

Objectifs

Malgré la puissance des outils interactifs comme ACCESS ou FOXPRO, de nombreuses actions ne sont pas accessibles par les menus. Il faut souvent passer par la programmation. Chacun de ces outils propose son langage de développement. ACCESS par exemple peut être programmé en VBA (*Visual Basic for Applications*) comme toutes les applications *Microsoft*. Ce TP a pour objectif de vous donner un bref aperçu du fonctionnement de VBA sous ACCESS.

Pensez à Activer le contenu au lancement Access, sinon VBA ne fonctionne pas

Exercice 1 : Calcul du nombre de notes inférieures à 10

L'objectif de cet exercice consiste à afficher dans l'état `etudiants` listant les notes par étudiant, le nombre de notes inférieures à 10 pour chaque étudiant. Dans les états tout est rythmé par les entêtes et pieds d'état, les entêtes et pieds de page, les entête et pied de groupe et les lignes de détail. Pour traiter notre problème, il faut d'abord identifier quelles sont les procédures de base nécessaires à l'application, puis déterminer le moment de leur activation.

Dans notre cas il faudra écrire 3 procédures, l'une qui initialise un compteur à chaque nouvel étudiant donc procédure réalisée à chaque `entête de groupe`, la seconde qui teste chaque note et la comptabilise éventuellement, procédure réalisée à chaque `détail` et une dernière qui affiche le résultat, procédure réalisée à chaque `pied de groupe`.

Q1. Créez une requête effectuant la jointure sur les 3 tables. Créez un état sur cette requête affichant les notes regroupées par étudiant.

Q2. Passez en mode construction et ajoutez un pied de groupe `num-etu` (Clic droit sur `Entete de Groupe` puis `Trier et Grouper` puis `+` et choisir `Avec section de Pied de Page`).

Q3. Ajoutez dans le pied de groupe une zone de texte avec comme libellé "**nombre de notes inférieures à 10**". Elle servira à afficher le résultat.

Q4. Nommez la partie qui permet l'affichage de la valeur **Compte** (`propriété > catégorie autre > nom`).

Renommer les champs d'Access permet d'être plus clair. Il faudra bien sûr que ce soit cohérent dans le code.

Q5. Sélectionnez l'entête de groupe. Il faut maintenant rentrer du code VBA associé à chaque entête (`Propriété catégorie événements / au formatage` puis clic sur `...`). Le code placé ici sera appelé à chaque entête de groupe (donc de chaque étudiant).

Q6. Tapez le code ci-dessous autour de la procédure créée qui doit avoir pour nom **EntêteGroupe0_Format** :

```
Private NbNotes as integer
Private sub EntêteGroupe0_Format(Cancel as Integer, FormatCount as Integer)
    NbNotes=0
End Sub
```

Q7. Fermez l'éditeur pour revenir au mode conception d'état.

Q8. De la même manière créez le code associé aux sections **Détail** et **Pied de Groupe** :

```
Private Sub Detail_Format(Cancel as Integer, FormatCount as Integer)
    If (Note <10) then NbNotes = NbNotes+1
End Sub
```

```
Private Sub PiedGroupe0_Format(Cancel as Integer, FormatCount as Integer)
    Compte = NbNotes
End Sub
```

Q9. Quittez l'éditeur et visualisez l'aperçu via `Aperçu avant impression`.

Exercice 2 : Calcul de la moyenne coefficientée

Q1. Sur le même principe que précédemment, faites en sorte que la moyenne coefficientée de chaque étudiant soit calculée puis affichée à chaque pied de groupe.

Exercice 3 : Affichage des notes en colonne¹.

Avec des outils comme Access il est très facile de créer des états qui mettent chaque enregistrement d'une table ou une requête sur une ligne. Malheureusement il est parfois plus judicieux de mettre les informations en colonne, mais Access n'offre rien pour cela. Il s'agit ici de permettre d'afficher les notes d'un étudiant non pas en ligne comme précédemment, mais en colonne afin de gagner la place perdue sur la feuille. Nous considérerons qu'il n'y a pas plus de 7 notes par étudiant. Voici quelques points pour vous aider :

- Utilisez un tableau de 7 réels (type Double) rempli par indice croissant. La déclaration d'un tel tableau se fait par **Private Ligne(7) as Double** et l'accès par **ligne(3)=...**
- Concevoir 7 zones de texte en pied de groupe qui recevront au plus les 7 valeurs du tableau
- Laisser une zone de texte note dans la section détail mais forcer la propriété visible de cette section à **non** afin de ne pas afficher les notes en ligne.

Exercice 4 : Rechercher et afficher un enregistrement

En général, un formulaire est associé à une table. Si on construit un formulaire de type `Colonne Simple` sur la table `Etudiant`, tous les champs affichés sont ceux directement en liaison avec la table `Etudiants`. Taper une valeur dans le champs `numetu` ne recherche pas l'étudiant concerné, mais modifie la clé de cet enregistrement !.

Voyons comment s'y prendre pour permettre la recherche d'un étudiant à partir de son nom.

Q1. Créez un nouveau formulaire `Etudiants` de type `Colonne Simple`.

Q2. Agrandissez le et ajoutez une zone de liste déroulante à ce formulaire.

Q3. Avec l'assistant complétez pour rechercher les valeurs `numetu,nom,prenom` dans la table `etudiants` et mémoriser pour un usage ultérieur.

Q4. Affichez maintenant les propriétés de ce composant. Sélectionner :

1. Access offre plusieurs moyens pour déboguer du VBA. L'un des moyens les plus simples consiste à afficher des fenêtres modales avec un message. Par exemple `MsgBox (sumNotes&" "&sumCoeffs)`.
`Debug.print()` quant à lui affiche dans la fenêtre de debug affichable avec un `Cntl G` en mode VBA

Q5. Dans le code de la procédure associée tapez (vérifiez et changez éventuellement les noms dans le code suivant) :

```
numetu.SetFocus  
Call DoCmd.FindRecord(Modifiable8.Value)
```

Testez maintenant ce nouveau composant vous permettant de rechercher une fiche à partir du nom et du prénom.