

# Compte Rendu de la Mission 1

Maëva Fluteau et Rémi Dubois

## Objectifs : Réalisation des tâches suivantes : T1.1, T1.4

1. T1.1 Gestion des membres : Cette tâche doit permettre la création, la modification et la suppression d'un membre à partir d'un formulaire. Il faudra réaliser la maquette de ce formulaire, le faire valider par le chef de projet avant de commencer le codage.
2. T1.4 Édition des licences : Cette tâche doit permettre l'édition de la licence de chaque membre sous le format suivant :

### T1.1 Gestion des Membres

Ce travail a été séparé en deux formulaires dans le but de proposer une mise en forme plus cohérente, l'un concerne l'ajout des membres, l'autre leur modification et suppression.

#### Formulaire d'Ajout

Ce formulaire consiste tout simplement à permettre l'ajout de membres dans la Table "Membre" de la base de donnée en respectant la mise en forme imposée de la mission

Il est composé de plusieurs de plusieurs zones de textes qui sont codifiées : *TxtNomDuTexte* et d'une liste déroulante : *IstNomDeLaListe* Pour que cette dernière puisse afficher ce que nous souhaitons, ici la liste des clubs, nous lui avons mit en critères la requête suivante : `<code sql>SELECT [CLUB].[NUMCLUB], [CLUB].[NOMCLUB] FROM CLUB ORDER BY [NOMCLUB];</code>`

Le bouton "Ajouter" du formulaire permet l'enregistrement des informations saisies par l'utilisateur à condition d'y intégrer un bloc de code VBA pour initialiser cette action. Ce bloc s'appelle : `Private Sub CmdAjouterClick()`

Afin de garder une cohérence dans les numéros de licence, nous avons tout d'abord imposé leur saisie à 10 caractères, sinon, un message d'erreur apparaît.

```
'----- gestion des erreurs de saisie
If Len(Txt_Licence) <> 10 Then
MsgBox "Licence Incorrecte, tapez 10 Chiffres", vbExclamation, "Gestion
Membre"
    Txt_Licence.SetFocus 'positionner le curseur sur le champ txt_Nom
Exit Sub 'Quitter la procédure
End If
```

Ici, nous avons adapté le code à notre formulaire, nous précisons à gauche le nom de la table et des champs qui doivent recevoir les informations saisies du formulaires, qui sont à droite.

```
'----- ajouter la personne dans la table membre
Set cMembre = CurrentDb().OpenRecordset("membre")
cMembre.AddNew
cMembre!NOM_MEMBRE = Txt_Nom
cMembre!PRENOM_MEMBRE = Txt_Prénom
cMembre!DATE_NAISSANCE = Txt_Date_Naissance
cMembre!ADR_VILLE_MEMBRE = txt_Ville
cMembre!ADR_RUE_MEMBRE = Txt_Adresse
cMembre!CODE_POST_MEMBRE = Txt_Code_Postal
cMembre!NUM_LICENCE = Txt_Licence
cMembre!NUM_CLUB = Txt_Num_Club
cMembre.Update
```

Ce code, une fois le membre ajouté à la base de donnée, va réinitialiser les champs de saisie de façon à les rendre vierge pour une prochaine saisie.

```
'initialiser les champs
Txt_Nom = "": Txt_Prénom = "":
Txt_Date_Naissance = "": txt_Ville = "":
Txt_Adresse = "": Txt_Code_Postal = "":
Txt_Licence = "": Txt_Num_Club = ""
```

A chaque ajout de membre, un message apparaît pour signaler à l'utilisateur qu'il a bien été ajouté dans la table.

```
MsgBox "Membre ajouté avec succès"
```

Enfin, le bouton "fermer" quand à lui permet à l'utilisateur de fermer le formulaire grâce au code suivant :

```
Private Sub Cmd_Fermer_Click()
DoCmd.Close
End Sub
```

### Formulaire de Modification

Ce formulaire, à la différence du précédent, permet à l'utilisateur de modifier ou supprimer les données contenues dans la table "Membre" avec toujours impératif de respecter la mise en forme.

Tout comme l'ajout, ce formulaire est composé de champs de texte et de liste déroulantes. à la différence du premier, nous avons ajouté une liste affichants le N° de licence, le nom et le prénom d'un membre, qui a pour but de faciliter la section d'un membre précis dont on souhaite faire afficher les informations en vue d'une modification ou suppression.

Pour faire afficher cette liste, il a là aussi fallu intégrer une requête dans les propriétés de la liste, mais pour bien obtenir les trois informations distinctes, il faut préciser dans le format le nombre de colonnes, ici 3.

```
SELECT [MEMBRE].[NUM_LICENCE], [MEMBRE].[NOM_MEMBRE],
[MEMBRE].[PRENOM_MEMBRE] FROM MEMBRE ORDER BY [NOM_MEMBRE];
```

Afin de compléter cette liste déroulante, il est nécessaire d'ajouter un code VBA "After Update" qui, en fonction du membre sélectionné de la liste déroulante, affichera les informations qui lui sont associées. Concernant la liste des clubs, nous avons ajouté un "Values" qui permet d'afficher le nom du club et non pas son numéro pour faciliter la modification des données par l'utilisateur.

```
Private Sub lst_Licence_AfterUpdate()
Dim bdKaraté As Database
Dim rsKaraté As Recordset
Dim requete As String
Set bdKaraté = CurrentDb()
requete = "SELECT * FROM MEMBRE WHERE NUM_LICENCE =" & lst_Licence & ""
Set rsKaraté = bdKaraté.OpenRecordset(requete, DB_OPEN_DYNASET)
Txt_Licence = rsKaraté!NUM_LICENCE
Txt_Nom = rsKaraté!NOM_MEMBRE

lst_Nom_Club.Value = rsKaraté!NUM_CLUB

Txt_Prénom = rsKaraté!PRENOM_MEMBRE
Txt_Date_Naissance = rsKaraté!DATE_NAISSANCE
Txt_Adresse = rsKaraté!ADR_RUE_MEMBRE
Txt_Code_Postal = rsKaraté!CODE_POST_MEMBRE
txt_Ville = rsKaraté!ADR_VILLE_MEMBRE
End Sub
```

Pour qu'on puisse modifier les informations, nous intégrons au bouton "Modifier" du code VBA, il faut là encore préciser les champs de la table qui recevront et sauvegarderont la modification des données.

```
'----- ajouter la personne dans la table membre
Set cMembre = CurrentDb().OpenRecordset("membre")
cMembre.Edit
cMembre!NOM_MEMBRE = Txt_Nom
cMembre!PRENOM_MEMBRE = Txt_Prénom
cMembre!DATE_NAISSANCE = Txt_Date_Naissance
cMembre!ADR_VILLE_MEMBRE = txt_Ville
cMembre!ADR_RUE_MEMBRE = Txt_Adresse
cMembre!CODE_POST_MEMBRE = Txt_Code_Postal
'cMembre!NUM_LICENCE = Txt_Licence 'le numéro de Licence n'est pas
modifiable
cMembre!NUM_CLUB = lst_Nom_Club
cMembre.Update
```

L'enregistrement se termine avec un message qui confirme le bon déroulement de la modification.

```
MsgBox "Membre modifié avec succès"
```

La dernière étape de ce formulaire consiste à supprimer de la base de donnée un membre, qui se fera

grâce au bouton "Supprimer" sur lequel nous avons intégré un code VBA. La requête sélectionne toutes les informations sur membre qui seront ensuite supprimé. Un message indique le bon déroulement de l'opération suivit de la réinitialisation de tous les champs.

```
Private Sub Cmd_Supprimer_Click()  
Dim bdKaraté As Database  
Dim rsKaraté As Recordset  
Dim requete As String  
Set bdKaraté = CurrentDb()  
requete = "SELECT * FROM MEMBRE WHERE NUM_LICENCE =" & lst_Licence & ""  
Set rsKaraté = bdKaraté.OpenRecordset(requete, DB_OPEN_DYNASET)  
  
rsKaraté.Delete  
  
'initialiser les champs  
Txt_Nom = "": Txt_Prénom = "":  
Txt_Date_Naissance = "": txt_Ville = "":  
Txt_Adresse = "": Txt_Code_Postal = "":  
Txt_Licence = "": lst_Nom_Club = "": lst_Licence = ""  
  
MsgBox "Membre Supprimé avec succès"
```

## T1.4 Édition des licences

L'édition des licences consiste ici à faire une mise en page des membres en fonction du format imposé. Pour se faire, nous avons d'abord créé une mise en page avec le nom des champs de la table à afficher sur la gauche et à droite, le nom du contenu du champs.

Afin de faire apparaître comme demandé la liste des membres, il faut au préalable créer une requête en cliquant sur la feuille de propriété. Nous avons donc fait la requête suivante :

Le résultat donne la mise en page suivante :

Le travail est terminé

## FIN

From:

/ - APs et stages du BTS SIO du lycée Suzanne Valadon

Permanent link:

[/doku.php/slam/ws/2011/ppe2.1/equipe1/accueil](http://doku.php/slam/ws/2011/ppe2.1/equipe1/accueil)

Last update: **2014/01/07 13:56**

