

ISTA NTIC SYBA

Programmation Événementielle

TP N° 2

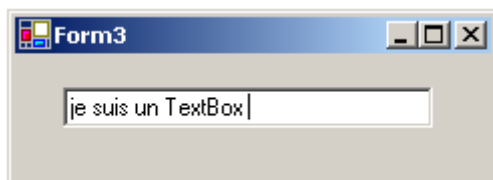
Objectifs :

- Apprendre à réaliser des interfaces simples avec MS Visual Studio
- Se familiariser avec les contrôles simples : TextBox, Label, Button...
- Apprendre à manipuler et à modifier les propriétés des contrôles (Graphiquement, Par code)
- Apprendre à utiliser InputBox et MsgBox

Rappel :

a. TextBox :

Le contrôle TextBox est certainement le contrôle le plus utilisé : il permet de saisir des chaînes de caractère d'une grande taille...



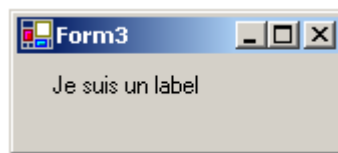
Propriété	Description
CanFocus	Détermine si le contrôle peut recevoir le focus
CharacterCasing	Détermine la casse du texte : majuscules (upper) ou minuscules (lower)
Focused	Indique si le contrôle détient le focus
ForeColor	Couleur du texte
HideSelection	Définit si le contrôle masque la sélection lorsqu'il perd le focus
Lines	Tableau correspondant aux lignes du contrôle
MaxLength	Nombre de caractères maxima du contrôle
Modified	Spécifie si le contenu du champ a été modifié depuis sa création
MultiLine	Définit si le contrôle est multi lignes
PasswordChar	Définit le caractère servant à masquer un mot de passe
ReadOnly	Contenu du champ en lecture seule
Scrollbars	Affiche ou masque les barres de défilement
Selectionlength	Longueur de la sélection
SelectionStart	Indice de début de la sélection dans le champ
Text	Contenu du champ
TextLength	Longueur du texte dans le contrôle
Name	Le nom attribué au contrôle et utilisé dans le code.
BackColor	Permet de donner une couleur au fond
BorderStyle	Permet d'enlever la bordure de la zone de texte.
CausesValidation	Indique si l'événement Validate du contrôle précédent sera déclenché lorsque notre TextBox sera activée.
Enabled	Permet de désactiver ou d'activer la zone de texte
Font	Détermine la police de caractères utilisée, sa taille, son style (gras, souligné ...).
ForeColor	Détermine la couleur d'écriture.
TabIndex	Position du contrôle dans l'ordre de tabulation du formulaire.
...	...

Méthode	Description
Clear	Efface le contenu du champ texte
Copy / Cut	Copie / coupe la sélection dans le presse papier
Focus	Donne le focus au contrôle
ResetText	Rétabli la valeur initiale du champs
...	...

Evénement	Description
TextChanged	Déclenché lorsque le texte change
...	...

b. Label

Le contrôle label est utilisé pour afficher du texte qui ne sera pas éditable par l'utilisateur. Il est généralement utilisé pour afficher le rôle des différents contrôles.



Propriété	Description
BorderStyle	Style de bordure
AutoSize	Le contrôle s'adapte à la taille du texte
Text	Contenu du label
...	...

c. Button

Le contrôle « button » est principalement utilisé pour déclencher une action lors du clic et a pratiquement les mêmes propriétés déjà vues

d. Utilisation des formulaires préétablis :

1) InputBox :

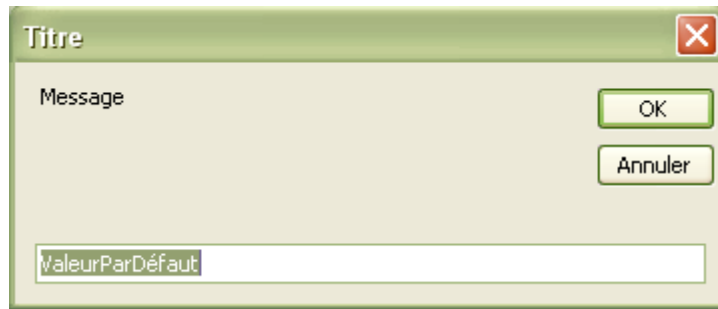
Ce formulaire permettra à l'utilisateur d'introduire une donnée qui sera récupérée dans une variable de type String, il peut être utilisé à travers la syntaxe suivante :

```
Dim s As String
s = InputBox("Message", "Titre", "ValeurParDéfaut", x, y)
```

Remarques :

Seul le 1^{er} paramètre est obligatoire
x et y représente les coordonnées du point d'affichage de la fenêtre sur l'écran
la valeur itroduite par l'utilisateur sera stockée dans la variable s et pour par la suite être utilisée...

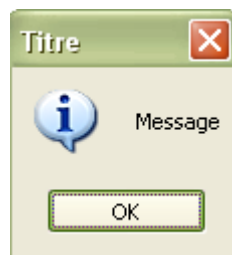
Ce qui donne l'affichage suivant :



2) MsgBox / MessageBox :

Ce formulaire permettra au programmeur d'afficher un message à l'utilisateur en vue de lui passer une information particulière, il peut être utilisé à travers une syntaxe simplifiée (MsgBox) ou détaillée (MessageBox) comme suit :

```
MsgBox("Message", MsgBoxStyle.Information, "Titre")
```



```
MessageBox.Show("Message", "Titre", MessageBoxButtons.YesNo, MessageBoxIcon.Exclamation)
```



Remarques :

Le programmeur peut faire un test sur la réponse de l'utilisateur (« Oui » ou « Non » par exemple), pour faire on utilise cette syntaxe pour les deux cas :

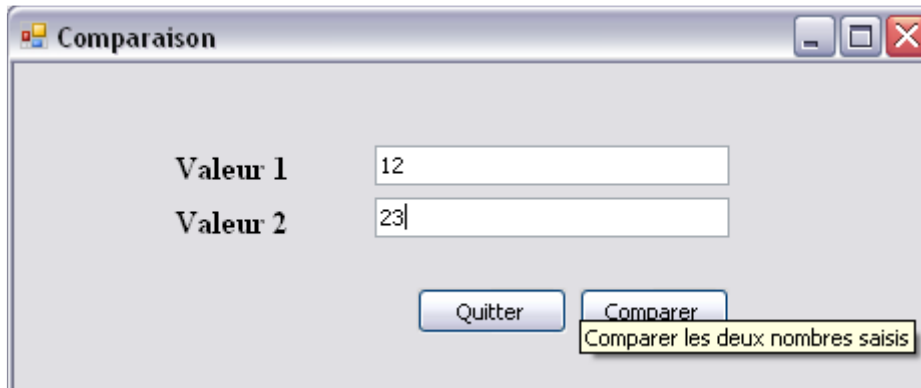
```
Dim Reponse = MsgBox(...) ou bien Dim Reponse As MsgBoxResult = MsgBox(...)
If Reponse = MsgBoxResult.Yes Then
    'Traitement 1
Else
    'Traitement 2
End If
```

Préliminaire :

Créer un projet Windows sous le nom "MonProjet2" et enregistrer le dans un dossier qui porte votre nom sur le bureau.

Exercice 1 :

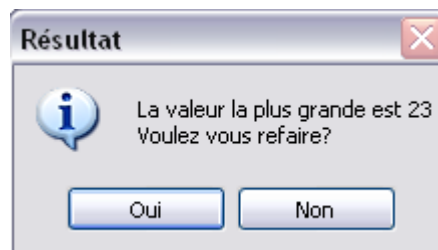
Construire une form conforme au modèle ci dessous permettant de comparer deux nombres introduits par l'utilisateur.



a. Intégrer les contrôles suivants :

- Il ne faut pas accepter les des caractères dans les zones de texte
- Le bouton Comparer ne devient actif qu'après la saisie des deux valeurs par des données exactes
- L'utilisateur n'a le droit de quitter un textbox que si la valeur qu'il a introduite est numérique (Utiliser l'événement LostFocus)

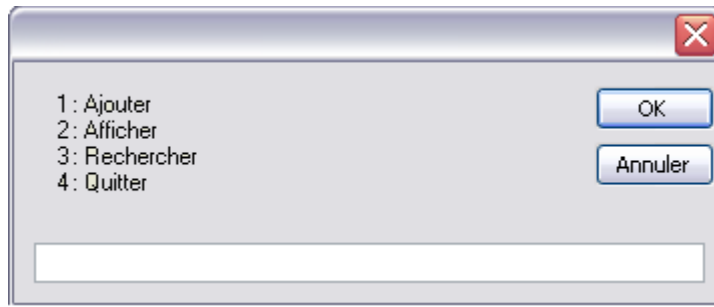
b. Le click sur le bouton comparer permet d'afficher un message personnalisé sous la forme suivante :



- Si l'utilisateur choisit "Oui" il peut recommencer la comparaison de deux autres valeurs
- Si l'utilisateur choisit "Non" tous les contrôles de la form seront désactivé sauf le bouton "Quitter" qui lui permettra de mettre fin à l'application

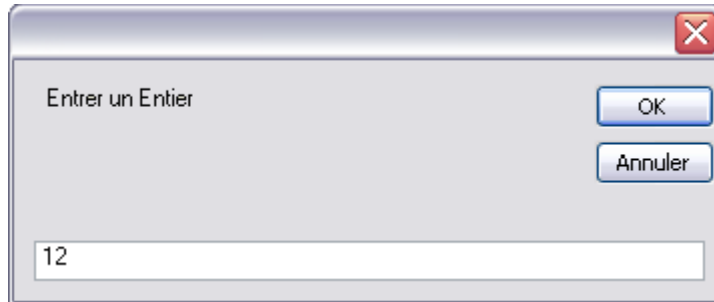
Exercice 2 :

L'idée de cet exercice est de pouvoir remplir un tableau d'entier à travers la fonction inputbox comme suit :



Selon le choix de l'utilisateur le programme permet de réaliser l'opération désirée :

- **Ajouter** : permet d'afficher un autre inputbox pour introduire un autre entier



- **Afficher** : permet d'afficher toutes les valeurs saisies ainsi que leur somme et leur moyenne dans un msgbox
- **Rechercher** : permet de chercher une valeur dans le tableau
- **Quitter** : permet de mettre fin à l'application.

Exercice 3 :

Construire une form conforme au modèle ci dessous permettent de réaliser les opérations de base sur deux valeurs introduites par l'utilisateur.



L'idée du projet est de saisir deux valeurs dans deux textbox et d'afficher le résultat du calcul suivant l'opération choisie.

Exercice 4 :

Construire un formulaire conforme au modèle ci dessous permettant de réaliser une calculatrice Windows simplifiée.

