



Calculatrice

Table des matières

I- Introduction.....	2
II- Réalisation	3
III-Conclusion.....	4
IV-Annexes	5
Code Commenté	5

I- Introduction

En ce début de deuxième année, le premier projet qu'il nous a été demandé de programmer était une calculatrice. Il ne nous était pas spécifié de langage particulier à utiliser, tant que celui-ci permettait la programmation orienté objet.

Plusieurs consignes ont cependant été demandées pour la réalisation du projet, il y a notamment :

- La présence d'un I.H.M (Interface Homme Machine) qui fait référence à un tableau de bord qui va permettre à l'utilisateur de communiquer avec le programme.
- La possibilité de réaliser des opérations avec des nombres positifs ainsi qu'avec des nombres négatifs.
- La possibilité d'utiliser des nombres à virgule.
- Les priorités de calculs, c'est-à-dire les parenthèses avant les multiplication/division eux-mêmes avant les additions/soustractions.
- Une isolation du cas impossible provoqué par la division d'un nombre par zéro, avec nécessairement une boucle try/catch.
- Un nombre d'opérandes illimités.
- Une touche scientifique.

II- Réalisation

Pour ce projet, j'avais tout d'abord choisi de partir sur du c#, afin de me familiariser avec l'IHM de ce langage. Cependant, il n'est pas possible de computer dans une même textbox en c#, ce qui rendait impossible soit la réalisation de la priorité de calcul, soit du nombre d'opérandes illimités.

J'ai ensuite trouvé la ligne de commande :

```
textcase1.Text = New DataTable().Compute(textcase1.Text, Nothing)
```

qui permet de réaliser l'ensemble des opérations d'une même textbox, en incluant les priorités de calcul et un nombre illimité d'opérandes. J'ai donc décidé de recommencer mon programme en VB cette fois-ci.

J'ai tout d'abord commencé par définir mon IHM comme ci-dessous :

Contenant l'ensemble des boutons nécessaire (les chiffres, les opérandes, les parenthèses ainsi qu'un bouton égal et une touche pour effacer le contenu de la textbox).

J'ai ensuite attribué du code comme celui-ci à chacun des boutons :

```
textcase1.Text += "0"  
textcase1.Text += "*"   
textcase1.Text += "."
```

incrémentant la case textcase1 contenant l'ensemble des chiffres et opérandes par le chiffre ou l'opérateur que l'on souhaite rajouter à l'opération.

Il a enfin fallu incrémenter la touche égal pour qu'elle réalise le calcul dans la textbox tout en mettant en place la boucle try :catch pour la division par zéro :

```
textcase1.Text = New DataTable().Compute(textcase1.Text, Nothing)  
  
    If textcase1.Text = "∞" Then  
        Try  
  
            textcase1.Text = textcase1.Text / 0  
  
        Catch ex As Exception  
  
        Finally  
  
            textcase1.Text = "Erreur"  
  
        End Try  
  
    End If
```

III-Conclusion

Il était ici question de se familiariser avec l'IHM, ainsi que d'approfondir encore plus notre expérience en développement en créant cette calculatrice. Bien qu'elle ne soit pas aussi performante que celle proposée par Windows par exemple, elle pourrait quand même être utilisée dans la vie de tous les jours.

IV-Annexes

CODE COMMENTÉ

```
Public Class Form1

    ' bouton chiffre
    Private Sub Button1_Click_1(sender As Object, e As EventArgs) Handles Button1.Click

        textcase1.Text += "1"

    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click

        textcase1.Text += "2"

    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click

        textcase1.Text += "3"

    End Sub

    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click

        textcase1.Text += "4"

    End Sub

    Private Sub Button5_Click(sender As Object, e As EventArgs) Handles Button5.Click

        textcase1.Text += "5"

    End Sub

End Class
```

```

Private Sub Button6_Click(sender As Object, e As EventArgs) Handles Button6.Click

    textcase1.Text += "6"

End Sub

Private Sub Button7_Click(sender As Object, e As EventArgs) Handles Button7.Click

    textcase1.Text += "7"

End Sub

Private Sub Button8_Click(sender As Object, e As EventArgs) Handles Button8.Click

    textcase1.Text += "8"

End Sub

Private Sub Button9_Click(sender As Object, e As EventArgs) Handles Button9.Click

    textcase1.Text += "9"

End Sub

Private Sub Button10_Click(sender As Object, e As EventArgs) Handles Button10.Click

    textcase1.Text += "0"

End Sub

' bouton clear

Private Sub cmdreset_Click(sender As Object, e As EventArgs) Handles cmdreset.Click

    textcase1.Clear()

End Sub

```

```
' bouton opérateurs

Private Sub cmdplus_Click(sender As Object, e As EventArgs) Handles cmdplus.Click

    If textcase1.Text = "" Then

        textcase1.Text = textcase1.Text

    Else

        textcase1.Text += "+"

    End If

End Sub

Private Sub cmdmoins_Click(sender As Object, e As EventArgs) Handles cmdmoins.Click

    If textcase1.Text = "" Then

        textcase1.Text += "-"

    Else

        textcase1.Text += "-"

    End If

End Sub

Private Sub cmdmulti_Click(sender As Object, e As EventArgs) Handles cmdmulti.Click

    If textcase1.Text = "" Then

        textcase1.Text = textcase1.Text

    Else

        textcase1.Text += "*"

    End If
```



```

End Sub

Private Sub cmddivision_Click(sender As Object, e As EventArgs) Handle
s cmddivision.Click

    If textcase1.Text = "" Then

        textcase1.Text = textcase1.Text

    Else

        textcase1.Text += "/"

    End If

End Sub

' bouton virgule

Private Sub cmdvirgule_Click(sender As Object, e As EventArgs) Handles
cmdvirgule.Click

    textcase1.Text += "."

End Sub

' bouton egal

Private Sub cmdegal_Click(sender As Object, e As EventArgs) Handles cm
degal.Click

    textcase1.Text = New DataTable().Compute(textcase1.Text, Nothing)

    If textcase1.Text = "∞" Then
        Try

            textcase1.Text = textcase1.Text / 0

        Catch ex As Exception

        Finally

            textcase1.Text = "Erreur"
        
```

```

        End Try

    End If

End Sub

' bouton parentheses

Private Sub Button11_Click(sender As Object, e As EventArgs) Handles Button11.Click
    textcase1.Text += ")"
End Sub

Private Sub Button12_Click(sender As Object, e As EventArgs) Handles Button12.Click
    textcase1.Text += "("
End Sub

End Class

```