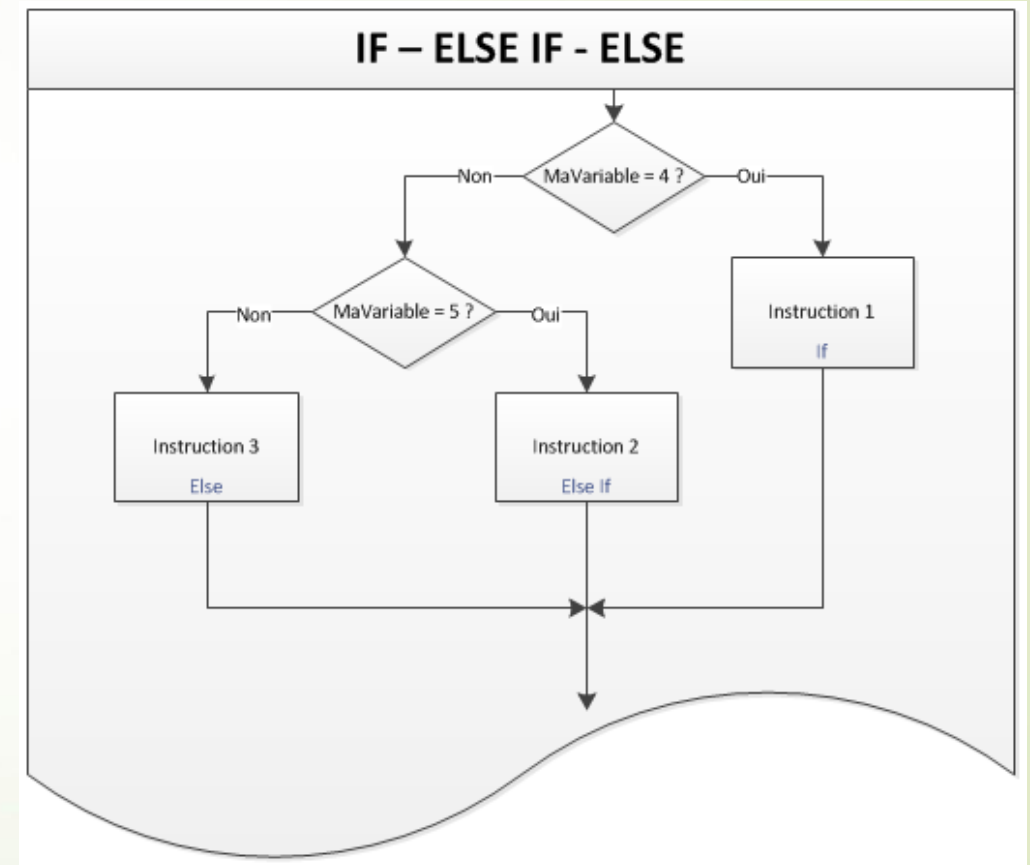
Elself mavariable = 5 Then

Instruction 2

Else

Instruction 3

End If



Structure If ...Then ... Else

➡ Exemple :

```
Dim X As Integer
```

```
X = InputBox(``Entrer la valeur de X``)
```

```
    If X > 0 Then
```

```
        MsgBox(``X est positive``)
```

```
    Else
```

```
        MsgBox(``X est négative ou nulle``)
```

```
End If
```

Structure conditionnelles imbriquées: Structure If ...Then ... Elself Else (Exemple)

```
Dim X As Integer
```

```
X = InputBox("`Entrer la valeur de X`")
```

```
If X > 0 Then MsgBox("`X est positive`")
```

```
Elself X < 0 Then MsgBox("`X est négative`")
```

```
Else MsgBox("`X est nulle`")
```

```
End If
```

Rappel: courts-circuits AndAlso et OrElse

- VB .NET propose deux nouveaux opérateurs logiques qu'on utilise dans les instructions conditionnelles : **AndAlso** et **OrElse**

If Cond1 **AndAlso** Cond2 **Then** Instruction1

Else Instruction2

End If

- Si Cond1 est **False** alors le programme passe directement à la clause **Else** soit Instruction2 sans évaluer la condition Cond2.

OrElse

```
If Cond1 OrElse Cond2 Then Instruction1  
Else Instruction2  
End If
```

- ➡ Si Cond1 est **True**, VB .NET commencera directement à exécuter les instructions de la structure **If** soit Instruction1.

Structure de sélection: Select Case

- Elle est semblable aux structures `If...Then...Else`
- Elle est plus efficace lorsque le branchement dépend de la valeur d'une variable-clé, ou valeur de test.
- C'est une structure qui est plus lisible qu'une structure `If...Then` équivalente.

Structure de sélection: Select Case (Syntaxe)

Select Case Variable

Case Valeur1

instructions1 `exécutée si Variable = Valeur1

Case Valeur2

instructions2 `exécutée si Variable = Valeur1

Case Valeur3

instructions3 `exécutée si Variable = Valeur1

...

Case Else

instructions `si aucune correspondance n'est trouvée

End Select

Structure de sélection: Select Case (Exemple)

```
Dim Age As Integer
```

```
Age = 25
```

```
Select Case Age
```

```
    Case 16
```

```
        MsgBox(``Vous avez le droit de conduire aux États-unis``)
```

```
    Case 18
```

```
        MsgBox(``Vous avez le droit de voter``)
```

```
    Case 21
```

```
        MsgBox(``Vous avez le droit de boire de l'alcool aux États-unis ``)
```

```
    Case 65
```

```
        MsgBox(``Il est temps de prendre votre retraite``)
```

```
    Case Else
```

```
        MsgBox(``Vous êtes dans votre bel age! Profitez-en``)
```

```
End Select
```


Opérateurs relationnels et structure Select Case (Exemple)

```
Select Case Age
```

```
Case Is <13 'Vérifier si l' Age est inférieur à 13
```

```
MsgBox("Profiter de votre jeunesse")
```

```
Case 13 To 19 'Vérifier un intervalle de valeurs (Ages)
```

```
MsgBox("Profiter de votre adolescence")
```

```
Case 21
```

```
MsgBox("Vous avez le droit de boire de l'alcool aux USA")
```

```
Case Is > 100
```

```
MsgBox("Vous avez l'air en forme !")
```

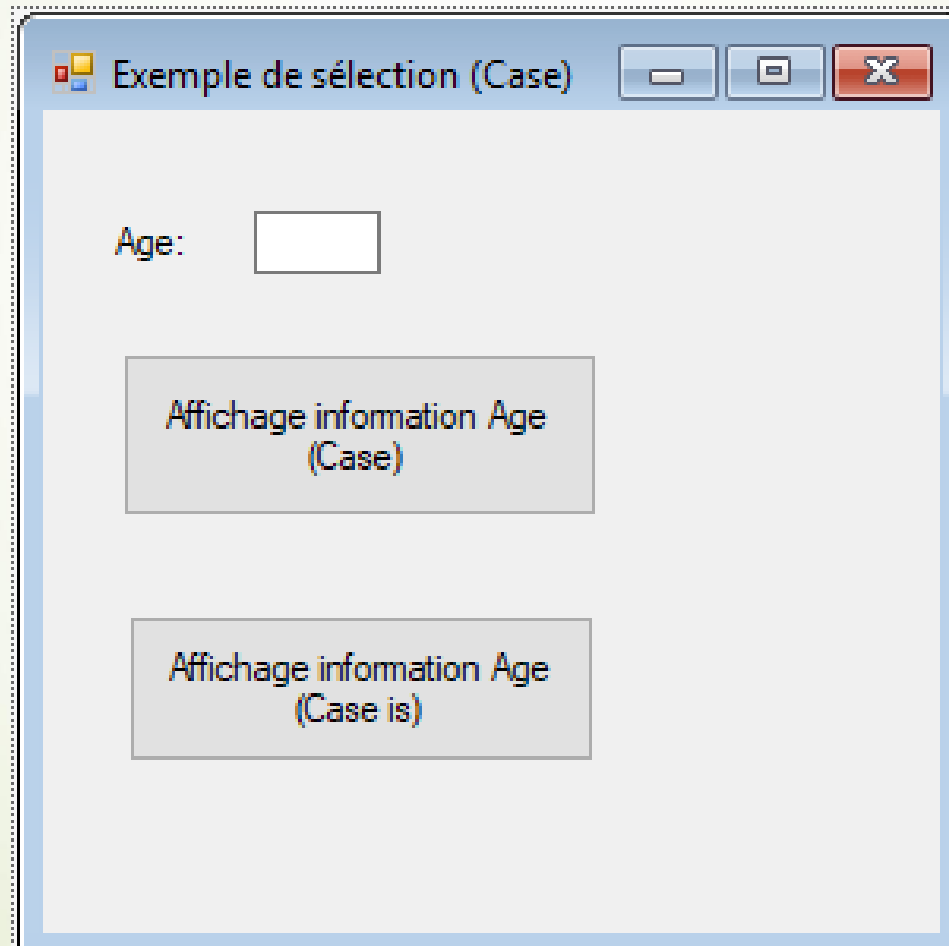
```
Case Else
```

```
MsgBox("Vous êtes dans votre bel age!")
```

```
End Select
```

Select Case: Exemple programmé

➡ Interface de l'application



Select Case: Exemple programmé

VBNET-EX6

➡ Code du bouton (Case)

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim Age As Integer
    Age = CInt(TextBoxAge.Text)
    Select Case Age
        Case 16
            MsgBox("Vous avez le droit de conduire aux États-unis")
        Case 18
            MsgBox("Vous avez le droit de voter")
        Case 21
            MsgBox("Vous avez le droit de boire de l'alcool aux États-unis")
        Case 65
            MsgBox("Il est temps de prendre votre retraite")
        Case Else
            MsgBox("Vous êtes dans votre bel age! Profitez-en")
    End Select
End Sub
```

Select Case: Exemple programmé

VBNET-EX6

➡ Code du bouton (Case Is)

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    Dim Age As Integer
    Age = CInt(TextBoxAge.Text)
    Select Case Age
        Case Is < 13      'Vérifier si l' Age est inférieur à 13
            MsgBox("Profiter de votre jeunesse")
        Case 13 To 19     'Vérifier un intervalle de valeurs (Ages)
            MsgBox("Profiter de votre adolescence")
        Case 21
            MsgBox("Vous avez le droit de boire de l'alcool aux USA")
        Case Is > 100
            MsgBox("Vous avez l'air en forme pour votre age!")
        Case Else
            MsgBox("Vous êtes dans votre bel age!")
    End Select
End Sub
```