Institut Supérieur d'Informatique et de Multimédia de Sfax Enseignant TP : CHTOUROU Mahdi

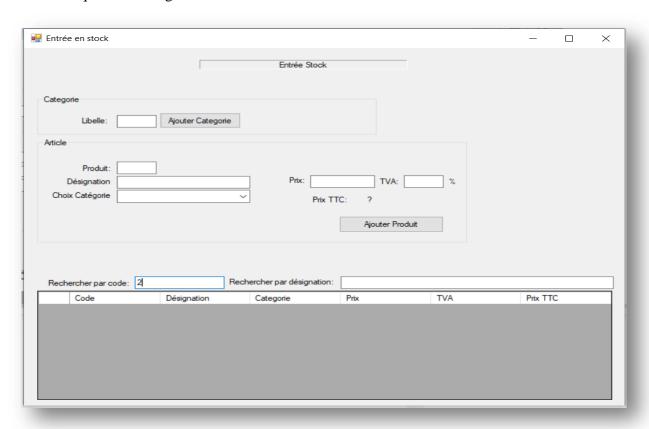
Auditoire: Deuxième année licence

2019-2020 Filière : D-TMW

Matière: Programmation évènementielle

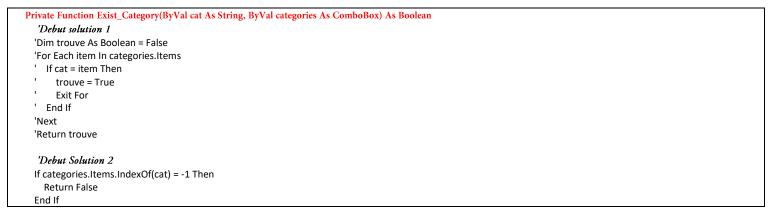
Examen TP

Nous voulons créer une application en VB.Net pour simuler l'entrée d'articles dans le stock, comme indiqué dans la figure ci-dessous.



L'interface contient une liste déroulante (ComboBox) dans laquelle différentes catégories sont stockées.

1- Déclarez une fonction appelée "Exist_Category" pour vérifier si une catégorie existe dans une liste déroulante



```
Return True
End Function
```

2- Implémenter le clic du bouton identifié par "btnAddCategory" qui permet d'ajouter l'étiquette d'une catégorie dans une liste de catégories (de type ComboBox)

```
Private Sub btnAjouterCategorie_Click(sender As Object, e As EventArgs) Handles btnAjouterCategorie.Click

If Not Exist_Category(libelleCategorie.Text, categories) Then
    categories.Items.Add(libelleCategorie.Text)
    txtCategorie.Clear()
End If
End Sub
```

3- Déclarez les tableaux de chaînes suivants: codesProduits, désignationsProduits, catégoriesProduits, TvaProduits

```
Public Class FicheProduit

Private codesProduits() As String = {}, designationsProduits() As String = {}, categoriesProduits() As String = {}

Private TvaProduits() As Double = {}

Private PrixProduits() As Double = {}

...

End Class
```

4- Sachant qu'un produit en stock est caractérisé par un code unique, une désignation, son prix et la TVA appliquée sur son prix et que le prix TTC = prix + (TVA*prix)/ 100, il vous est demandé de mettre en œuvre l'événement que pour chaque modification du prix ou de la valeur de TVA calculera à nouveau le montant total net à payer(PrixTTC). Une étiquette intitulée par "Prix TTC" permet d'afficher le prix TTC obtenu. Traitez les exceptions en coloriant le champ en rouge pour informer l'utilisateur qu'il a saisit des données invalides dans les champs non numérique.

```
Private Sub prix_TextChanged(sender As Object, e As EventArgs) Handles prix.TextChanged
  'supprimer les espaces vides avant la comparaison
    txtPrix.Text = Trim(txtPrix.Text)
    If txtPrix.Text = "" Then
      txtPrix.BackColor = Color.White
      prixTTC.Text = "?"
    Else
      Dim prix As Double = 0
      Dim tva As Double = 0
        prix = CDbl(txtPrix.Text)
        Try
          tva = CDbl(txtTva.Text)
          prixTTC.Text = prix + prix * tva / 100
        Catch ex As Exception
          prixTTC.Text = "?"
        End Try
        txtPrix.BackColor = Color.White
      Catch ex As Exception
        txtPrix.BackColor = Color.Red
        prixTTC.Text = "?"
      End Try
    End If
End Sub
```

```
Private Sub tva_TextChanged(sender As Object, e As EventArgs) Handles tva.TextChanged
'supprimer les espaces vides avant la comparaison
    txtTva.Text = Trim(txtTva.Text)
    If txtTva.Text = "" Then
      txtTva.BackColor = Color.White
      prixTTC.Text = "?"
    Else
      Dim prix As Double = 0
      Dim tva As Double = 0
        tva = CDbl(txtTva.Text)
          prix = CDbl(txtPrix.Text)
          prixTTC.Text = prix + prix * tva / 100
        Catch ex As InvalidCastException
          prixTTC.Text = "?"
        End Try
        txtTva.BackColor = Color.White
      Catch ex As InvalidCastException
        txtTva.BackColor = Color.Red
        prixTTC.Text = "?"
      End Try
    Fnd If
End Sub
```

5- Sachant que pour créer un nouveau produit, il faut récupérer son code et l'ajouter dans un tableau de codes, ainsi que la désignation dans un tableau de désignations, et ainsi de suite pour tous les autres champs, déclarer les tableaux suivants: codesProduits, designationsProduits, categoriesProduits, prixProduits, tvaProduits.

```
Private codesProduits() As String = {}, designationsProduits() As String = {}, categoriesProduits() As String = {}
Private TvaProduits() As Double = {}
Private PrixProduits() As Double = {}
```

- 6- Ecrivez une fonction appelée "AddProduct" qui vous permet de récupérer tous les champs du produit et d'enregistrer chaque valeur dans son propre tableau.
 - NB: Pour ajouter dans un tableau, il faut le redimensionner avec une nouvelle case tout en gardant son contenu.
- 7- Vider les champs après chaque enregistrement
- 8- sachant que pour accepter d'être ajouté, il faut que toutes les conditions citées cidessous soient bien remplies
 - Code produit obligatoire non vide (chaîne vide non autorisée)
 - Nom du produit obligatoire non vide (chaîne vide non autorisée)
 - Catégorie sélectionnée
 - Prix numérique valide et supérieur à zéro
 - Taux de TVA> 0

Améliorer la fonction "AddProduct" avec ces conditions

9- Implémenter le clic du bouton identifié par "btnAddProduct" pour ajouter un produit dans un DataGridView

6-7-8-9 (Réponse)

```
Private Sub btnAjouterProduit_Click(sender As Object, e As EventArgs) Handles btnAjouterProduit.Click
    Dim code As String = Trim(txtCode.Text)
   Dim designation As String = Trim(txtDesignation.Text)
    Dim categorie As String = Trim(categories.SelectedItem)
    Dim valid As Boolean = code <> "" And designation <> "" And categorie <> "" And prixTTC.Text <> "?" And CDbl(prixTTC.Text) > 0
    If valid Then
      ReDim Preserve codesProduits(codesProduits.Count)
      ReDim Preserve designationsProduits(designationsProduits.Count)
      ReDim Preserve categoriesProduits(categoriesProduits.Count)
      ReDim Preserve PrixProduits(TvaProduits.Count)
      ReDim Preserve TyaProduits(TyaProduits Count)
      codesProduits(codesProduits.Count - 1) = code
      designationsProduits(designationsProduits.Count - 1) = designation
      categoriesProduits(categoriesProduits.Count - 1) = categorie
      PrixProduits(PrixProduits.Count - 1) = CDbl(txtPrix.Text)
      TvaProduits(TvaProduits.Count - 1) = CDbl(txtTva.Text)
      dgvProduits.Rows.Add(code, designation, categorie, txtPrix.Text, txtTva.Text, prixTTC.Text)
      '5-)vider vos champs
      txtCode.Clear()
      txtDesignation.Clear()
      txtPrix.Clear()
      txtTva.Clear()
      prixTTC.Text = "?"
    End If
  End Sub
```

10-L'utilisateur peut rechercher un produit par son code ou sa désignation. Implémenter l'événement qui affiche, dans la liste (DataGridView), uniquement les produits contenant des valeurs entrées

```
Private Sub codeSearch_TextChanged(sender As Object, e As EventArgs) Handles codeSearch.TextChanged
    dgvProduits.Rows.Clear()
    For i = 0 To codesProduits.Count - 1
      If\ UCase (codes Produits (i)). Starts With (UCase (codeSearch. Text))\ Or\ UCase (codes Produits (i)). Ends With (UCase (codeSearch. Text))\ Then
         dgvProduits.Rows.Add(codesProduits(i), designationsProduits(i), categoriesProduits(i), PrixProduits(i), TvaProduits(i),
                    PrixProduits(i) * (1 + TvaProduits(i)) / 100)
      End If
    Next
  Private Sub designationSearch_TextChanged(sender As Object, e As EventArgs) Handles designationSearch.TextChanged
    dgvProduits.Rows.Clear()
    For i = 0 To designationsProduits.Count - 1
      If UCase(designationsProduits(i)).StartsWith(UCase(designationSearch.Text)) Or UCase(designationsProduits(i)).EndsWith(UCase(designationSearch.Text)) Then
         dgvProduits.Rows.Add(codesProduits(i), designationsProduits(i), categoriesProduits(i), PrixProduits(i), TvaProduits(i),
                    PrixProduits(i) * (1 + TvaProduits(i)) / 100)
      End If
    Next
  End Sut
```

Annexe:

Pour déclarer un tableau

Private tableauName () As String

Pour Traiter une exception

Try

//code qui risque de lancer une erreur à l'exécution

Catch ex As InvalidCastException

//traiter l'erreur

End Try

Pour redimensionner un tableau:

Dim Preserve t(index_maximal)