

Bases de Données

Access : (initiation) la base Access « associations, éventuellement d'utilité publique »

On considère le schéma relationnel suivant :

Les schémas des relations :

Membre (N°Membre , NomMembre , PrénomMembre)
 Appartenir (N°Membre , N°Association)
 Associations (N°Association , NomAssociation , ObjetAssociation , N°Dédution , DateDécret)
 Dédutions (N°Dédution , LigneDédution , PourcentageDédution , MaximumDédution)

Les contraintes de clé (primaire) :

Membre.N°Membre est la clé de Membre
 (Appartenir.N°Membre , Appartenir.N°Association) est la clé de Appartenir
 Associations.N°Association est la clé de Associations
 Dédutions.N°Dédution est la clé de Dédutions

Les contraintes d'intégrité référentielles :

Appartenir.N°Membre référence Membre.N°Membre
 ... et donc valeurs(Appartenir.N°Membre) \subseteq valeurs(Membre.N°Membre)
 Appartenir.N°Association référence Associations.N°Association
 ... et donc valeurs(Appartenir.N°Association) \subseteq valeurs(Associations.N°Association)
 Associations.N°Dédution référence Dédutions.N°Dédution
 ... et donc valeurs(Associations.N°Dédution) \subseteq valeurs(Dédutions.N°Dédution)

Les contraintes d'unicité :

Unicité de (NomMembre , PrénomMembre)
 Unicité de NomAssociation
 Unicité de LigneDédution

Les contraintes existentielles :

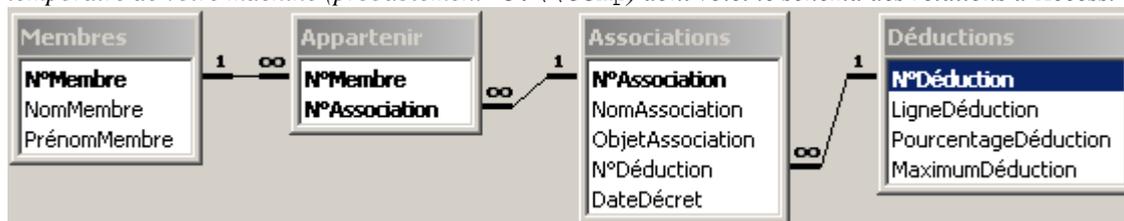
Seuls Associations.N°Dédution, DateDécret et MaximumDédution sont facultatifs

Les autres contraintes :

Associations.N°Dédution est renseigné si et seulement si DateDécret est renseigné

Par exemple, les dons versés à un organisme d'aide aux personnes en difficulté (comme pour l'association « UNICEF » reconnue d'utilité publique par un décret du 2 décembre 1970) donnent droit à 75% de réduction d'impôts (ligne UD, jusqu'à un maximum de 495 euros pour la déclaration 2009) tandis que les dons aux organismes d'intérêt général (comme pour l'association « Prévention Routière » reconnue d'utilité publique par un décret du 3 mai 1955) donnent droit à 66% de réduction d'impôts (ligne UF, Dons autres que UD, dans la limite de 20% du revenu imposable) ; en revanche, les dons versés à l'association des « Anciens étudiants du département Informatique de l'IUT Bordeaux 1 » ne donnent droit à aucune déduction fiscale.

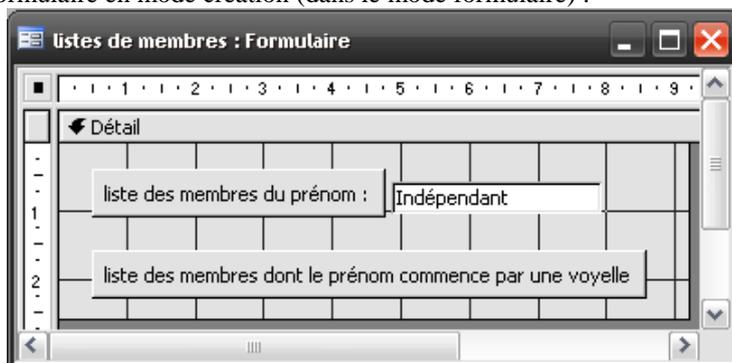
1. Copiez la base Access [\\Info\Bibliothèque\BD\Access\asso_util_publicq.mdb](#) sous le répertoire temporaire de votre machine (probablement c:\temp) dont voici le schéma des relations d'Access.



2. Créez une nouvelle table intitulée Adresses composée des sept champs N°Adresse (la clé, de type NuméroAuto), N°Rue (de type Numérique, entier court strictement positif), Rue (de type Texte de taille de champ 100), RueComplément (facultatif, de type Texte de taille de champ 100), CodePostal (de type Texte de taille de champ 5 de masque de saisie 0&000), Ville (de type Texte de taille de champ 25) et N°Association (de type Numérique de taille de champ Entier long) ; enregistrez-la.
3. Ajoutez Adresses dans le schéma des relations et créez un lien avec la table Associations qui vérifie la contrainte d'intégrité référentielle.
4. Modifiez la table Associations afin d'ajouter une propriété à la table (et non à l'un quelconque des champs) Valide si EstNull([DateDécret]) Eqv EstNull([N°Dédution]) avec comme Message si erreur « La date du décret est à renseigner si et seulement si l'association donne droit à une déduction fiscale. ».
5. Ouvrez la table Dédutions et saisissez deux déductions de votre choix.

6. Ouvrez la table *Associations* et saisissez deux associations, la première d'utilité publique (connectée à l'une des deux déductions précédemment saisies) mais pas la seconde ; essayez de ne pas affecter de date de décret pour la première association (finalement, saisissez une date) et d'en affecter une pour la seconde.
7. Créez une nouvelle requête en mode création basée sur la table *Déductions* affichant (dans cet ordre sur les champs) le numéro, la ligne et le pourcentage des déductions triées en ordre (alphabétique) croissant sur la ligne de déduction ; passez du mode de création de requête d'Access au code SQL (vérifiez que votre requête est proche de celle-ci : `SELECT Déductions.[N°Déduction], Déductions.LigneDéduction, Déductions.PourcentageDéduction FROM Déductions ORDER BY Déductions.LigneDéduction`) et inversement (les deux interagissent).
8. Modifiez le champ *N°Déduction* de la table *Associations* en changeant le texte de la légende et de la description.
9. Modifiez la propriété *Afficher le contrôle* de l'onglet *Liste de choix* du champ *N°Déduction* de la table *Associations* en optant pour une zone de liste déroulante : vérifiez que la valeur est bien 1 pour la propriété *Colonne liée*, indiquez la requête précédemment créée pour la propriété *Contenu*, saisissez 3 pour la propriété *Nbre colonnes*, et saisissez 0cm;1cm;1cm pour la propriété *Largeurs colonnes*.
10. Ouvrez la table *Associations* et saisissez une troisième association d'utilité publique (connectée à la même déduction que la première association).
11. Créez un nouveau formulaire¹ de type *Colonne simple* basé sur la table *Associations* (notez l'étiquette du numéro de déduction), consultez un à un les trois associations précédemment saisies en utilisant les boutons de déplacement (*[premier]*, *[précédent]*, *[suivant]* et *[dernier]*), et saisissez une quatrième association (utilisez le bouton de déplacement *[nouvel enregistrement]*) d'utilité publique connectée à la même déduction que les première et troisième associations (notez le texte de la barre d'état pour le numéro de déduction).
12. Créez un nouvel état² de type *tabulaire* basé sur la table *Associations*.
13. Créez une nouvelle macro dont l'action consiste à ouvrir l'état précédemment créé en choisissant *Aperçu avant impression* pour l'affichage.
14. Modifiez le formulaire basé sur la table *Associations* en rajoutant un bouton qui sur l'événement clic appelle la macro précédemment créée.
15. Créez un nouveau formulaire de type *Colonne simple* basé sur la table *Déductions*, et insérez³-lui un sous-formulaire basé sur la table *Associations* (champs correspondant au numéro de déduction, aux nom et objet de l'association). Modifiez les propriétés du sous-formulaire en choisissant le format d'affichage par défaut « *Feuille de données* ».
16. Créez une nouvelle macro composée de deux actions : la première ouvre le formulaire basé sur la table *Associations* en mode fenêtre *Boîte de dialogue* tandis que la seconde actualise le sous-formulaire.
17. Modifiez le formulaire basé sur la table *Déductions* en rajoutant un bouton qui sur l'événement clic exécute la macro précédente.
18. Ouvrez ce formulaire basé sur la table *Déductions*, cliquez sur le bouton que vous venez de rajouter, ajoutez une cinquième association⁴, sortez du formulaire basé sur la table *Associations* pour revenir sur le formulaire basé sur la table *Déductions*, consultez la liste (actualisée) des cinq numéros de déduction, et quittez ce formulaire.
19. Créez une nouvelle macro (dénommée *AutoExec*) qui affiche un message de bienvenue (action *BoîteMsg*).
20. Fermez la base Access (en l'enregistrant) ... et ouvrez-la de nouveau.
21. Créez un nouveau formulaire (en vous inspirant des copies d'écran ci-après) composé de deux boutons⁵ (dénommés *ListeMembres1Prénom* et *ListeMembresPrénomCommenceVoyelle*) et d'une zone de texte (dénommée *Prénom*).

Formulaire en mode création (dans le mode formulaire) :



¹ Si vous ne disposez pas de l'assistant, utilisez le mode création et insérez tous les champs.

² Si vous ne disposez pas de l'assistant, utilisez le mode création et insérez tous les champs en adoptant une présentation tabulaire.

³ Un simple glisser/déplacer du sous-formulaire depuis la fenêtre « Base de données » (onglet *Formulaires*) dans le formulaire principal en mode création devrait suffire.

⁴ Pour faire ce test correctement, la cinquième association ajoutée dans le formulaire basé sur la table *Associations* doit être de la même déduction que celle du formulaire basé sur la table *Déductions* ; vous noterez alors qu'en fermant le formulaire basé sur la table *Associations*, l'association ajoutée vient se rajouter dans le sous-formulaire basé sur la table *Associations*.

⁵ Cliquez sur le bouton *[Annuler]* pour éviter d'utiliser l'assistant de bouton de commande.

Propriétés du formulaire (dans le mode formulaire) :

Format	Données	Événement	Autres	Toutes
Légende				listes de membres
Affichage par défaut				Formulaire unique
Autoriser le mode Formulaire				Oui
Autoriser le mode Feuille de données				Non
Autoriser le mode Tableau croisé dynamique				Non
Autoriser le mode Graphique croisé dynamique				Oui
Barre défilement				Aucune
Afficher sélecteur				Non
Boutons de déplacement				Non
Diviseurs d'enregistrements				Non
Taille ajustée				Oui
Auto centrer				Non
Style bordure				Dimensionnable
Boîte contrôle				Oui
Boutons MinMax				Les deux activés
Bouton Fermer				Oui
Bouton Aide				Non
Largeur				9,198cm
Image				(aucune)
Type image				Intégré
Mode d'affichage de l'image				Découpage
Alignement de l'image				Centre
Mosaïque d'images				Non
Grille X				10
Grille Y				10
Polices imprimante				Non
Sous-feuille données hauteur				0cm
Sous-feuille données étendue				Non
Source palette				(Par défaut)
Orientation				De gauche à droite
Déplaçable				Oui

Format	Données	Événement	Autres	Toutes
Sur activation				
Avant insertion				
Après insertion				
Avant MAJ				
Après MAJ				
Si modification				
Sur annulation				
Sur suppression				
Avant suppression				
Après suppression				
Sur ouverture				
Sur chargement				
Sur redimensionnement				
Sur libération				
Sur fermeture				
Sur activé				
Sur désactivé				
Sur réception focus				
Sur perte focus				
Sur clic				
Sur double clic				
Sur souris appuyée				
Sur souris déplacée				
Sur souris relâchée				
Sur roulement de la souris				
Sur touche appuyée				
Sur touche relâchée				
Sur touche activée				
Aperçu des touches				Non
Sur erreur				
Sur filtre				
Sur filtre appliqué				
Sur minuterie				
Intervalle minuterie				0
Avant info-bulle				
Lors de la commande Cmd Enabled				
Lors de la commande Cmd Checked				
Lors de la commande Cmd Before Execute				
Lors de la commande Cmd Execute				
Lors de modifications des données				
Lors de modifications du jeu de données				
Lors de modifications du tableau croisé dynamique				
Lors de modifications de la sélection				
Lors de modifications de l'affichage				
Lors de la connexion				
Lors de la déconnexion				
Avant requête				
Sur requête				
Après mise en page				
Avant rendu				
Après rendu				
Après rendu final				

Format	Données	Événement	Autres	Toutes
Source				
Filtre				
Tri par				
Filtrage autorisé				Oui
Modif autorisée				Oui
Suppr autorisée				Oui
Ajout autorisé				Oui
Entrée données				Non
Type Recordset				Feuille de réponse dynamique
Verrouillage				Aucun
Extraction des paramètres par défaut				Oui

Format	Données	Événement	Autres	Toutes
Fen indépendante				Non
Fen modale				Non
Cycle				Tous les enregistrements
Barre de menus				
Barre outils				
Menu contextuel				Oui
Barre de menu contextuel				
Impression laser rapide				Oui
Fichier Aide				
Contexte Aide				0
Remarque				
Avec module				Oui
Autoriser modifications structure				Tous les modes

Propriétés du formulaire (dans le code) :



Saisissez le code¹ suivant (grâce au générateur de code de l'événement Sur clic des propriétés des boutons)

Format	Données	Événement	Autres	Toutes
Nom				ListeMembres1Prénom
Légende				liste des membres du prénom :
Image				(aucune)
Type image				Intégré
Transparent				Non
Par défaut				Non
Annuler				Non
Auto répéter				Non
Texte barre état				
Adresse lien hypertexte				
Sous-adresse lien hypertexte				
Visible				Oui
Afficher				Toujours
Activé				Oui
Arrêt tabulation				Oui
Index tabulation				0
Gauche				0,499cm
Haut				0,399cm
Largeur				4,286cm
Hauteur				0,714cm
Couleur texte				-2147483630
Police				Tahoma
Taille de police				8
Épaisseur police				Standard
Italique				Non
Souligné				Non
Barre de menu contextuel				
Texte d'Info-bulle				
Contexte Aide				0
Remarque				
Sur entrée				
Sur sortie				
Sur réception focus				
Sur perte focus				
Sur clic				Procédure événementielle
Sur double clic				
Sur souris appuyée				
Sur souris déplacée				
Sur souris relâchée				
Sur touche appuyée				
Sur touche relâchée				
Sur touche activée				
Sens lecture				Contexte

et

Format	Données	Événement	Autres	Toutes
Nom				ListeMembresPrénomCommenceVoyelle
Légende				liste des membres dont le prénom commence par une voyelle
Image				(aucune)
Type image				Intégré
Transparent				Non
Par défaut				Non
Annuler				Non
Auto répéter				Non
Texte barre état				
Adresse lien hypertexte				
Sous-adresse lien hypertexte				
Visible				Oui
Afficher				Toujours
Activé				Oui
Arrêt tabulation				Oui
Index tabulation				2
Gauche				0,499cm
Haut				1,6cm
Largeur				8,095cm
Hauteur				0,714cm
Couleur texte				-2147483630
Police				Tahoma
Taille de police				8
Épaisseur police				Standard
Italique				Non
Souligné				Non
Barre de menu contextuel				
Texte d'Info-bulle				
Contexte Aide				0
Remarque				
Sur entrée				
Sur sortie				
Sur réception focus				
Sur perte focus				
Sur clic				Procédure événementielle
Sur double clic				
Sur souris appuyée				
Sur souris déplacée				
Sur souris relâchée				
Sur touche appuyée				
Sur touche relâchée				
Sur touche activée				
Sens lecture				Contexte

¹ En cas de problème sur la déclaration d'un objet DAO (Database, QueryDef, Recordset, etc.) lors de l'exécution, vérifiez dans les références de Microsoft Visual Basic que Microsoft ActiveX Data Objects 2.7 Library (qui doit remplacer la version 2.1) précède Microsoft DAO 3.6 Object Library ; ensuite, préfixez la déclaration de l'objet par DAO. (i. e. DAO.Database, DAO.QueryDef, DAO.Recordset, etc.).

```

Option Compare Database
Option Explicit
' -----
' affiche les membres d'un prénom saisi par l'utilisateur
' -----
Private Sub ListeMembres1Prénom_Click()
' déclarations
Dim dbs As DAO.Database ' base de données courante
Dim frm As Form ' formulaire courant
Dim ctlPrénom As Control ' contrôle du prénom
Dim script_req As String ' script de la requête d'interrogation
Dim req As DAO.QueryDef ' requête d'interrogation
Const nom_req_tmp As String = "requête temporaire" ' nom de la requête d'interrogation temporaire
' initialisations
Set dbs = CurrentDb ' la base de données
Set frm = Me ' le formulaire
Set ctlPrénom = frm.Prénom ' = frm![Prénom] ' le prénom
' traitement
If ctlPrénom.Value <> "" Then ' il y a alors un prénom de saisi
    script_req = "SELECT [N°Membre] , NomMembre FROM Membres WHERE PrénomMembre = '" &
    ctlPrénom.Value & "' ;" ' script de la requête d'interrogation
    Set req = dbs.CreateQueryDef(nom_req_tmp, script_req) ' association du script de la
    requête à une requête temporaire
    DoCmd.OpenQuery req.Name ' exécution de la requête temporaire
    dbs.QueryDefs.Delete req.Name ' suppression de la requête temporaire
End If
End Sub
' -----
' affiche tous les membres dont le prénom commence par une voyelle
' -----
Private Sub ListeMembresPrénomCommenceVoyelle_Click()
' déclarations
Dim dbs As DAO.Database ' base de données courante
Dim frm As Form ' formulaire courant
Dim rstMembres As DAO.Recordset ' membres
' initialisations
Set dbs = CurrentDb ' la base de données
Set frm = Me ' le formulaire
' traitement
Set rstMembres = dbs.OpenRecordset("SELECT * FROM Membres WHERE Left(PrénomMembre,1) IN
('A','E','I','O','U','Y') ;", dbOpenDynaset) ' ouvre les membres
Do While Not rstMembres.EOF ' tant qu'il y a des membres
    MsgBox ("Membre de numéro" & Str(rstMembres.Fields(0)) & " de nom " &
    Trim(rstMembres.Fields(1)) & " de prénom " & Trim(rstMembres.Fields(2))) ' affichage du
    membre
    rstMembres.MoveNext ' membre suivant
Loop
rstMembres.Close ' ferme les membres
End Sub

```

Utilisez le formulaire (et testez ainsi le code) pour le jeu d'essai suivant :

- liste des membres du prénom pour : un prénom commun à plusieurs membres, un prénom d'un seul membre et un prénom d'aucun membre ;
- liste des membres dont le prénom commence par une voyelle.

22. Créez dans un premier temps un nouveau formulaire (en vous inspirant de celui-ci-après) composé d'une première zone de liste indépendante pour afficher ou non les champs de la table Associations, d'un premier groupe d'options pour appliquer un filtre, d'un second groupe d'options pour effectuer un tri et d'un bouton pour afficher le résultat de la recherche d'informations parmi les associations.

Écrivez dans un second temps (c'est du SQL dynamique) le code Visual Basic édition Application (en créant une procédure événementielle réagissant à un clic sur le bouton, et dont vous trouverez l'esquisse du code que vous devriez écrire ci-après) qui crée tout d'abord le script d'une requête d'interrogation (pour l'exemple : `SELECT [N°Association], [NomAssociation], [ObjetAssociation], [DateDécret] FROM Associations WHERE [N°Dédution] IS NULL ORDER BY [NomAssociation] ;`) qui est ensuite exécutée pour afficher le résultat de la recherche d'informations parmi les associations, en appliquant la projection, le filtre et le tri.

Pour cela, consultez l'aide depuis Access (et non depuis Visual Studio) et plus particulièrement les mots-clés (donnés en ordre alphabétique) : Control (lisez l'aide), CreateQueryDef (lisez plus particulièrement la syntaxe et l'exemple), CurrentDb (lisez plus particulièrement la première remarque), Database (lisez l'aide), Delete (lisez tous les paragraphes, mais pas l'exemple), Dim (lisez plus particulièrement le premier paragraphe et les exemples), DoCmd (lisez l'aide et consultez la liste des méthodes), For Each (lisez l'aide), Form (lisez l'aide et notez les différentes utilisations), ItemData (lisez attentivement l'aide : premier paragraphe, remarques et exemple), ItemsSelected (lisez plus particulièrement le premier paragraphe et les exemples), Me (lisez l'aide), OpenQuery (lisez plus particulièrement le premier paragraphe et l'exemple), **QueryDef** (lisez l'aide sachant que vous pouvez y accéder via le mot-clé QueryDefs), QueryDefs (lisez l'aide), Set (lisez l'aide sur l'instruction VBA), String (lisez l'aide concernant le type de données), Value (lisez l'aide et l'exemple, pour la propriété Value d'Access ou DAO), Variant (lisez l'aide).

Voici l'esquisse du code que vous devriez écrire :

```

déclarations des variables (base de données, formulaire, contrôles, QueryDef, etc.)
initialisations (base de données, formulaire, contrôles)
création du script de la requête à partir des choix de l'utilisateur
création d'une requête temporaire (QueryDef) contenant le script précédemment créé
exécution de la requête temporaire
suppression de la requête temporaire

```

23. Créez dans un premier temps un nouveau formulaire (identique à celui ci-dessous) composé d'une première zone de liste indépendante pour les membres (numéro affiché sur 0 cm, nom et prénom), d'un bouton et d'une seconde zone de liste indépendante pour les associations (numéro affiché sur 0 cm et nom).

les membres		les associations
Nom du membre	Prénom du membre	Nom de l'association
Onze	Marianne	Unicef
Quatorze	Virginie	Sécurité routière
Quatre	Céline	Restos du coeur
Quinze	Zoé	Amnesty International
Seize	Quentin	Ligue contre le cancer
Sept	William	
Six	Thierry	
Treize	Bénédicte	
Trois	Aurélien	

Créer une occurrence dans
Appartenir
entre le membre et l'association sélectionnés

Écrivez dans un second temps (c'est du SQL intégré) le code Visual Basic édition Application qui affecte le membre et l'association sélectionnés dans *Appartenir* (en vérifiant au préalable que l'occurrence n'existe pas déjà).

Pour cela, consultez l'aide depuis Access (et non depuis Visual Studio) et plus particulièrement les mots-clés (donnés en ordre alphabétique) : *AddNew* (lisez plus particulièrement l'aide et l'exemple), *Close* (lisez l'aide et l'exemple, pour la méthode *Close* de la bibliothèque DAO), *Column* (lisez plus particulièrement le premier paragraphe et la première remarque), *Control* (lisez l'aide), *CurrentDb* (lisez plus particulièrement la première remarque), *Database* (lisez l'aide), *dbOpenDynaset* (lisez le paragraphe correspondant disponible sur l'aide du mot-clé *OpenRecordset*), *Form* (lisez l'aide et notez les différentes utilisations), *ListIndex* (lisez plus particulièrement le premier paragraphe et la première remarque), *Me* (lisez l'aide), *MsgBox* (lisez attentivement l'aide et l'exemple), *OpenRecordset* (lisez plus particulièrement le premier paragraphe et la syntaxe, ainsi que l'exemple), *RecordCount* (lisez plus particulièrement le premier paragraphe et l'exemple), **Recordset** (lisez attentivement l'aide et les exemples), *Update* (lisez plus particulièrement le premier paragraphe et l'exemple).

Voici l'esquisse du code que vous devriez écrire :

```

déclarations des variables (base de données, formulaire, recordset, contrôle)
initialisations (base de données, formulaire, contrôle)
Si un membre et une association ont été sélectionnés (dans leurs zones de liste)
Alors Créer un recordset = requête recherchant (membre,association) dans Appartenir
    Si (membre,association) n'est pas dans le recordset
    Alors Créer un recordset = table Appartenir (pour la mettre à jour)
        Ajouter un nouvel enregistrement au recordset
        Affecter les valeurs au recordset
        Enregistrer les modifications du recordset
        Fermer le recordset basé sur la table Appartenir
        Informer l'utilisateur que (membre,association) a été ajouté
    Sinon Informer l'utilisateur que (membre,association) existait déjà
    Fermer le recordset basé sur la requête recherchant dans Appartenir
Sinon Informer l'utilisateur qu'il doit sélectionner un membre et une association

```