Utilitaires méconnus de StrataFrame

Voici quelques classes d'utilitaires, parmi ceux qui se trouvent dans le NameSpace MicroFour.StrataFrame.Tools. Cette liste n'est pas exhaustive, et les regroupements dans lesquels je les présente ne sont qu'un choix personnel.

Pour utiliser ces utilitaires, pensez à référencer la DLL `MicroFour StrataFrame UI Expanded` dans votre projet.

Couche visuelle

Faire un LockScreen d'un Form

Pour redessiner la couche visuelle des forms uniquement quand c'est nécessaire (par exemple pour éviter un rafraichissement à chaque contrôle ajouté ou modifié). Cette fonction booléenne peut recevoir en paramètre le nom du form à locker, ou bien le handle de la fenêtre.

Cette fonction est membre de la classe LockScreen

Exemple :

```
LockScreen.Lock(Me)
```

Le déverrouillage sera obtenu par Unlock

Faire une capture d'écran

Pour faire une copie d'écran dans une image en mémoire ou dans un fichier sur le disque. La zone capturée peut être l'écran entier, ou une fenêtre spécifiée par son handle. Les différents types de format d'image sont acceptés.

Cette fonction est membre de la classe ScreenCapture

Exemple :

```
Dim Ecran As String = "d:\developpement\test.bmp"
Dim CeForm As String = "d:\developpement\test2.bmp"
CaptureScreenToFile(Ecran, System.Drawing.Imaging.ImageFormat.Bmp)
CaptureWindowToFile(Me.Handle,CeForm,System.Drawing.Imaging.ImageFormat.bmp)
```

Fonctions sur les nombres et les dates

Le between de Fox!

Pour déterminer si une valeur est comprise entre deux bornes. La valeur à tester et les bornes peuvent être de type Int32, Decimal, ou Double pour les numériques, ou de type DateTime pour les dates. On peut choisir si les bornes sont incluses ou non (ce paramètre est optionnel)

Ces fonctions booléennes sont membres de la classe Helpers.

Exemple :

```
Dim x As Integer, y As Integer, z As Integer
if between(x,y,z,true) then
    MsgBox("x est compris entre y et z, sans être égal à y ou à z")
End If
```

Calcul de l'âge

Pour calculer l'âge d'une personne à la date d'aujourd'hui, ou son âge à une date donnée. Renvoie le résultat sous forme de chaine de caractère.

Ces fonctions Chaines sont membres de la classe Common.

Exemple :

```
Dim x As DateTime = #01/01/1965#
MsgBox("né le " &x.ToLongDateString & ", il a aujourdhui " & GetAge(x) & " ans")
MsgBox("le 31 décembre 2020, il aura " & GetAge(x,#12/31/2020#) & " ans")
MsgBox("et le lendemain, il fêtera ses " & GetAge(x,#01/01/2021#) & " ans")
```

Traitement des chaines de caractères

Centrer une chaîne en ajoutant des espaces ou un caractère spécifique à droite et à gauche

Pour obtenir ce combiné de PadL et de PadR, passez une chaine de caractère, la longueur désirée, et le caractère de remplissage.

Ces fonctions Chaines sont membres de la classe Common.

Exemple :

```
MsgBox(PadCenter("Bonjour",20,CChar("*")))
```

Séparer par des espaces une chaîne en casse mixte

Les Énumérations sont très pratiques, mais leur défaut est de ne pas supporter les espaces dans les chaînes de description des items. Cette fonction String découpe une chaine de caractères à chaque lettre en Capitale en insérant un espace. Les 2 surcharges permettent de reconnaître les chiffres comme caractère discriminant, et de spécifier une limite au nombre de mots à séparer.

Ces fonctions Chaines sont membres de la classe Common.

Exemple :

```
MsgBox(SplitAtCaps("BonjourCeciEstUneChaineVenantD'uneEnumération").ToLower)
MsgBox(SplitAtCaps("Et2MotsSéparésPar1Chiffre",True,0).ToLower)
```

Obtenir la chaîne de caractère d'un item d'une énumération

Pour obtenir la chaîne de caractère de description d'un item d'une énumération.

Cette fonction Chaine est membre de la classe Common.

Exemple :

```
MsgBox(GetEnumDisplayValue(CType(4, Border3DStyle)))
```

Tables et données

Construire et peupler une DataTable à partir d'une Enum

Cet ensemble de fonctions Table permet de construire et peupler une DataTable depuis une Enum (un peu comme un SELECT … INTO ARRAY … en Fox). La table résultat comporte 2 colonnes (Description et Value). On peut en option spécifier l'ordre croissant ou décroissant dans lequel seront traités les éléments de l'Enum.

Ces fonctions Tables sont membres de la classe Common.

Exemple :

```
Dim uneTable As Data.DataTable
uneTable = BuildDataTableFromEnum(gettype(CreditCardType))
```

Exporter le contenu d'un BO vers un document Excel (un COPY to … XLS)

Cette méthode exporte le contenu d'un Business Object dans un fichier Excel. Une ligne d'en-tête est créée, comportant le nom des colonnes.

Cette fonction est membre de la classe ExcelHelper

Exemple :

```
ExportToExcel(me.BO_pays1 , "d:\developpement\test.xls")
```

Attention ! aucune ligne ne doit comporter de NULL dans aucune colonne !

Importer une feuille Excel dans une DataTable (un APPEND FROM … TYPE XLS)

Pour importer dans une DataTable le contenu d'une feuille Excel spécifique dans un classeur ; on peut préciser si les données comportent une ligne d'en-tête comportant le nom des colonnes à créer (en l'absence d'en-tête ; les colonnes créées seront nommées F1, F2,…).

Cette fonction est membre de la classe ExcelHelper

Exemple :

```
Dim uneTable As Data.DataTable
uneTable = ImportExcel("d:\developpement\test.xls",True,1)
```

Fichiers

Obtenir les informations et l'icône d'une extension

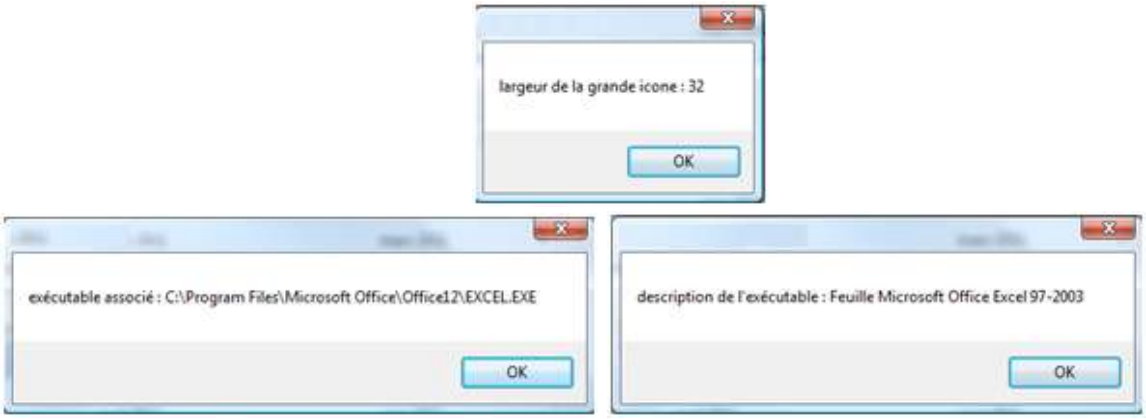
En lisant la base de registre, cette fonction renvoie un objet de type FileTypeInformation (défini dans le NameSpace Microfour.StrataFrame.Win32), qui présente les propriétés ShellExecutable et TypeDescription (pour l'exécutable lié à cette icône sur la machine), et FileIconLarge et FileIconSmall (qui décrivent les dimensions des icônes de cette extension)

Cette fonction est membre de la classe Common

Exemple :

```
Dim ext As String = "xls"
MsgBox("largeur de la grande icone : " & _
       GetFileExtensionDetails(ext).FileIconLarge.Width.ToString)
MsgBox("exécutable associé : " & _
       GetFileExtensionDetails(ext).ShellExecutable)
MsgBox("description de l'exécutable : " & _
               GetFileExtensionDetails(ext).TypeDescription)
```

affichera :



Afficher la taille d'un fichier sous une forme facilement lisible

Cette fonction String retourne un numérique long sous forme d'une chaine, avec l'unité adéquate à la taille (byte, KB, MB).

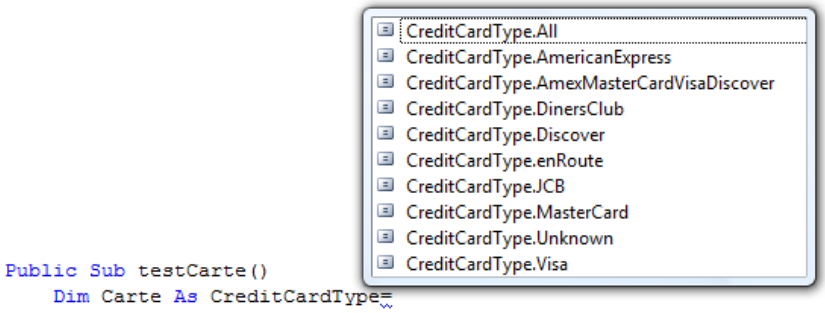
Cette fonction est membre de la classe Common

Exemple :

```
Dim Taille As Long = 2347613
MsgBox(GetFileSizeStringRepresentation(Taille,2))
```

Cartes Bancaires

Les différents types de cartes bancaires sont disponibles dans une énumération fournie :



cette énumération **CreditCardType** est publique dans le NameSpace Tools.

Vérifier la composition du numéro

Pour s'assurer que le numéro est correctement formé, cette fonction booléenne refait le calcul de la clé de Luhn. Le numéro à vérifier est passé sous forme de chaine.

Cette fonction est membre de la classe Common

Exemple :

```
If ValidateCreditCardNumber("1234567890123456")
    MsgBox("n° correct")
Else
    MsgBox("n° de carte bancaire incorrect")
End If
```

Attention, dans cette fonction, le numéro passé en paramètre sous forme de chaine ne doit pas comporter d'espaces!

Obtenir le type de la carte par son numéro

À partir du numéro de la carte, cette fonction Enum renvoie le type de carte parmi l'énumération. Le numéro à vérifier est passé sous forme de chaine.

Cette fonction est membre de la classe Common

Exemple :

```
Dim Numero As String = "1234 5678 9012 3456"
MsgBox(GetEnumDisplayValue(GetCreditCardType(Numero)))
```

Vérifier que le numéro correspond au type de la carte

Cette fonction booléenne permet de vérifier que le numéro correspond au type de la carte. Le numéro est passé sous forme de chaine, le type de la carte est choisi dans l'énumération fournie.

Cette fonction est membre de la classe Common

Exemple :

```
Dim Carte As CreditCardType=CreditCardType.Visa
Dim Numero As String = "1234567890123456"
If IsValidCreditCardType(Carte,Numero)
    MsgBox("Le numéro et le type de carte correspondent")
Else
    MsgBox("Ce numéro n'est pas celui d'une carte " & GetEnumDisplayValue(Carte))
End If
```

Attention, dans cette fonction, le numéro passé en paramètre sous forme de chaine ne doit pas comporter d'espaces!