# **Projet KARATE**

Le projet karaté va permettre à la ligue De Lorraine de Karaté d'organiser tout au long de l'année des compétitions de «Kata» dans les différents clubs qui lui sont affiliés et de numériser l'ensemble des informations pour une gestion plus simple et rapide. Cette application sera compatible avec Windows et reliée à une base de données.



# Présentation de l'équipe et de la mission



#### T2.2 Saisie de l'inscription d'un membre à une compétition:

- → Permettre la création
- → La modification et la suppression d'une inscription d'un membre à une compétition à partir d'un formulaire

### ► T2.5 Affichage des résultats finals de la compétition :

- → Affichage des résultats d'une compétition donnée
- → Affichage de la liste des jurys d'une compétition
- → Mise à jour de la note globale obtenue par chaque compétiteur

# La documentation technique

La documentation technique sera répartie en trois sous parties démontrant le processus de réalisation afin de répondre aux 2 objectifs ci-dessus.

#### **Prérequis**

Nous allons présenter les prérequis nécessaires à la création de notre application en différentes sous parties :

- $\rightarrow$  VB.NET
- $\rightarrow$  Git

- → Module de connexion a la BDD
- → Notre BDD Access

#### **VB.NET**

Visual Basic .NET est un langage de programmation à la syntaxe similaire à celle de Visual Basic 6.

VB.NET permet de développer en .NET via Visual Studio, sur les systèmes d'exploitation Windows. Comme tout autre langage de la plateforme .NET, VB.NET est compilé vers le langage intermédiaire CIL.

L'installation d'un logiciel est nécessaire pour développer. Nous utiliserons Visual Studio.



#### **GIT**

Git est un système de contrôle de version distribué gratuit et open source conçu pour gérer tout, des petits projets aux très grands projets avec rapidité et efficacité.

#### Nous gérons notre dépôt avec différents types de commande

```
git add --all
git commit -m "votre message"
git push origin master

#Vous aurez besoin de mettre à jour votre dépôt
git pull origin master
```



#### **MODULE DE CONNEXION A LA BDD**

Nous avons créé un objet « Connexion » pour avoir accès à la base de données. Nous indiquons dans les propriétés « ConnectionString » les paramètres de la base de données.

Nous avons ensuite créé le module de connexion :

```
Imports System.Data.OleDb
Module ModuleConnexion
    Public maConnexion As New OleDbConnection()
    Public Sub connexion(ByVal nombd As String)
            maConnexion.ConnectionString =
"Provider=Microsoft.ACE.OLEDB.12.0; data source = " &
                                Application.StartupPath & "\" & nombd
            maConnexion.Open()
        Catch ex As Exception
            MessageBox.Show("Impossible d'établir la connexion à la base." &
ex.Message)
        End Try
    End Sub
    Public Sub deconnexion()
        Try
            maConnexion.Close()
        Catch ex As Exception
            MessageBox.Show("Impossible d'établir la connexion à la base." &
ex.Message)
        End Try
    End Sub
End Module
```

Nous utilisons l'instruction *try catch* qui exécute notre programme et définit une réponse si l'une de ces instructions provoque une exception.

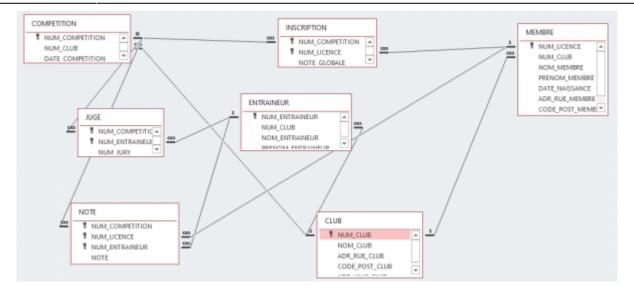


#### **BDD ACCESS**

Nous avons également besoin d'une base de données afin de répertorier les membres, les compétitions etc...



Nous avons donc répertorié 7 tables.



Notre base de données est stockée en local sous Access.



### **Objectif T2.2**

### Rappel de l'objectif :

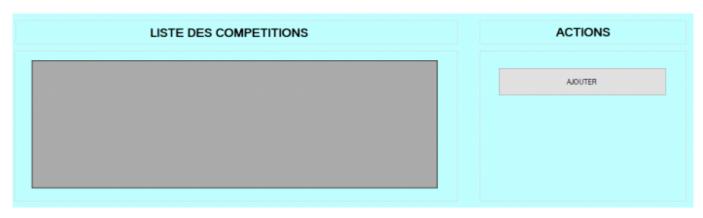
- → Permettre la création
- → La modification et la suppression d'une inscription d'un membre à une compétition à partir d'un formulaire

#### PERMETTRE LA CRÉATION D'UNE INSCRIPTION D'UN MEMBRE À UNE COMPÉTITION

Pour répondre à ce besoin nous avons décidé de répertorier l'ensemble des compétitions pour ensuite permettre le choix d'ajouter un membre.

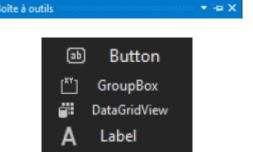
Visuellement nous avons mis en place un DataGridView, ainsi qu'un bouton redirigeant sur la liste des membres.

# La page actuelle : Competitions.vb[Design]



La réalisation de cette page graphique a été guidée par la "Boîte à outils"

Et ses différents éléments :



Un simple glissé déposé permet l'insertion de ces éléments.

Passons dès maintenant au développement.

# La page actuelle: Competitions.vb

```
Imports System.Data.OleDb
```

Il est nécessaire d'ajouter cette ligne étant donné que System. Data. Ole Db est le fournisseur de données .NET Framework pour OLE DB.

```
Private Sub Competitions_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        connexion("bdKarate.accdb")
        Me.LISTE COMPETITION.Rows.Clear()
        Me.LISTE COMPETITION.ColumnCount = 3
        Me.LISTE COMPETITION.Columns(0).Width = 200
        Me.LISTE COMPETITION.Columns(1).Width = 233
        Me.LISTE COMPETITION.Columns(2).Width = 200
        Me.LISTE COMPETITION.Columns(0).Name = "NUM COMPETITION"
        Me.LISTE COMPETITION.Columns(1).Name = "NOM CLUB"
        Me.LISTE COMPETITION.Columns(2).Name = "DATE COMPETITION"
    '** Evite d'avoir une ligne blanche en fin, destinée à la
    '** saisie d'une nouvelle ligne
    Me.LISTE COMPETITION.AllowUserToAddRows = False
        Dim req As String
        req = "select * from COMPETITION INNER JOIN CLUB ON
COMPETITION.NUM CLUB=CLUB.NUM CLUB ORDER BY NUM COMPETITION"
        'On déclare une variable de type objet "Command" qui exécute une
commande (requête SQL) sur la base de données
        Dim cmdRech As New OleDbCommand(reg, maConnexion)
        'On déclare une variable de type objet "DataReader" qui récupère les
enregistrements de la requête de type SELECT
        Dim rdrRech As OleDbDataReader = cmdRech.ExecuteReader()
```

```
Dim numligne As Integer
While rdrRech.Read()

Me.LISTE_COMPETITION.Rows.Add()
    numligne = Me.LISTE_COMPETITION.RowCount - 1
    Me.LISTE_COMPETITION.Item(0, numligne).Value =
rdrRech.Item("NUM_COMPETITION")
    Me.LISTE_COMPETITION.Item(1, numligne).Value =
rdrRech.Item("NOM_CLUB")
    Me.LISTE_COMPETITION.Item(2, numligne).Value =
rdrRech.Item("DATE_COMPETITION")

End While

End Sub
```

## Explications:

```
Private Sub Competitions_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
```

Lors de l'évènement "Load", soit le chargement du Forms, nous allons définir les colonnes de notre DataGridView

```
Me.LISTE_COMPETITION.ColumnCount = 3
Me.LISTE_COMPETITION.Columns(0).Width = 200
Me.LISTE_COMPETITION.Columns(1).Width = 233
Me.LISTE_COMPETITION.Columns(2).Width = 200
Me.LISTE_COMPETITION.Columns(0).Name = "NUM_COMPETITION"
Me.LISTE_COMPETITION.Columns(1).Name = "NOM_CLUB"
Me.LISTE_COMPETITION.Columns(2).Name = "DATE_COMPETITION"
```

Ensuite, nous devons permettre l'affichage des informations de la base de données à l'intérieur de ce dernier.

```
Dim req As String
    req = "select * from COMPETITION INNER JOIN CLUB ON
COMPETITION.NUM_CLUB=CLUB.NUM_CLUB ORDER BY NUM_COMPETITION"
```

Nous déclarons une variable 'reg' qui est un String. Nous écrivons ensuite notre requête.

```
Dim cmdRech As New OleDbCommand(req, maConnexion)
```

Puis nous déclarons une variable de type objet "Command" qui exécute une commande (requête SQL) sur la base de données

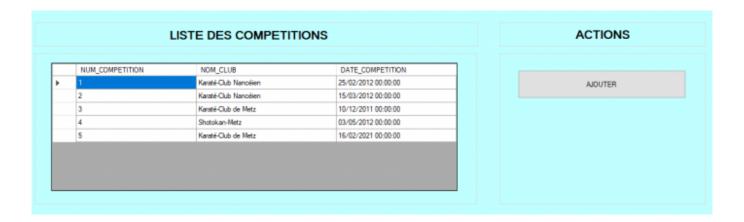
```
Dim rdrRech As OleDbDataReader = cmdRech.ExecuteReader()
```

Nous déclarons une variable de type objet "DataReader" qui récupère les enregistrements de la requête de type SELECT

Ensuite nous déclarons une nouvelle variable de type entier : numligne Nous créons une boucle, qui recommence tant qu'il y a encore des enregistrements. S'il y a encore des enregistrements nous les affichons comme ceci :

rdrRech.Item("LE NOM DE LA COLONNE DANS LA BDD")





## Maintenant gérons le bouton :

```
'On assigne la valeur NUM_LICENCE à une variable
NumCompetitionSelectionne = Me.LISTE_COMPETITION.Item(0,
LigneSectionne).Value

FormInscriptionMembreCompetition.Show()

End Sub
```

# Explications:

Private Sub Button3\_Click(sender As Object, e As EventArgs) Handles button\_ajouter.Click

Lors de l'évènement "Click", soit quand l'utilisateur clique sur le bouton.

```
Dim LigneSectionne As Integer
   LigneSectionne = (Me.LISTE_COMPETITION.CurrentRow.Index)
```

On déclare une variable qui récupère la ligne sélectionnée sur le DataGridView

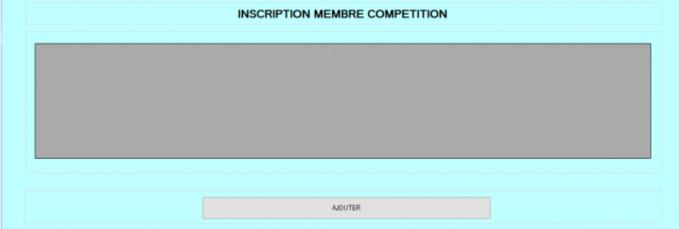
```
NumCompetitionSelectionne = Me.LISTE_COMPETITION.Item(0,
LigneSectionne).Value
```

On récupère le premier élément de la colonne du DataGridView (NUM\_COMPETITION) et on le stock dans une variable.

FormInscriptionMembreCompetition.Show()

Puis nous affichons la page d'inscription.





Nous ne détaillerons pas les éléments utilisés car il s'agit des mêmes que le formulaire précédent. Cette page va récupérer le choix de la compétition fait sur la page *Competitions.vb*, puis va afficher l'ensemble des membres de la table MEMBRE.

Passons directement au développement :

## </>>

#### La page actuelle : FormInscriptionMembreCompetition.vb

```
Private Sub FormInscriptionMembreCompetition Load(sender As Object, e As
EventArgs) Handles MyBase.Load
       Me.LISTE MEMBRE.Rows.Clear()
       Me.LISTE MEMBRE.ColumnCount = 8
       Me.LISTE MEMBRE.Columns(0).Width = 130
       Me.LISTE MEMBRE.Columns(1).Width = 130
       Me.LISTE MEMBRE.Columns(2).Width = 134
        Me.LISTE MEMBRE.Columns(3).Width = 130
       Me.LISTE MEMBRE.Columns(4).Width = 130
        Me.LISTE MEMBRE.Columns(5).Width = 134
        Me.LISTE MEMBRE.Columns(6).Width = 130
       Me.LISTE MEMBRE.Columns(7).Width = 130
        Me.LISTE MEMBRE.Columns(0).Name = "NUM LICENCE"
        Me.LISTE MEMBRE.Columns(1).Name = "NOM CLUB"
       Me.LISTE MEMBRE.Columns(2).Name = "NOM MEMBRE"
       Me.LISTE MEMBRE.Columns(3).Name = "PRENOM MEMBRE"
       Me.LISTE MEMBRE.Columns(4).Name = "DATE NAISSANCE"
       Me.LISTE MEMBRE.Columns(5).Name = "ADR RUE MEMBRE"
       Me.LISTE MEMBRE.Columns(6).Name = "CODE POST MEMBRE"
        Me.LISTE MEMBRE.Columns(7).Name = "ADR VILLE MEMBRE"
        '** Evite d'avoir une ligne blanche en fin, destinée à la
        '** saisie d'une nouvelle ligne
        Me.LISTE MEMBRE.AllowUserToAddRows = False
        Dim reg As String
        req = "select * from MEMBRE INNER JOIN CLUB ON
MEMBRE.NUM CLUB=CLUB.NUM CLUB"
        'On déclare une variable de type objet "Command" qui exécute une
commande (requête SQL) sur la base de données
       Dim cmdRech As New OleDbCommand(reg, maConnexion)
        'On déclare une variable de type objet "DataReader" qui récupère les
enregistrements de la reguête de type SELECT
        Dim rdrRech As OleDbDataReader = cmdRech.ExecuteReader()
       Dim numligne As Integer
       While rdrRech.Read()
            Me.LISTE MEMBRE.Rows.Add()
            numligne = Me.LISTE_MEMBRE.RowCount - 1
            Me.LISTE MEMBRE.Item(0, numligne).Value =
rdrRech.Item("NUM LICENCE")
            Me.LISTE MEMBRE.Item(1, numligne).Value =
rdrRech.Item("NOM CLUB")
            Me.LISTE_MEMBRE.Item(2, numligne).Value =
rdrRech.Item("NOM MEMBRE")
            Me.LISTE MEMBRE.Item(3, numligne).Value =
```

Nous ne détaillerons pas l'affichage des données étant donné qu'il s'agit de la même chose que précédemment hormis la requête SQL

# Explications :

Private Sub FormInscriptionMembreCompetition\_Load(sender As Object, e As
EventArgs) Handles MyBase.Load

Lors de l'évènement "Load", soit le chargement du Forms, nous allons définir les colonnes de notre DataGridView

SELECT\* FROM MEMBRE INNER JOIN CLUB ON MEMBRE.NUM CLUB=CLUB.NUM CLUB

Cette requête est de type **SELECT** permettant d'afficher les membres et **NOM\_CLUB** d'où l'**INNER JOIN**.

Nous allons ensuite récupérer le **NUM\_COMPETITION** sélectionné précédemment, ainsi que le NUM\_LICENCE de même sélectionner sur cette même page. Nous établirons une requête permettant son insertion dans la BBD.

# Deu d'essai :



# Maintenant gérons le bouton :

```
Private Sub Button1 Click(sender As Object, e As EventArgs) Handles
Button1.Click
        'On déclare une variable qui récupère la ligne sélectionnée sur le
DataGrid
        Dim LigneSectionne As Integer
        LigneSectionne = (Me.LISTE MEMBRE.CurrentRow.Index)
        'On assigne la valeur NUM LICENCE à une variable
        Dim NumLicenceSelectionne As String
        NumLicenceSelectionne = Me.LISTE MEMBRE.Item(0,
LigneSectionne).Value
        Dim CompetSelect As String
        CompetSelect = Competitions.NumCompetitionSelectionne
        'Try Catch test doublon
        Try
            Dim rega As String
            'On écrit une requête de type INSERT
            rega = "insert into INSCRIPTION values (@NUM COMPETITION,
@NUM LICENCE, @NOTE GLOBALE)"
            'On déclare une variable de type objet "Command" qui exécute une
commande (requête SQL) sur la base de données
            Dim cmdInsert As New OleDbCommand(rega, maConnexion)
            'Passage des paramètres
            cmdInsert.Parameters.AddWithValue("@NUM_COMPETITION", teste)
            cmdInsert.Parameters.AddWithValue("@NUM LICENCE",
NumLicenceSelectionne)
            cmdInsert.Parameters.AddWithValue("@NOTE GLOBALE", DBNull.Value)
            'On exécute la requête dans la base de données
            cmdInsert.ExecuteNonQuery()
            MsgBox("Le membre a bien été ajoute à la compétition...")
        Catch ex As Exception
            MessageBox.Show("Le membre est déjà inscrit à cet compétition")
        End Try
        Me.Close()
   End Sub
```

# Explications:

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles

#### Button1.Click

S'exécute lors de l'évènement "Click" sur le bouton.

```
Dim LigneSectionne As Integer
LigneSectionne = (Me.LISTE_MEMBRE.CurrentRow.Index)
```

On déclare une variable qui récupère la ligne sélectionnée sur le DataGridView

```
Dim NumLicenceSelectionne As String
NumLicenceSelectionne = Me.LISTE_MEMBRE.Item(0, LigneSectionne).Value
Dim CompetSelect As String
CompetSelect = Competitions.NumCompetitionSelectionne
```

On récupère le premier élément du DataGridView et on le stock dans une variable, NumLicenceSelectionne.

Puis nous récupérons *NumCompetitionSelectionne* de la page **Competitions.vb** que nous stockons également.

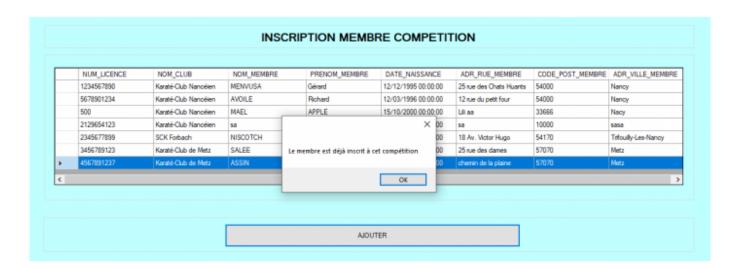
Ensuite, nous écrivons une requête :

```
reqa = "insert into INSCRIPTION values (@NUM_COMPETITION, @NUM_LICENCE,
@NOTE_GLOBALE)"
```

De type INSERT, nous insérons le **NUM\_LICENCE**, ainsi que le **NUM\_COMPETITION** dans la table **INSCRIPTION**.

Comme précédemment nous utilisons un Try Catch afin de vérifier qu'il n'y aura pas de doublon dans la table **INSCRIPTION**.

# Deu d'essai finaux:

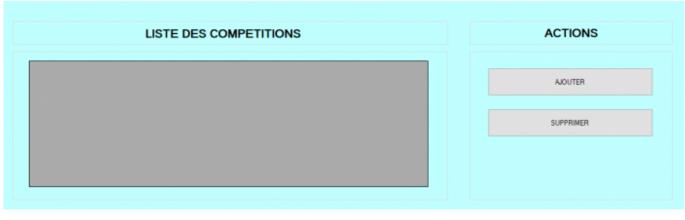




#### PERMETTRE LA SUPPRESSION D'UNE INSCRIPTION D'UN MEMBRE À UNE COMPÉTITION

Pour répondre à ce besoin, nous avons décidé d'utiliser la page *Competitions.vb* qui répertorie l'ensemble des compétitions pour ensuite permettre le choix d'ajouter et maintenant de supprimer une inscription.

Nous avons donc rajouter un bouton **SUPPRIMER** sur la page *Competitions.vb* :



Private Sub button\_supprimer\_Click(sender As Object, e As EventArgs) Handles button\_supprimer.Click

'On déclare une variable qui recupère la ligne selectionnée sur le DataGrid

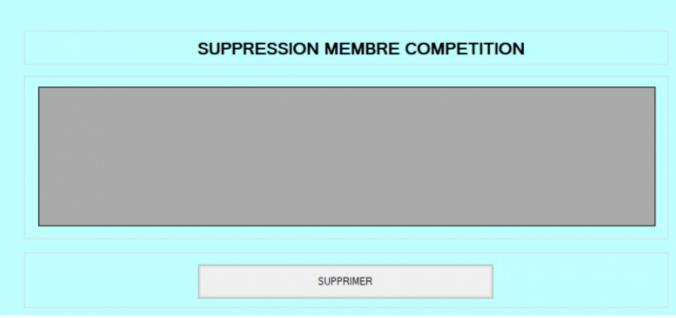
Dim LigneSectionne As Integer
LigneSectionne = (Me.LISTE\_COMPETITION.CurrentRow.Index)

'On assigne la valeurs NUM\_LICENCE à une variable
NumCompetitionSelectionne = Me.LISTE\_COMPETITION.Item(0, LigneSectionne).Value

FormSupressionMembreCompetition.Show()
End Sub

Nous ajoutons un nouveau bout de code lors de l'évènement "Click" pour ce nouveau bouton permettant de récupérer le choix de la compétition : NUM\_COMPETITION, puis on affiche FormSupressionMembreCompetition.vb

#### La page actuelle: FormSupressionMembreCompetition.vb[Design]



Nous ne détaillerons pas les éléments utilisés car il s'agit des mêmes que le formulaire précédent. Cette page va récupérer le choix de la compétition fait sur la page *Competitions.vb*, puis va afficher l'ensemble des membres de la table **INSCRIPTION** dont le **NUM\_COMPETITION** correspond au choi sur la page.

### </>>

#### La page actuelle : FormSupressionMembreCompetition.vb

```
Private Sub FormSupressionMembreCompetition Load(sender As Object, e As
EventArgs) Handles MyBase.Load
       Me.LISTE MEMBRE.Rows.Clear()
       Me.LISTE MEMBRE.ColumnCount = 8
        Me.LISTE MEMBRE.Columns(0).Width = 130
        Me.LISTE MEMBRE.Columns(1).Width = 130
       Me.LISTE MEMBRE.Columns(2).Width = 134
       Me.LISTE MEMBRE.Columns(3).Width = 130
        Me.LISTE MEMBRE.Columns(4).Width = 130
        Me.LISTE MEMBRE.Columns(5).Width = 134
       Me.LISTE MEMBRE.Columns(6).Width = 130
       Me.LISTE MEMBRE.Columns(7).Width = 130
        Me.LISTE_MEMBRE.Columns(0).Name = "NUM LICENCE"
       Me.LISTE MEMBRE.Columns(1).Name = "NOM CLUB"
        Me.LISTE MEMBRE.Columns(2).Name = "NOM_MEMBRE"
        Me.LISTE MEMBRE.Columns(3).Name = "PRENOM MEMBRE"
       Me.LISTE MEMBRE.Columns(4).Name = "DATE NAISSANCE"
       Me.LISTE MEMBRE.Columns(5).Name = "ADR RUE MEMBRE"
        Me.LISTE MEMBRE.Columns(6).Name = "CODE POST MEMBRE"
        Me.LISTE MEMBRE.Columns(7).Name = "ADR VILLE MEMBRE"
        '** Evite d'avoir une ligne blanche en fin, destinée à la
        '** saisie d'une nouvelle ligne
        Me.LISTE MEMBRE.AllowUserToAddRows = False
        Dim regAffich As String
```

```
regAffich = "SELECT CLUB.NOM CLUB, CLUB.NUM CLUB AS CLUB NUM CLUB,
INSCRIPTION.NUM COMPETITION, INSCRIPTION.NUM LICENCE AS
INSCRIPTION NUM LICENCE, MEMBRE.NUM LICENCE AS MEMBRE NUM LICENCE,
MEMBRE.NUM CLUB AS MEMBRE NUM CLUB, MEMBRE.NOM MEMBRE, MEMBRE.PRENOM MEMBRE,
MEMBRE.DATE NAISSANCE, MEMBRE.ADR RUE MEMBRE, MEMBRE.CODE POST MEMBRE,
MEMBRE.ADR_VILLE MEMBRE
        FROM (CLUB INNER JOIN MEMBRE ON CLUB.[NUM CLUB] = MEMBRE.[NUM CLUB])
INNER JOIN INSCRIPTION ON MEMBRE.[NUM LICENCE] = INSCRIPTION.[NUM LICENCE]
WHERE NUM COMPETITION=@NUM COMPETITION AND NOTE GLOBALE IS NULL"
        'On déclare une variable de type objet "Command" qui exécute une
commande (requête SQL) sur la base de données
        Dim cmdAffich As New OleDbCommand(regAffich, maConnexion)
        Dim sup As Integer
        sup = Competitions.NumCompetitionSelectionne
        cmdAffich.Parameters.AddWithValue("@NUM_COMPETITION", sup)
        'On déclare une variable de type objet "DataReader" qui récupère les
enregistrements de la requête de type SELECT
        Dim rdrRech As OleDbDataReader = cmdAffich.ExecuteReader()
        Dim numligne As Integer
       While rdrRech.Read()
            Me.LISTE MEMBRE.Rows.Add()
            numligne = Me.LISTE MEMBRE.RowCount - 1
            Me.LISTE MEMBRE.Item(0, numligne).Value =
rdrRech.Item("INSCRIPTION NUM LICENCE")
            Me.LISTE MEMBRE.Item(1, numligne).Value =
rdrRech.Item("NOM CLUB")
            Me.LISTE MEMBRE.Item(2, numligne).Value =
rdrRech.Item("NOM MEMBRE")
            Me.LISTE MEMBRE.Item(3, numligne).Value =
rdrRech.Item("PRENOM MEMBRE")
            Me.LISTE MEMBRE.Item(4, numligne).Value =
rdrRech.Item("DATE NAISSANCE")
            Me.LISTE_MEMBRE.Item(5, numligne).Value =
rdrRech.Item("ADR RUE MEMBRE")
            Me.LISTE MEMBRE.Item(6, numligne).Value =
rdrRech.Item("CODE POST MEMBRE")
            Me.LISTE MEMBRE.Item(7, numligne).Value =
rdrRech.Item("ADR VILLE MEMBRE")
        End While
   End Sub
```

Nous ne détaillerons pas l'affichage des données étant donné qu'il s'agit de la même chose que précédemment hormis la requête SQL



### $lap{}$ Explications:

Private Sub FormInscriptionMembreCompetition\_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Lors de l'évènement "Load", soit le chargement du Forms, nous allons définir les colonnes de notre DataGridView

SELECT CLUB.NOM CLUB, CLUB.NUM CLUB AS CLUB NUM CLUB, INSCRIPTION.NUM COMPETITION, INSCRIPTION.NUM LICENCE AS INSCRIPTION NUM LICENCE, MEMBRE.NUM LICENCE AS MEMBRE NUM LICENCE, MEMBRE.NUM CLUB AS MEMBRE NUM CLUB, MEMBRE.NOM MEMBRE, MEMBRE.PRENOM MEMBRE, MEMBRE.DATE NAISSANCE, MEMBRE.ADR RUE MEMBRE, MEMBRE.CODE POST MEMBRE, MEMBRE.ADR VILLE MEMBRE FROM (CLUB INNER JOIN MEMBRE ON CLUB. [NUM CLUB] = MEMBRE. [NUM CLUB]) INNER JOIN INSCRIPTION ON MEMBRE.[NUM LICENCE] = INSCRIPTION.[NUM LICENCE] WHERE NUM COMPETITION=@NUM COMPETITION AND NOTE GLOBALE IS NULL

#### Cette requête est de type **SELECT** permettant d'afficher :

NUM_LICENCE NOM_CLUB NOM_MEME	RE PRENOM_MEMBRE DATE_NAISSANC	CE ADR_RUE_MEMBRE CODE_PO	ST_MEMBRE ADR_VILLE_MEMBRE
-------------------------------	--------------------------------	---------------------------	----------------------------

Nous allons ensuite récupérer le **NUM COMPETITION** sélectionné précédemment, ainsi que le NUM LICENCE de même sélectionner sur cette même page. Nous établirons la requête permettant sa suppression dans la BBD.

# Jeu d'essai :



# Maintenant gérons le bouton :

Private Sub Button1 Click(sender As Object, e As EventArgs) Handles suppr.Click

```
'On déclare une variable qui récupère la ligne sélectionnée sur le
DataGrid
        Dim LigneSectionne As Integer
        LigneSectionne = (Me.LISTE MEMBRE.CurrentRow.Index)
        'On assigne la valeur NUM LICENCE à une variable
        Dim NumLicenceSelectionne As String
       NumLicenceSelectionne = Me.LISTE MEMBRE.Item(0,
LigneSectionne).Value
        'Try Catch test doublon
        Try
            Dim req As String
            'On écrit une requête de type DELETE
            reg = "DELETE FROM INSCRIPTION WHERE NUM LICENCE=@NUM LICENCE &
NUM COMPETITION=@NUM COMPETITION"
            'On déclare une variable de type objet "Command" qui exécute une
commande (requête SQL) sur la base de données
            Dim cmdDelete As New OleDbCommand(reg, maConnexion)
            'Passage des paramètres
            cmdDelete.Parameters.AddWithValue("@NUM LICENCE",
NumLicenceSelectionne)
            cmdDelete.Parameters.AddWithValue("@NUM COMPETITION",
Competitions.NumCompetitionSelectionne)
            'On exécute la requête dans la base de données
            cmdDelete.ExecuteNonQuery()
            MsgBox("Le membre a bien été supprimé de la compétition...")
        Catch ex As Exception
            MessageBox.Show("Le membre est déjà supprimé de la compétition")
        End Try
       Me.Close()
   End Sub
```

# Explications:

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles
Button1.Click

S'exécute lors de l'évènement "Click" sur le bouton.

Nous ne détaillerons que les nouveautés et non les points abordés précédemment!

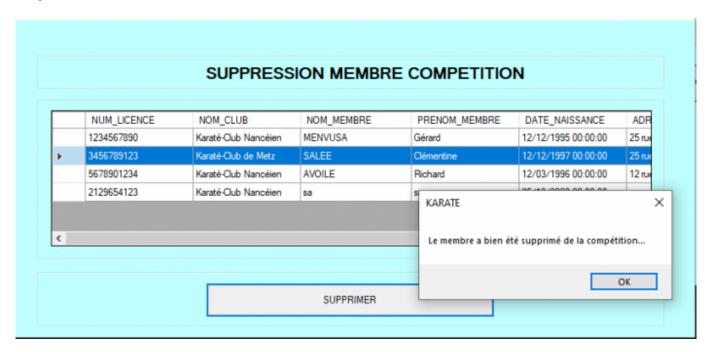
req = "DELETE FROM INSCRIPTION WHERE NUM\_LICENCE=@NUM\_LICENCE &
NUM\_COMPETITION=@NUM\_COMPETITION"

De type **DELETE**, nous supprimons l'enregistrement lié au paramètre **NUM\_LICENCE**, ainsi que le **NUM COMPETITION**.

Si nous avions traité cette requête qu'avec le **NUM\_LICENCE**, nous aurions supprimé toutes les inscriptions liées aux membres sur l'ensemble des compétitions. Or, nous souhaitons le supprimer que sur une compétition.

Comme précédemment nous utilisons un Try Catch afin de traiter les cas erreurs/exceptions.

# **Jeu d'essai finaux:**



#### PERMETTRE LA MODIFICATION D'UNE INSCRIPTION D'UN MEMBRE À UNE COMPÉTITION

La modification de l'inscription d'un membre n'est pas pertinente.



La table **INSCRIPTION** ne comporte que 3 champs : **NUM\_LICENCE**, **NUM\_COMPETITION**, **NOTE\_GLOBALE**.

Ces trois champs sont inchangeables et cela n'est pas pertinent de les modifier.

### **Objectif T2.5**

# Rappel de l'objectif :

→ Affichage des résultats d'une compétition donnée

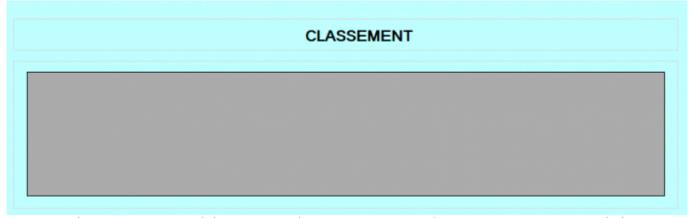
- → Affichage de la liste des jurys d'une compétition
- → Mise à jour de la note globale obtenue par chaque compétiteur

#### AFFICHAGE DES RÉSULTATS D'UNE COMPÉTITION DONNÉE

Pour répondre à ce besoin, nous avons décidé de créer une nouvelle page : FormClassementCompetition.vb



### 🞾 La page actuelle : FormClassementCompetition.vb[Design]



Nous ne détaillerons pas les éléments utilisés car il s'agit des mêmes que le formulaire précédent.

Ajoutons dès maintenant un bouton sur la page principale (Competitions.vb), permettant de récupérer le choix de la compétition ainsi que rediriger l'utilisateur vers le classement.

Nous avons donc rajouter un bouton **CLASSEMENT** sur la page *Competitions.vb*:



Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Classement.Click 'On déclare une variable qui recupère la ligne selectionnée sur le DataGrid Dim LigneSectionne As Integer LigneSectionne = (Me.LISTE COMPETITION.CurrentRow.Index) 'On assigne la valeur NUM LICENCE à une variable NumCompetitionSelectionne = Me.LISTE COMPETITION.Item(0, LigneSectionne).Value FormClassementCompetition.Show()

#### End Sub

Nous ajoutons un nouveau bout de code lors de l'évènement "Click" sur ce nouveau bouton permettant de récupérer le choix de la compétition : NUM COMPETITION, puis on affiche FormClassementCompetition.vb

Passons directement au développement :

### La page actuelle : FormClassementCompetition.vb

```
Me.Classement.Rows.Clear()
        Me.Classement.ColumnCount = 6
       Me.Classement.Columns(0).Width = 115
       Me.Classement.Columns(1).Width = 132
       Me.Classement.Columns(2).Width = 132
       Me.Classement.Columns(3).Width = 132
       Me.Classement.Columns(4).Width = 132
       Me.Classement.Columns(5).Width = 132
       Me.Classement.Columns(0).Name = "CLASSEMENT"
       Me.Classement.Columns(1).Name = "NUM LICENCE"
       Me.Classement.Columns(2).Name = "NOM MEMBRE"
       Me.Classement.Columns(3).Name = "PRENOM MEMBRE"
       Me.Classement.Columns(4).Name = "NOM CLUB"
       Me.Classement.Columns(5).Name = "NOTE GLOBALE"
        '** Evite d'avoir une ligne blanche en fin, destinée à la saisie
d'une nouvelle ligne
        Me.Classement.AllowUserToAddRows = False
        Dim req As String
        reg = "SELECT MEMBRE.NUM LICENCE, MEMBRE.NOM MEMBRE,
MEMBRE.PRENOM MEMBRE, CLUB.NOM CLUB, INSCRIPTION.NOTE GLOBALE
        FROM (CLUB INNER JOIN MEMBRE ON CLUB. [NUM CLUB] = MEMBRE. [NUM CLUB])
INNER JOIN INSCRIPTION ON MEMBRE.[NUM LICENCE] = INSCRIPTION.[NUM LICENCE]
        WHERE NUM COMPETITION=@NUM COMPETITION AND NOTE GLOBALE IS NOT NULL
ORDER BY NOTE GLOBALE DESC"
        'On déclare une variable de type objet "Command" qui exécute une
commande (requête SOL) sur la base de données
        Dim cmdAffichage As New OleDbCommand(req, maConnexion)
        cmdAffichage.Parameters.AddWithValue("@NUM_COMPETITION",
Competitions.NumCompetitionSelectionne)
        Dim rdrRech As OleDbDataReader = cmdAffichage.ExecuteReader()
        Dim numligne As Integer
       While rdrRech.Read()
           Me.Classement.Rows.Add()
```

## Explications:

```
Private Sub FormClassementCompetition_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
```

Lors de l'évènement "Load", soit le chargement du Forms, nous allons définir les colonnes de notre DataGridView

Ensuite, nous devons permettre l'affichage des informations de la base de données à l'intérieur de ce dernier à travers une requête SQL :

```
req = "SELECT MEMBRE.NUM_LICENCE, MEMBRE.NOM_MEMBRE,
MEMBRE.PRENOM_MEMBRE, CLUB.NOM_CLUB, INSCRIPTION.NOTE_GLOBALE
        FROM (CLUB INNER JOIN MEMBRE ON CLUB.[NUM_CLUB] = MEMBRE.[NUM_CLUB])
INNER JOIN INSCRIPTION ON MEMBRE.[NUM_LICENCE] = INSCRIPTION.[NUM_LICENCE]
        WHERE NUM_COMPETITION=@NUM_COMPETITION AND NOTE_GLOBALE IS NOT NULL
ORDER BY NOTE_GLOBALE DESC"
```

Cette requête comporte 2 jointures, elle permet de récupérer les informations du membre, le nom de son club, ainsi que sa note globale, nous avons ajouté 2 conditions, la note globale ne doit pas être nulle (pour un classement c'est pas top...), et nous trions ces mêmes notes par ordre décroissant.

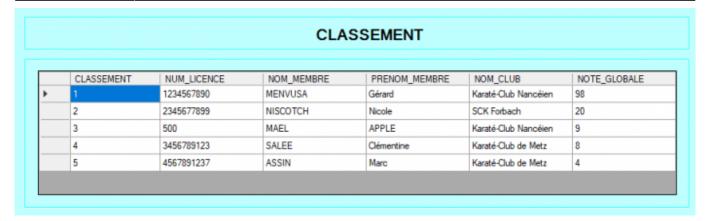
```
Dim numligne As Integer

While rdrRech.Read()

Me.Classement.Rows.Add()
    numligne = Me.Classement.RowCount - 1
    Me.Classement.Item(0, numligne).Value = (numligne + 1)
```

Nous avons ajouté un petit compteur afin d'afficher le rang du membre avec sa **NOTE \_GLOBALE**.

# Deu d'essai :



#### AFFICHAGE DE LA LISTE DES JURYS D'UNE COMPÉTITION

Nous allons opérer sur la même page (FormClassementCompetition.vb), en ajoutant lors de l'évènement load l'affichage d'un nouveau DataGridView afin d'afficher la liste des jurys.

# La page actuelle: FormClassementCompetition.vb[Design]



Nous ne détaillerons pas les éléments utilisés car il s'agit des mêmes que le formulaire précédent.

# La page actuelle: FormClassementCompetition.vb

```
'** Traitement du deuxieme datagrid
Me.Jurys.Rows.Clear()

Me.Jurys.ColumnCount = 4
Me.Jurys.Columns(0).Width = 115
Me.Jurys.Columns(1).Width = 132
Me.Jurys.Columns(2).Width = 132
Me.Jurys.Columns(3).Width = 132
```

```
Me.Jurys.Columns(0).Name = "NUM JURY"
        Me.Jurys.Columns(1).Name = "NOM ENTRAINEUR"
        Me.Jurys.Columns(2).Name = "PRENOM ENTRAINEUR"
        Me.Jurys.Columns(3).Name = "NOM CLUB"
        Me.Jurys.Rows.Clear()
        '** Evite d'avoir une ligne blanche en fin, destinée à la saisie
d'une nouvelle ligne
       Me.Jurys.AllowUserToAddRows = False
        Dim reqJury As String
        regJury = "SELECT JUGE.NUM JURY, ENTRAINEUR.NOM ENTRAINEUR,
ENTRAINEUR.PRENOM ENTRAINEUR, CLUB.NOM CLUB
        FROM (CLUB INNER JOIN ENTRAINEUR ON CLUB. [NUM CLUB] =
ENTRAINEUR.[NUM CLUB]) INNER JOIN JUGE ON ENTRAINEUR.[NUM ENTRAINEUR] =
JUGE.[NUM ENTRAINEUR] WHERE NUM COMPETITION=@NUM COMPETITION AND NUM JURY IS
NOT NULL ORDER BY ENTRAINEUR.NUM ENTRAINEUR"
        'On déclare une variable de type objet "Command" qui exécute une
commande (requête SQL) sur la base de données
       Dim cmdAffichageJury As New OleDbCommand(reqJury, maConnexion)
        cmdAffichageJury.Parameters.AddWithValue("@NUM COMPETITION",
Competitions.NumCompetitionSelectionne)
        Dim rdrRechJury As OleDbDataReader =
cmdAffichageJury.ExecuteReader()
        Dim numligneJury As Integer
       While rdrRechJury.Read()
            Me.Jurys.Rows.Add()
            numligneJury = Me.Jurys.RowCount - 1
            Me.Jurys.Item(0, numligneJury).Value =
rdrRechJury.Item("NUM JURY")
            Me.Jurys.Item(1, numligneJury).Value =
rdrRechJury.Item("NOM ENTRAINEUR")
            Me.Jurys.Item(2, numligneJury).Value =
rdrRechJury.Item("PRENOM ENTRAINEUR")
            Me.Jurys.Item(3, numligneJury).Value =
rdrRechJury.Item("NOM CLUB")
        End While
```

Explications: Ce bout de code est placé dans l'évènement load.

```
Private Sub FormClassementCompetition_Load(sender As Object, e As
EventArgs) Handles MyBase.Load
```

Ensuite, nous devons permettre l'affichage des jurys de la base de données à l'intérieur de ce DataGridView à travers une requête SQL :

Cette requête comporte 2 jointures, elle permet de récupérer les informations des entraîneurs, le numéro du jury... Nous avons ajouté une condition, **NUM\_JURY** ne doit pas être nulle.

# Deu d'essai :

			CLASS	SEMENT		
_	CLASSEMENT	NUM_LICENCE	NOW NEMBER	DDENOM MEMBRE	NOM CLUB	NOTE CLOSALE
<b>-</b>	1	1234567890	NOM_MEMBRE MENVUSA	PRENOM_MEMBRE Gérard	NOM_CLUB Karaté-Club Nancéien	NOTE_GLOBALE 98
_	2	2345677899	NISCOTCH	Nicole	SCK Forbach	20
	3	500	MAEL	APPLE	Karaté-Club Nancéien	9
	4	3456789123	SALEE	Clémentine	Karaté-Club de Metz	8
	5	4567891237	ASSIN	Marc	Karaté-Club de Metz	4
	-	1001001201				
		CETTE COMPE	PRENOM_ENTRAINEUR			
JUI	RYS POUR	CETTE COMPE    NOM_ENTRAINEUR   TOURNIE	PRENOM_ENTRAINEUR	Karaté-Club Nancéien		
	RYS POUR    NUM_JURY   1   2	NOM_ENTRAINEUR TOURNIE BOURGEOIS	PRENOM_ENTRAINEUR Alain Agnès	Karaté-Club Nancéien Karaté-Club Nancéien		
	RYS POUR  NUM_JURY  1 2 3	NOM_ENTRAINEUR TOURNIE BOURGEOIS BOGUSZ	PRENOM_ENTRAINEUR Alain Agnès Thieny	Karaté-Club Nancéien Karaté-Club Nancéien Karaté-Club de Metz		
	RYS POUR    NUM_JURY   1   2	NOM_ENTRAINEUR TOURNIE BOURGEOIS	PRENOM_ENTRAINEUR Alain Agnès	Karaté-Club Nancéien Karaté-Club Nancéien		

## MISE À JOUR DE LA NOTE GLOBALE OBTENUE PAR CHAQUE COMPÉTITEUR

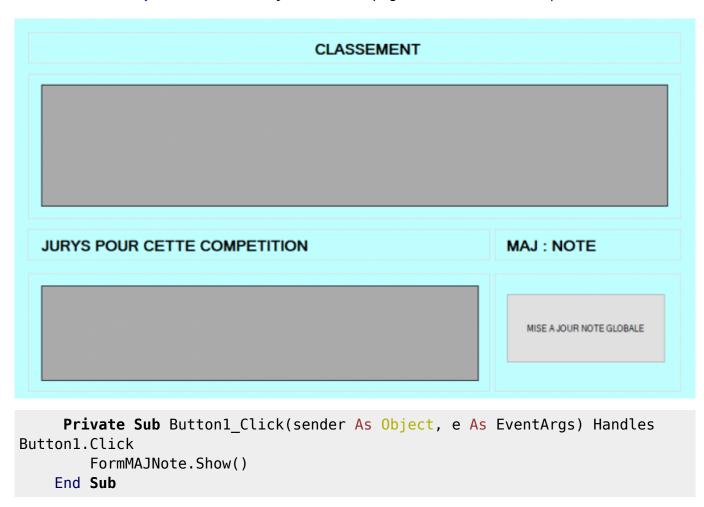
Pour répondre à ce dernier objectif, nous avons décidé de créer une nouvelle page : FormMAJNote.vb

La page actuelle : FormMAJNote.vb[Design]



Nous ne détaillerons pas les éléments utilisés car il s'agit des mêmes que le formulaire précédent.

Nous avons donc rajouté un bouton MAJ NOTE sur la page FormClassementCompetition.vb:



Nous ajoutons un nouveau bout de code lors de l'évènement "Click" affiche FormMAJNote.vb

Passons au développement :

La page actuelle : FormMAJNote.vb

```
Private Sub FormMAJNote Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        Me.LISTE MEMBRE.Rows.Clear()
        Me.LISTE MEMBRE.ColumnCount = 8
        Me.LISTE MEMBRE.Columns(0).Width = 130
        Me.LISTE MEMBRE.Columns(1).Width = 130
        Me.LISTE MEMBRE.Columns(2).Width = 134
        Me.LISTE MEMBRE.Columns(3).Width = 130
        Me.LISTE MEMBRE.Columns(4).Width = 130
        Me.LISTE MEMBRE.Columns(5).Width = 134
        Me.LISTE MEMBRE.Columns(6).Width = 130
        Me.LISTE MEMBRE.Columns(7).Width = 130
        Me.LISTE MEMBRE.Columns(0).Name = "NUM LICENCE"
        Me.LISTE MEMBRE.Columns(1).Name = "NOM CLUB"
        Me.LISTE MEMBRE.Columns(2).Name = "NOM MEMBRE"
        Me.LISTE MEMBRE.Columns(3).Name = "PRENOM MEMBRE"
        Me.LISTE MEMBRE.Columns(4).Name = "DATE NAISSANCE"
        Me.LISTE MEMBRE.Columns(5).Name = "ADR RUE MEMBRE"
        Me.LISTE MEMBRE.Columns(6).Name = "CODE_POST_MEMBRE"
        Me.LISTE MEMBRE.Columns(7).Name = "ADR VILLE MEMBRE"
        '** Evite d'avoir une ligne blanche en fin, destinée à la
        '** saisie d'une nouvelle ligne
        Me.LISTE MEMBRE.AllowUserToAddRows = False
        Dim regAffich As String
        regAffich = "SELECT CLUB.NOM CLUB, CLUB.NUM CLUB AS CLUB NUM CLUB,
INSCRIPTION.NOTE_GLOBALE, INSCRIPTION.NUM COMPETITION,
INSCRIPTION.NUM_LICENCE AS INSCRIPTION_NUM_LICENCE, MEMBRE.NUM LICENCE AS
MEMBRE NUM LICENCE, MEMBRE.NUM CLUB AS MEMBRE NUM CLUB, MEMBRE.NOM MEMBRE,
MEMBRE.PRENOM MEMBRE, MEMBRE.DATE NAISSANCE, MEMBRE.ADR RUE MEMBRE,
MEMBRE.CODE POST MEMBRE, MEMBRE.ADR VILLE MEMBRE
        FROM (CLUB INNER JOIN MEMBRE ON CLUB.[NUM CLUB] = MEMBRE.[NUM_CLUB])
INNER JOIN INSCRIPTION ON MEMBRE.[NUM LICENCE] = INSCRIPTION.[NUM LICENCE]
WHERE NUM COMPETITION=@NUM COMPETITION"
        'On déclare une variable de type objet "Command" qui exécute une
commande (requête SQL) sur la base de données
        Dim cmdAffich As New OleDbCommand(regAffich, maConnexion)
        Dim sup As Integer
        sup = Competitions.NumCompetitionSelectionne
        cmdAffich.Parameters.AddWithValue("@NUM COMPETITION", sup)
        'On déclare une variable de type objet "DataReader" qui récupère les
enregistrements de la requête de type SELECT
        Dim rdrRech As OleDbDataReader = cmdAffich.ExecuteReader()
        Dim numligne As Integer
        While rdrRech.Read()
```

```
Me.LISTE MEMBRE.Rows.Add()
            numligne = Me.LISTE MEMBRE.RowCount - 1
            Me.LISTE MEMBRE.Item(0, numligne).Value =
rdrRech.Item("INSCRIPTION NUM LICENCE")
            Me.LISTE MEMBRE.Item(1, numligne).Value =
rdrRech.Item("NOM CLUB")
            Me.LISTE MEMBRE.Item(2, numligne).Value =
rdrRech.Item("NOM MEMBRE")
            Me.LISTE MEMBRE.Item(3, numligne).Value =
rdrRech.Item("PRENOM MEMBRE")
            Me.LISTE MEMBRE.Item(4, numligne).Value =
rdrRech.Item("DATE NAISSANCE")
            Me.LISTE MEMBRE.Item(5, numligne).Value =
rdrRech.Item("ADR RUE MEMBRE")
            Me.LISTE MEMBRE.Item(6, numligne).Value =
rdrRech.Item("CODE POST MEMBRE")
            Me.LISTE MEMBRE.Item(7, numligne).Value =
rdrRech.Item("ADR VILLE MEMBRE")
        End While
```

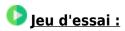
Nous ne détaillerons pas l'affichage des données dans le DataGridView étant donné qu'il s'agit de la même chose que précédemment

# $\stackrel{ extbf{Q}}{=} Explications:$

```
Private Sub FormMAJNote_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
```

Lors de l'évènement "Load", soit le chargement du Forms, nous allons définir les colonnes de notre DataGridView

Cette requête est de type **SELECT** permettant d'afficher les membres et **NOM\_CLUB**, la **NOTE\_GLOBALE** et l'ensemble des informations **MEMBRE**.



	NUM_LICENCE	NOM_CLUB	NOM_MEMBRE	PRENOM_MEMBRE	DATE_NAISSANCE	A
•	1234567890	Karaté-Club Nancéien	MENVUSA	Gérard	12/12/1995 00:00:00	25
	2345677899	SCK Forbach	NISCOTCH	Nicole	11/05/1996 00:00:00	18
	3456789123	Karaté-Club de Metz	SALEE	Clémentine	12/12/1997 00:00:00	25
	4567891237	Karaté-Club de Metz	ASSIN	Marc	10/01/1997 00:00:00	cł
	500	Karaté-Club Nancéien	MAEL	APPLE	15/10/2000 00:00:00	Lil_
	2129654123	Karaté-Club Nancéien	sa	sa	25/10/2000 00:00:00	Sē."
4						<b>•</b>

## Maintenant gérons le bouton :

```
Private Sub Button1 Click(sender As Object, e As EventArgs) Handles
suppr.Click
        'On déclare une variable qui recupère la ligne selectionnée sur le
DataGrid
       Dim LigneSectionne As Integer
        LigneSectionne = (Me.LISTE MEMBRE.CurrentRow.Index)
        'On assigne la valeurs NUM LICENCE a une variable
        Dim NumLicenceSelectionne As String
       NumLicenceSelectionne = Me.LISTE MEMBRE.Item(0,
LigneSectionne).Value
        'Récupération des notes de la table NOTE
        Dim reqnote As String
        'On écrit une requête de type SELECT
        regnote = "SELECT SELECT Avg(NOTE) As [NOTE] WHERE
NUM LICENCE=@NUM LICENCE AND NUM COMPETITION=@NUM COMPETITION"
        'On déclare une variable de type objet "Command" qui exécute une
commande (requête SQL) sur la base de données
        Dim cmdSelectNote As New OleDbCommand(regnote, maConnexion)
        cmdSelectNote.Parameters.AddWithValue("@NUM_LICENCE",
NumLicenceSelectionne)
```

http://ppe.boonum.fr/ Printed on 2022/04/04 20:22

cmdSelectNote.Parameters.AddWithValue("@NUM COMPETITION",

'On exécute la requête dans la base de données

Competitions.NumCompetitionSelectionne)

```
Dim rdrRecher As OleDbDataReader = cmdSelectNote.ExecuteReader()
        rdrRecher.Read()
        'Try Catch test doublon
            Dim req As String
            'On écrit une requête de type UPDATE
            req = "UPDATE INSCRIPTION SET NOTE GLOBALE=@NOTE GLOBALE WHERE
NUM LICENCE=@NUM LICENCE AND NUM COMPETITION=@NUM COMPETITION"
            'On déclare une variable de type objet "Command" qui exécute une
commande (requête SQL) sur la base de données
            Dim cmdDelete As New OleDbCommand(reg, maConnexion)
            'Passage des paramètres
            cmdDelete.Parameters.AddWithValue("@NOTE GLOBALE",
rdrRecher.Item("NOTE"))
            cmdDelete.Parameters.AddWithValue("@NUM LICENCE",
NumLicenceSelectionne)
            cmdDelete.Parameters.AddWithValue("@NUM COMPETITION",
Competitions.NumCompetitionSelectionne)
            'On exécute la requête dans la base de données
            cmdDelete.ExecuteNonQuery()
            MsgBox("La note a bien été mise à jour !")
        Catch ex As Exception
            MessageBox.Show("Erreur !")
        End Try
        Me.Close()
```

# Explications:

Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles
suppr.Click

S'exécute lors de l'évènement "Click" sur le bouton.

```
Dim LigneSectionne As Integer
LigneSectionne = (Me.LISTE_MEMBRE.CurrentRow.Index)
```

On déclare une variable qui récupère la ligne sélectionnée sur le DataGridView

```
Dim NumLicenceSelectionne As String
NumLicenceSelectionne = Me.LISTE_MEMBRE.Item(0, LigneSectionne).Value
Dim CompetSelect As String
CompetSelect = Competitions.NumCompetitionSelectionne
```

On récupère le premier élément du DataGridView et on le stock dans une variable, NumLicenceSelectionne.

Puis nous récupérons *NumCompetitionSelectionne* de la page **Competitions.vb** que nous stockons également.

Nous devons maintenant écrire 2 requêtes : une qui sélectionne l'ensemble des notes obtenu par un membre inscrit pour une compétition, puis une qui récupère le resultat de la requête précédente et met a jour la nouvelle **NOTE GLOBALE** du membre dans la table **INSCRIPTION**.

Notre première requète :

```
reqnote = "SELECT Avg(NOTE) As [NOTE] FROM [NOTE] WHERE
NUM_LICENCE=@NUM_LICENCE AND NUM_COMPETITION=@NUM_COMPETITION"
```

De type SELECT afin de faire ma moyenne a l'aide de l'agrégat **AVG** de l'ensemble des notes du membres appartenant à la compétition sélectionné précedemment.

Notre deuxième requête :

```
req = "UPDATE INSCRIPTION SET NOTE_GLOBALE=@NOTE_GLOBALE WHERE
NUM_LICENCE=@NUM_LICENCE AND NUM_COMPETITION=@NUM_COMPETITION""
```

De type UPDATE, nous mettons à jour le champs NOTE GLOBALE de la table INSCRIPTION.

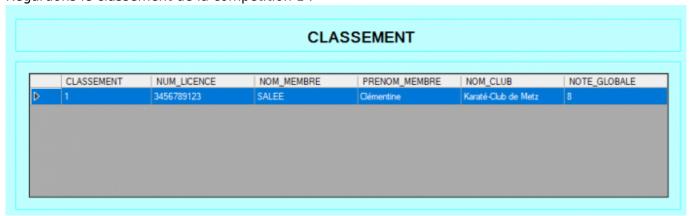
Comme précédemment nous utilisons un Try Catch afin de vérifier qu'il n'y aura pas de doublon dans la table **INSCRIPTION**.

# Deu d'essai finaux:

La table NOTE de la base de donnée pour le NUM LICENCE : 1234567890

1	NUM_COM -	NUM_LICE1 -	NUM_ENTRAINE -	NOTE -	-
	1	1234567890	1		8
	1	1234567890	2		8
	1	1234567890	3	1	0
	1	1234567890	4	7,	5
	1	1234567890	5		8

Regardons le classement de la compétition 1 :

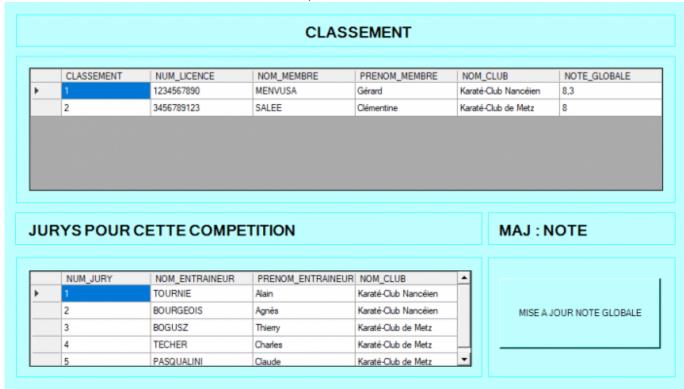


Nous remarquons que nous n'avons pas mis à jour la NOTE\_GLOBALE pour le membre dont la licence est 1234567890

Nous allons donc mettre à jour sa NOTE\_GLOBALE :



Vérifions maintenant le classement de la compétition 1 :



Nous remarquons que le membre dont la licence est 1234567890 à bien été ajouté au classement. Et la moyenne de l'ensemble de ses notes définissent sa **NOTE\_GLOBALE**.

Tout fonctionne! Nous pouvons d'ores et déjà présenter la version 1.0 de notre petite application Windows Forms!



Nous utilisons *try catch* pour chacune de nos requêtes mais par soucis de clarté nous ne les avons pas afficher ici  $\stackrel{\textcircled{\tiny }}{\oplus}$ 

From:

http://ppe.boonum.fr/ - AP.SIO

Permanent link:

http://ppe.boonum.fr/doku.php?id=slam:ws:2021:sio:ap2.1:equipe9:accueil



