

# TRAVAUX PRATIQUES SGBD

## INITIATION À ACCESS

### PARTIE 1

## I. Introduction

### I.1. Objectif du TP

- Savoir créer une base simple, comportant quelques tables, des requêtes et un formulaire, avec puis sans l'aide des assistants.
- Comprendre le lien entre la partie théorique du cours et ce que vous faites avec Access

Les conventions typographiques suivantes vous indiquent que :

Ceci est un commentaire, une explication.

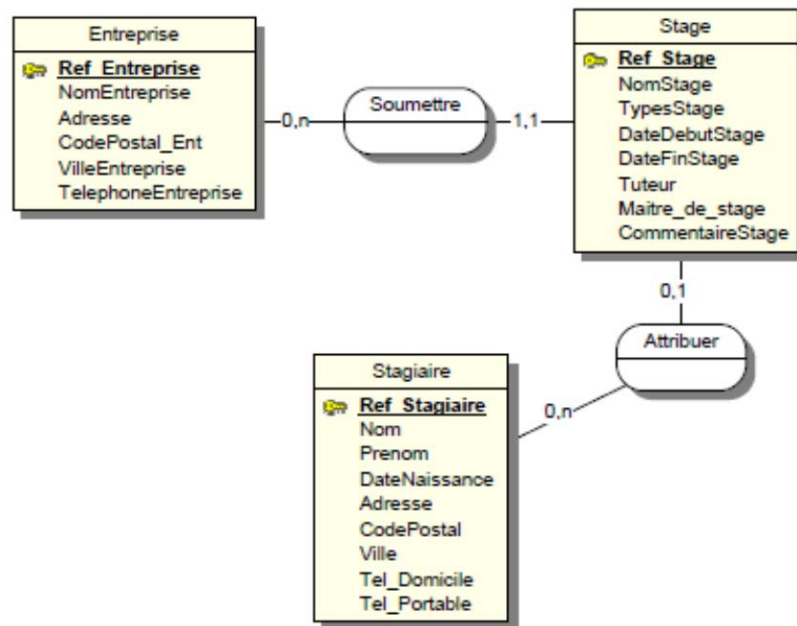
➤ Ceci est une action que vous devez effectuer

### I.2. Enoncé du besoin

Le service de Stages de L'école d'ingénieurs d'ISIS vous confie le soin de créer une base de données dans le but de gérer les stagiaires. Voici un brouillon de la fiche désirée

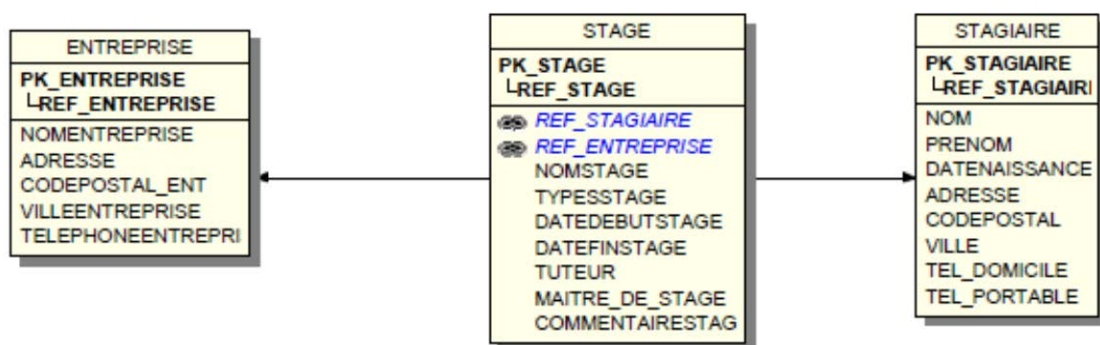
FICHE STAGIAIRE
<b>Code Stagiaire:</b> 124 (automatique) <b>Nom :</b> Dupont <b>Prénom :</b> Nadine <b>Date de naissance :</b> 03/03/1984
<b>Adresse :</b> 18 rue des bons enfants <b>Code postal :</b> 81000 <b>Ville:</b> Albi <b>Téléphone Domicile:</b> 05 63 49 33 22 <b>Téléphone portable:</b> 06 63 41 19 67
<b>Sujet du stage :</b> Analyse de la gestion des flux <b>Type de stage :</b> Stage d'ingénieur <b>Début de Stage :</b> 9/05/2006 <b>Fin du Stage :</b> 9/05/2006 <b>Tuteur :</b> Mr. Lamine
<b>Entreprise :</b> Clinique Toulouse-Lautrec <b>Téléphone :</b> 05 63 48 45 45 <b>Adresse :</b> Val de Caussels <b>Code postal :</b> 81000 <b>Ville:</b> Albi <b>Correspondant :</b> Mr. Thary
<b>Observations</b>  Connaissances requises xxxxxxxxxxxx Rémunération ???
<b>Fiche créée le:</b> 28/02/06 <b>Par :</b> Lamine <b>Fiche modifiée le:</b> xxxxxx

Dans une première analyse on s'aperçoit qu'il y a trois types de données : celles qui concernent les stagiaires individuellement, celles qui concernent les stages et celles qui concernent les entreprises. Le modèle conceptuel suivant représente l'organisation de ces données :



**Figure 1 : Modèle conceptuel**

On en déduit le modèle relationnel suivant :



**Figure 2 : Modèle relationnel de données**

### I.3. Travail demandé

L'exercice consiste donc à :

1. Créer cette base, qui comprend 3 tables (une table Stagiaire, qui viendra "piocher" dans la table Stage, la table stage fera référence à des entreprises existantes dans la table Entreprise) ;
2. Créer des formulaires, pour aider à saisir les informations ;
3. Mettre en place de différentes requêtes, selon le besoin du service de stage ou du département ;
4. Création de différents états, pour imprimer le résultat des requêtes.
5. Créer un menu qui se lancera automatiquement au démarrage et qui permettra de lancer le formulaire des stages et celui des stagiaires.

## II. Démarrage / Création d'une base

### II.1. Présentation

Access fait partie du lot Office, qui dispose d'une interface groupée, en haut du menu démarrer, ou selon les installations dans le menu Applications bureautiques.

La boîte de démarrage par défaut permet d'ouvrir une base de données existante dans le répertoire par défaut ou de créer une nouvelle base de données soit manuellement soit à l'aide de l'assistant de création.

Pour Access, **une base = un fichier d'extension mdb**. Tous les objets de la base sont créés dans ce même fichier, qui peut être déplacé, copié renommé sans perte d'informations.

Une fois la base créée ou ouverte, Access vous présente la fenêtre "base de données": 6 onglets permettent de lister les 6 types d'objets qui composent une application Access.

### II.2. Aide

L'aide en ligne est extrêmement pratique et utile ! N'hésitez pas à l'utiliser si un point vous semble obscur.

Taper <F1> pour ouvrir l'aide. Par exemple, dans l'onglet Sommaire de l'aide, cliquez sur Introduction visuelle à Microsoft Access pour lancer un tutorial.

- Double cliquer sur l'icône d'Access (clef) pour démarrer l'application Access
- Créer une nouvelle BD vide en choisissant explicitement un répertoire de travail.
- Saisissez le nom de BD « SGBD\_Stages ».

Apparaît alors la fenêtre BD avec 3 boutons et 6 onglets.

## III. Les Tables

Dans cette partie, vous allez apprendre à :

- Créer une nouvelle table
- Entrer des données dans une table existante
- Indiquer à Access quelle est la clé primaire d'une table
- Indiquer à Access des contraintes sur les données saisies
- Créer et modifier une requête

### III.1. La fenêtre création / modification de table.

Dans l'onglet tables, cliquer sur "Nouvelle", puis choisir le mode création.

Dans ce mode de "création", la liste des champs (ou colonnes, ou attributs) de la table est affichée, vous pouvez saisir ou modifier pour chaque champ (ou colonne, ou attribut, notions équivalentes) les caractéristiques suivantes :

- Nom (64 caractères maximum, pas de caractères !, . [ ], `)

- Type de donnée contenue dans cette colonne (en modèle relationnel, on pourrait dire domaine), les types suivants sont reconnus par Access:

Texte	255 caractères maxi.
Numérique	octet, entier(2), entier long(4), simple(4), double(8)
Monétaire	8 octets, précision 15.4, pas d'arrondis
Numérotation automatique	4 octets (correspond à un entier long)
Date/heure	8 octets
Booléen (oui/non)	1 bit
Lien OLE	limité par la taille du disque (1Go)
Mémo	64000 caractères

- **Créer un attribut Ref\_Entreprise, de type "Numéro automatique"**
  - Le type numéro automatique indique à Access de générer un nombre entier de façon séquentielle au fur et à mesure de la création de lignes dans cette table.
- **Créer les attributs NomEntreprise, Adresse, CodePostal, VilleEntreprise, TelephoneEntreprise de type texte ( cf. Tableau 1)**
- **Ne pas s'occuper pour le moment des autres caractéristiques des champs, telles que:**
  - la taille maximum du champ,
  - le format d'affichage,
  - le masque de saisie,
  - les règles de validité,
  - la valeur par défaut.
  -

## III.2. Définition d'index et de clé primaire.

- Il est préférable que toute table contienne un identifiant ou clé. Une clé est constituée d'un ou plusieurs champs de la table.
- Pour définir une clé, sélectionner la ou les lignes correspondant au(x) champ(s) composant la clé, puis cliquer sur le bouton « clé » ou cocher l'option :Édition/Clé primaire.
- **Indiquer comme clé le champ Ref\_Entreprise**
- **Fermer la fenêtre de création et indiquer le nom de la table "Entreprise"**
- L'enregistrement de la structure de la table est nécessaire avant toute utilisation de cette table, que ce soit une visualisation, une saisie, une modification de données ou la définition de **relations** entre les tables.

	Nom des tables	Les champs	Type	Commentaires
1	<b>Entreprise</b>	Ref_Entreprise NomEntreprise Adresse CodePostal_Ent VilleEntreprise TelephoneEntreprise	NuméroAuto Texte – 30 Texte – 80 Texte – 10 Texte – 40 Texte - 20	Clé primaire

**Tableau 1 : Table Entreprise**

## III.3. Le mode feuille de données.

Pour modifier les données de cette table dans le mode feuille de données :

- **Sélectionnez la table Entreprise**
- **Cliquez sur le bouton "Ouvrir".**
- **Saisir les entreprises suivantes :**

NomEntreprise	Adresse	CodePostal_Ent	VilleEntreprise	TelephoneEntreprise
Pierre Fabre	14, rue albert	81000	ALBI	O563626362
CGX	46 rue edit	81100	Castres	O554535678

➤ **Refermez la table, puis cliquer sur le bouton "Modifier"**

Une fois la table ouverte dans l'un ou l'autre mode (par le bouton Modifier ou Ouvrir), vous pouvez basculer du mode **création** au mode **feuille de données** et inversement par le bouton situé tout à fait à gauche de la barre d'outils (icône représentant une table ou des outils du genre règle et équerre).

Le mode feuille de données permet de visualiser, d'ajouter, de modifier ou de supprimer des informations dans la table.

Dans ce mode, vous pouvez modifier (= éditer) les données, avec les fonctions habituelles d'un outil bureautique (couper, copier, coller, supprimer une ligne, rechercher, remplacer...)

Vous pouvez **modifier la présentation** (largeur des colonnes...).

Vous pouvez **trier** les données sur les valeurs de n'importe quelle colonne.

➤ **Sélectionnez la table Entreprise et cliquez sur le bouton "Ouvrir"**

NomEntreprise	Adresse	CodePostal_Ent	VilleEntreprise	TelephoneEntreprise
Sun	16 rue Morice	75000	Paris	O147301812
EquaSys	2 place Soult	73000	Chambery	O433044455
Cvital	99 rue Cantepeau	34090	Montpellier	O443056712
Almerys	Avenue Pompidou	75000	Paris	O121810374

➤ **Triez sur le nom par ordre alphabétique, puis en ordre inverse, faire de même pour trier sur le Code Postal, puis sur la ville**

### III.4. Déclaration des clés étrangères ("relations")

Une déclaration de clé étrangère permet de signifier à Access la présence d'une association entre deux tables, en liant une clé étrangère d'une table dépendante à une clé primaire d'une table maîtresse. Access nomme ces déclaration de clés étrangères "relations", ne pas confondre ce terme avec le terme relation signifiant table, utilisé globalement pour définir un modèle relationnel.

Les déclarations de clés étrangères à Access ne sont pas obligatoires, mais elles permettent de se faciliter la tâche, puisque Access générera automatiquement les conditions de jointure basées sur les associations déclarées à chaque fois qu'une requête fera intervenir les deux tables associées.

➤ **Créez les tables Stage et Stagiaire sans indiquer les associations (relations) qui les lient.**

	Nom des tables	Les champs	Type	Commentaires
2	<b>Stage</b>	ReferenceStage NomStage TypeStage DateDebutStage DateFinStage Tuteur Correspondant CommentaireStage ReferenceEntreprise ReferenceStagiaire	NuméroAuto Texte – 80 Texte – 80 Date/Heure Date/Heure Texte – 20 Texte – 20 Texte – 255 Numérique Numérique	Clé primaire       pour les annotations permettra de faire le lien avec la table Entreprise permettra de faire le lien avec la table stagiaire
3	<b>Stagiaire</b>	ReferenceStagiaire NomStagiaire Prenom DateNaissance Adresse CodePostal_Stag VilleStagiaire Tel_Domicile Tel_Portable	NuméroAuto Texte - 30 Texte - 30 Date/Heure Texte - 80 Texte - 10 Texte - 40 Texte - 20 Texte - 20	Clé primaire

**Tableau 2 : Table Stage et Stagiaire**

La fenêtre *Relations* permettant de définir ou de modifier des relations est accessible par la commande Outils/Relations du menu de la fenêtre *Base de données* ou à l'aide du bouton **Relations** de la même fenêtre.

La commande « Relations / Ajouter une table » fait apparaître une fenêtre permettant de choisir parmi les tables et les requêtes existantes celles que l'on veut lier. Cette fenêtre est également accessible à l'aide du bouton **Ajouter une table**

➤ **Ouvrir la fenêtre Relations, Ajouter les tables Entreprise, Stage et Stagiaire**

- Une association met généralement en correspondance une **table primaire** dont le champ lié doit être **indexé sans doublons** (en général une clé primaire) et une **table liée** pour laquelle aucun index n'est nécessaire pour le champ lié.
- Pour créer une relation : Cliquer sur le champ lié de la table primaire et glisser jusque sur le champ correspondant de la table liée.
  - **Lier la table Entreprise à la table Stage par l'attribut ReferenceEntreprise**
  - **Lier la table Stage à la table Stagiaire par l'attribut ReferenceStagiaire**

### III.5.Modification/Suppression d'associations

Pour modifier ou supprimer une relation existante, il faut d'abord sélectionner cette relation en effectuant un double-clic sur le lien symbolisant cette relation.

- On peut alors sélectionner la commande Relations/Modifier une relation ou la commande Edition/Supprimer

- Il est également possible d'accéder directement à ces commandes en cliquant sur le lien symbolisant la relation à l'aide du bouton **droit** de la souris.

### III.6. Gestion de l'intégrité référentielle dans des associations

- Si le champ de la table primaire à l'origine du lien est indexé sans doublons, on peut cocher la case intégrité référentielle de la fenêtre de modification des relations.
- Cela permet de définir une association de type un à plusieurs (ou un à un si le champ de la table liée est également indexé sans doublons), et surtout d'empêcher toute suppression d'un enregistrement de la table primaire qui serait en correspondance avec un enregistrement de la table associée.
- **De plus les enregistrements de la table primaire doivent exister avant d'être références par la table associée.**
  - **Modifier le lien liant la table Entreprise à la table Stage, imposer l'intégrité référentielle avec mise à jour et effacement en cascade**
  - **Modifier le lien liant la table Stagiaire à la table Stage, imposer l'intégrité référentielle avec mise à jour et effacement en cascade**
  - **À partir de la fenêtre Tables, insérer les données suivantes :**

Ref	Nom	Prenom	Date Naissance	Adresse	CodePostal	Ville
1	BUSOLIN	Winifried	02/01/1983	27, rue bastat	81000	Albi
2	QUEVEDO	Tamer	03/08/1987	6 rue editat	75000	Paris
3	DORLIGH	Layla	08/07/1986	1 rue Mozart	81000	Albi
4	PUIBUSQUE	Querrien	23/09/1984	2 place etoile	73000	Chambery

NomStage	Type Stage	Date DebutStage	Date FinStage	Tuteur	Maitre de stage	Reference Entreprise	Reference Stagiaire
Mise au point d'IHM	ouvrier	01/02/2006	28/02/2006	Bajja	ALLOSA	6	2
traitement automatique du langage naturel	technicien	01/05/2008	31/07/2008	ZAMENGO	PALCHINE	1	3
Conception d'un système de gestion de fluides	assistant	01/05/2008	31/07/2008	Bajja	PIOVESAN	6	1
Fouille de données	ingénieur	01/02/2009	30/06/2009	VAZZOLER	VAGLIENTI	2	1
Analyse des spécificités du MAD	technicien	01/05/2007	31/07/2007	ROSADA	WIOSKA	3	2

- **Essayer de saisir un Stage avec un numéro d'Entreprise inexistant, que constatez-vous ?**
- **Saisir un nouveau stagiaire et un nouveau stage. Affecter le nouveau stagiaire à ce dernier.**
- **Supprimer ce stagiaire, puis vérifier que le stage correspondant a été supprimé.**

### III.7. Liste de valeurs

Afin de faciliter la saisie des valeurs d'un attribut qui appartiennent à une liste prédéfinie, Access permet de présenter sous forme d'une liste déroulante les valeurs possibles d'un attribut.

- **Ouvrir la table Stage en mode « création »**
- **Sélectionner l'attribut « TypeStage », aller sur l'onglet "liste de choix"**
- **Sélectionner « zone de liste déroulante » comme le type de « contrôle » pour ce champ**
- **Indiquer « liste de valeurs » comme source**

- **Créer maintenant le contenu de la liste de valeurs en séparant les valeurs possibles par des « ; », sachant qu'il y a quatre types de stage :**
  - Ouvrier ;Technicien ; Assistant d'ingénieur ; travaux de fin d'études,
- **Enregistrer la table et passer en mode "feuille de données" pour visualiser le résultat**

## IV. Les requêtes

Une requête est un ordre stocké, qui génère une table résultat à partir d'une ou plusieurs tables de départ.

Access permet de définir graphiquement et d'exécuter des requêtes, qui sont directement traduites en langage SQL avant exécution.

On peut ainsi travailler sur différents types de requêtes : Sélection (type par défaut), analyse croisée, création de table, Mise à jour, Ajout, Suppression.

Chaque requête peut apparaître en mode **création**, **feuille de données** ou **SQL**.

### IV.1. Les requêtes sélection

Les requêtes *sélection* permettent de faire des **extractions** de données à partir de tables, mais également à partir du résultat d'autres requêtes (puisque le résultat d'une requête est une table).

On accède à la création d'une requête en cliquant sur un des boutons **Nouveau** ou **Modifier** de la fenêtre Base de Données si l'onglet « **Requêtes** » est sélectionné.

- **Créer une requête donnant la liste des Entreprise triée par code postal :**
- **Requête/Ajouter une table, ou bouton Ajouter une table, choisir la table Entreprise**
- **Sélectionner l'attribut CodePostal\_Ent, le faire glisser dans la case "champ" de la première colonne**
- **Sélectionner l'attribut NomEntreprise, le faire glisser dans la case "champ" de la 2ème colonne**
- **Choisir Tri Ascendant dans la colonne CP**
- **Passer en mode "feuille de données" pour visualiser le résultat**

- Pour chaque champ sélectionné, il est possible de le faire afficher ou non (peut être utile pour trier sur un champ dont on ne souhaite pas afficher la valeur)
- On peut également exprimer des critères de sélection:
  - **Noter dans la table Entreprise un code postal (pour validité de l'exercice suivant)**
  - **Revenir en mode "modification" et écrire = (le code postal noté précédemment) dans le champ critères de la colonne Cp.**
  - **Passer en mode "feuille de données" pour visualiser le résultat**
  - **Passer en mode SQL pour visualiser le SQL automatiquement généré par Access.**

### IV.2. Requêtes paramétrées.

- Pour définir une requête paramétrée, c'est-à-dire une requête pour laquelle un ou plusieurs critères de sélection dépendent d'une valeur saisie en début d'exécution, il faut placer entre [**crochets droits**] dans la case critère du champ à comparer un texte d'invite à la saisie.



- Lors de son exécution, la requête fera apparaître une fenêtre de saisie pour chacun des paramètres ainsi définis.
  - **Dans la zone des critères, remplacer = 81000 par = [Donner la valeur du Code Postal recherché], puis exécuter la requête**
  - **Entrer une valeur quand Access vous le demande**

### IV.3. Des expressions dans une requête : regroupements, comptage

Certaines colonnes d'une requête peuvent être des expressions calculées à partir des divers champs contenus dans cette requête. Ces expressions peuvent soit être saisies directement par le concepteur dans la case *champ* de la colonne, soit générée par un outil appelé **générateur d'expression**.

- Si l'on veut que ces colonnes aient un en-tête particulier, la syntaxe à utiliser est la suivante : *Nouvelle en-tête : nom du champ*
  - **Créer une nouvelle requête sur la table Entreprise, avec les attributs Ville et Nom**
  - **Cliquer sur le signe Sigma, signifiant "Opérations"**
  - **Indiquer « regroupement » dans la colonne VilleEntreprise, « comptage » dans la colonne NomEntreprise**
  - **Passer en mode "feuille de données" pour visualiser le résultat: on a la liste du nombre de clients par ville**

Le **générateur d'expression**, accessible en cliquant sur les 3 petits points à droite d'un champ, permet d'accéder directement à tout objet de la base de données (champs, tables, requêtes, ...) ainsi qu'à des constantes prédéfinies (vrai, faux, null,...), à des opérateurs de différents types (logique, arithmétique, comparaison) et aux nombreuses fonctions intégrées dans le logiciel.

### IV.4. Fenêtre feuille de données d'une requête sélection

La fenêtre feuille de données d'une requête sélection est équivalente à une fenêtre en mode feuille de données d'une table. On y trouve les mêmes options de la barre de menu.

Des modifications d'informations peuvent éventuellement être effectuées, sous réserve qu'elles ne violent pas les contraintes d'intégrité et qu'elles soient valides conceptuellement (par exemple, on ne peut pas modifier le nombre de clients ayant un code postal donné).

Ces modifications sont automatiquement répercutées dans la ou les tables contenant les champs dont le contenu a été modifié.

## V. Formulaires

Un formulaire est une fenêtre qui permet notamment de visualiser, modifier, ajouter des informations provenant d'une table ou d'une requête.

Un formulaire ne peut être associé à plus d'une table ou d'une requête. Si l'on désire travailler dans un même formulaire sur des informations provenant de plusieurs tables ou requêtes, il faut créer des sous-formulaires que l'on insère ensuite dans le formulaire principal, chaque formulaire étant associé à au plus une table ou requête.

Un formulaire peut n'être associé à aucune table. Un tel formulaire peut servir de menu ou d'écran d'aide en ligne.

D'une façon générale, un formulaire contient des **contrôles** au sens *Visual Basic* du terme. Ainsi, les champs de la table associée peuvent apparaître sous la forme de **zones de texte** (text box) et les légendes correspondantes sous forme d'**étiquettes** (label). On peut également inclure à un formulaire toute sorte de contrôle : boutons bascule, boutons de commande, boutons d'options, cases à cocher, listes déroulantes modifiables ou non, ...

Un formulaire peut contenir plusieurs sections :

- La section détail qui est la section par défaut du formulaire
- Les sections en-tête de formulaire et pied de formulaires (optionnelles)
- Les sections en-tête de page et pied de page (optionnelles)

## V.1. Création d'un formulaire simple

### V.1.1. Avec l'assistant

- **Créer un formulaire basé sur la table Entreprise, en utilisant l'assistant de création de formulaire.**
- **Y mettre tous les attributs de la table**
- **Choisir le format "justifié", qui place les champs au mieux, puis un style (sobre !)**
- **Lui donner le nom "frmEntreprise"**
- **Ouvrir le formulaire frmEntreprise en mode saisie de données**
- **L'ouvrir et constater son bon fonctionnement: déplacement de ligne en ligne par la barre de boutons de navigation en particulier.**
- **Saisir des nouvelles entreprises.**

Il est important de choisir une règle de nommage des objets et de s'y tenir; il devient sinon très vite difficile de s'y retrouver parmi les nombreux objets créés.

Une norme possible est de préfixer les noms d'objets par leur type abrégé: req, frm, tbl...

### V.1.2. En mode création

- **Sélectionner le formulaire créé, et cliquer sur Modifier**

Outre la fenêtre de création du formulaire, l'écran fait apparaître les fenêtres :

- Boîte à outils qui permet de sélectionner un contrôle à insérer dans le formulaire
- Liste de champs qui permet d'insérer un champ et sa légende dans le formulaire par glissement de la souris.
- « Propriétés » qui permet de visualiser et de modifier les différentes propriétés liées à l'objet sélectionné
  - **Modifier le contenu de la propriété « légende » de l'étiquette « FrmEntreprise » par « Fiche Entreprise »**
  - **Modifier la taille et la fonte de la police correspondante à cette étiquette**
  - **Modifier l'étiquette VilleEntrepise par « Ville »**
  - **Masquer les contrôles associés à Ref\_Entrepise**

## V.2. Intégration de sous-formulaires.

L'utilisation des sous-formulaires permet de rassembler sur un même formulaire, des données provenant de différentes tables ou requêtes.

Pour créer un formulaire principal contenant un sous-formulaire :

- Créer le formulaire principal
- Créer le formulaire secondaire
- Insérer le sous-formulaire dans le formulaire principal (en glissant avec la souris)
- Définir éventuellement les champs **père** et **fils** (à l'aide des propriétés du sous-formulaire), si Access ne l'a pas déduit des associations déclarées entre les tables.

- **Créer un formulaire basé sur la table Stage, en utilisant l'assistant. de création de formulaire**
- **Y mettre tous les attributs de la table**
- **Choisir le format "tabulaire", puis un style (sobre !)**
- **Lui donner le nom "frmStage"**
- **Fermer en sauvegardant**
- **Créer un formulaire principal basé sur la table Stagiaire, en utilisant l'assistant. de création de formulaire**
- **Y mettre tous les attributs de la table**
- **Choisir le format "justifié", qui place les champs au mieux, puis un style (sobre !)**
- **Lui donner le nom "frmStagiaire"**
- **Faire glisser le formulaire frmStage (pris dans l'onglet formulaires) SUR le formulaire frmStagiaire ouvert**
- **Fermer en sauvegardant puis rouvrir en mode saisie de données**

Vous pouvez alors naviguer sur la table Stagiaire (dite table maître), pour chaque Stagiaire affiché vous verrez les stages correspondants de la table Stage (dite table esclave).

Access a automatiquement lié la navigation dans le formulaire commande à l'affichage dans le formulaire ligne, car vous lui aviez indiqué qu'il existait une association de type 1 à N entre ces deux tables.

Ce système peut fonctionner à plusieurs niveaux :

- **Ouvrir le formulaire frmEntreprise en mode Modification/Création**
- **Faire glisser le formulaire frmStage (pris dans l'onglet formulaires) sur le formulaire frmEntreprise ouvert**
- **Sauvegarder le formulaire**
- **Ouvrir le formulaire frmEntreprise en mode saisie de données**
- **Saisir des entreprises supplémentaires, manipuler les boutons de déplacement, supprimer une ligne**
- **Pour une Entreprise donnée, lui ajouter un Stage, puis affecter un Stagiaire pour ce stage**

---

### V.3. Aide au choix : les listes modifiables

Il arrive fréquemment qu'on doive contrôler que la valeur d'un champ saisi soit une valeur présente dans une autre table (clé étrangère).

Plutôt que de contrôler "a posteriori", il est bien entendu préférable de proposer la liste des valeurs possibles à l'utilisateur; le contrôle "liste modifiable" permet cela.

Nous allons utiliser cette possibilité afin de "normaliser" en partie la table "Entreprise", en supprimant l'attribut VilleEntreprise et en créant une table Ville comportant les attributs CodePostal\_Ent et VilleEntreprise.

- **Créer une table Ville comportant les attributs CodePostal et Ville**
- **La remplir avec quelques lignes**
- **Supprimer l'attribut VilleEntreprise. de la table Entreprise**

### **V.3.1. Définition d'une liste modifiable sur l'attribut lui-même**

Access permet de définir la présentation sous forme de liste modifiable sur la définition de l'attribut lui-même.

- **Sélectionner l'attribut CodePostal\_Ent, aller sur l'onglet "liste de choix"**
- **Indiquer que l'origine de la source est « table/requête » et que le contenu est « Ville »**
- **Indiquer de présenter deux colonnes (CodePostal et Ville)**
- **Indiquer que la colonne liée est la première (c'est l'attribut Cp qu'il faut récupérer)**
- **Passer en mode feuille de données et vérifier le bon fonctionnement**

### **V.3.2. Indépendance logique / Définition d'une liste modifiable sur un formulaire**

- **Ouvrir le formulaire Entreprise créé tout à l'heure en mode saisie**

Le champ Ville\_Ent n'est plus affiché, puisqu'il ne fait plus partie de la table. Nous allons reprendre le formulaire pour que le champ Ville soit affiché.

Les systèmes de gestion de bases de données relationnels offrent l'avantage de pouvoir facilement rendre un programme indépendant des modifications de la structure logique des tables sous-jacentes.

Ici, la table Entreprise a été scindée en deux tables: Entreprise et Ville, de façon à éliminer une redondance; **Il faut penser à définir la contrainte d'intégrité référentielle associée**

Pour ne pas avoir à modifier de façon trop lourde le formulaire, nous allons créer une requête dont le résultat donne l'exact équivalent de l'ancienne table Entreprise, et que nous utiliserons comme source de données pour le formulaire frmEntreprise.

- **Créer une requête nomme EntrepriseVille, donnant tous les attributs de la nouvelle table Entreprise + la Ville par association avec la table Ville (jointure sur le Code postal)**
- **Ouvrir le résultat de cette requête: on voit la table Entreprise comme elle était avant découpage.**
- **À partir de la fenêtre « propriétés », indiquer au formulaire de prendre ses données 'source de contrôle) dans la requête EntrepriseVille et non plus dans la table Entreprise**

On mesure ici l'un des avantages des SGBD, l'indépendance vis-à-vis des modifications de structure; il nous a suffi de créer une vue identique à l'ancienne table "Entreprise" (qui incluait la ville) pour que notre formulaire continue à fonctionner correctement.

- **A partir des formulaires conçus, saisir les données suivantes :**

Ref Stagiaire	Nom	Prenom	Date Naissance	Adresse	CodePostal	Ville
5	BORE	Teddy	14/06/1989	9 rue des bons enfants	75000	Paris
6	ROMIGUIERES	Winnie	28/12/1988	Avenue foch	81000	Albi
7	SKUTNIK	Léa	21/04/1990	100 route de ganges	34000	Montpellier
8	DENIA	Quentin	08/03/1989	2 rur occitanie	34000	Montpellier

NomStage	Type Stage	Date DebutStage	Date FinStage	Tuteur	Maitre de stage	Reference Entreprise	Reference Stagiaire
Conception d'un système de télésurveillance	ingénieur	01/02/2005	31/07/2005	GRANIZO	FOGLIENI	4	4
Mise en place d'un workflow	ingénieur	01/02/2006	31/08/2006	RUZAFI	WIOSKA	3	5
Gestion d'un parc d'exploitation	technicien	01/05/2007	31/07/2007	VAZZOLER	STREHAIANO	2	6
Administration réseaux	assistant	01/05/2009	31/07/2009	ONA	PAGNONCELL	5	8
Système d'information pour un restaurant	ouvrier	01/02/2009	28/02/2009	BOSAK	PAGNONCELL	5	7

## V.4. Champs calculés dans des formulaires

On souhaite afficher le nombre total de stages d'une entreprise dans le formulaire Entreprise.

Le nombre total est une fonction des stages, ne peut donc être calculé par Access que dans le sous-formulaire "stage". Or, nous souhaitons afficher ce montant total dans le formulaire "Entreprise". Nous allons donc créer un champ intermédiaire "Total" caché situé dans le sous-formulaire présentant les stages, puis créer un champ "Total" visible dans le formulaire entreprise, et qui fera référence au champ total caché précédemment.

- **Créer un champ "Total" dans le sous-formulaire des stages d'une entreprise**
- **Indiquer qu'il est le total des nomStage: dans Source Contrôle, créer l'expression**
- **Vérifier le bon fonctionnement**
- **Créer un champ Total dans le formulaire Entreprise, et indiquer qu'il est la recopie du champ Total du formulaire stagiaire**
- **Vérifier le bon fonctionnement**

## VI. Les états

Un « état » sous Access est destiné à imprimer efficacement sur papier les enregistrements contenus dans la base de données avec la possibilité de définir la taille et la mise en forme de chaque élément de l'état.

La plupart de ces informations contenues dans un état proviennent de tables ou de requêtes formant la source des données.

Access propose trois types d'état : le type standard « état » est utilisé généralement pour imprimer les enregistrements de la base de données sous forme d'un rapport ; l'état « graphique » est utilisé pour imprimer les enregistrements sous forme d'un graphique et le type « étiquette » permet de préparer des étiquettes pour du publipostage en exploitant les données de la base.

Il existe deux façons de créer un état. On peut utiliser l'Assistant, celui-ci vous guide dans la création d'un état simple. On dispose aussi du Mode Création dans lequel vous contrôlez toutes les composantes de l'état.

### VI.1. Création d'un état basé sur une table

- *Nous souhaitons créer un état qui imprime la liste des entreprises groupée par département.*

#### VI.1.1. Avec l'assistant

1. Créer un état basé sur la table Entreprise, en utilisant l'assistant de création d'état.
2. Y mettre tous les attributs de la table
3. Choisir « CodePostal » comme niveau de regroupement.

Ceci permet d'afficher la liste d'entreprises par groupe d'entreprise ayant le même code postal (à chaque fois que la valeur du champ « CodePostal » va changer, un nouveau regroupement sera généré).

Il est possible de faire des "sous regroupements" c.à.d. des regroupements à l'intérieur de regroupements.

- **Cliquer sur le bouton « Options de regroupement... » et choisir « 2 lettres initiales » comme intervalle de regroupement. Quel est l'intérêt de cette manipulation ?**
- **Trier les enregistrements par « CodePostal » puis par « NomEntreprise »**
- **Choisir le mode paysage et "Contour 1" comme format de présentation, puis le style « Société »**
- **Donner le nom "Eta\_Liste\_Entreprise" à votre état**
- **L'ouvrir et constater la cohérence du résultat**

### **VI.1.2. En mode création**

- **Sélectionner l'état créé, et cliquer sur mode création**
  - **Modifier le contenu de la propriété « légende » de l'étiquette « Eta\_Liste\_Entreprise » par « Liste d'entreprises » et celle de « CodePostal par 2 lettres ... » par « Département »**
  - **Repérer les différentes sections qui composent un état**
- *Par défaut, seules les sections En-tête et pied de page sont affichés. Pour afficher les cinq principales, cliquez sur Affichage, En-tête/pied d'état.*
  - *Tous contrôles insérés dans la section « En-tête d'état » seront imprimés que sur la 1<sup>ère</sup> page de l'état*
  - *Tous contrôles insérés dans la section « En-tête de page » seront imprimés au début de chaque page de l'état*
  - *Tous contrôles insérés dans la section « En tête de groupe » seront imprimés à chaque nouvelle valeur du groupe*
  - *Tous contrôles insérés dans la section « Détail » seront imprimés aussi souvent que le nombre d'enregistrements*
  - *Tous contrôles insérés dans la section « Pied de page » seront imprimés au bas de chaque page de l'état*
  - *Tous contrôles insérés dans la section « Pied de l'état » seront imprimés que sur la dernière page de l'état*
- Nous désirons imprimer le nombre d'entreprises pour chaque département dans l'entête de groupe d'état
    - **Ajouter un contrôle « zone de texte » dans l'entête du groupe**
    - **Nommer l'étiquette « Nombre d'entreprise : »**
    - **Indiquer qu'il est le total des RefEntreprise dans Source Contrôle en créant l'expression ( = compte(Refentreprise) ).**
    - **Vérifier le bon fonctionnement en cliquant sur « aperçu avant impression »**
    - **Refaire la même manipulation pour imprimer le nombre total d'entreprise dans l'entête d'état**

## VI.2. Création d'un état basé sur une requête

- Nous désirons imprimer la liste des stagiaires ainsi que les stages ( refStage, Sujet de stage, Nom de l'entreprise, Type de Stage, Date de début, date de fin) qu'ils ont effectué.
  - **Créer une requête qui inclut les champs requis.**
  - **En utilisant cette requête, créer un nouvel état en utilisant l'assistant**
  - **Choisir une présentation par « Stagiaire » et un groupement par « type de stage »**
  - **Trier par date de début de stage et choisir contour 1 comme mode de présentation**
  - **Vérifier le résultat**
  - **Modifier l'état afin d'imprimer dans ce dernier le nombre total de stagiaires et le nombre de stages effectué par stagiaire.**

# VII. Les Macros

Une macro permet d'automatiser certaines actions d'Access, qui auraient du être faites par l'utilisateur. Lorsqu'on exécute une macro, Access exécute automatiquement les actions qu'elle contient dans l'ordre dans lequel elles apparaissent. Certaines de ces actions, permettent d'afficher des fenêtres (boîtes de dialogue, formulaire, état,...), de tester la réponse fournie par l'utilisateur, d'afficher une barre de menus personnalisée... et de développer une application autonome sans avoir recours à la programmation de modules en Visual Basic (bien que les actions fassent référence à des instructions en Visual Basic).

La méthode de création d'une macro dépend de deux éléments:

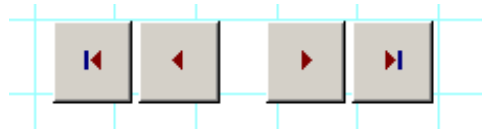
- **L'environnement de départ** de la macro : certaines macros peuvent être exécutées quelle que soit la fenêtre active, d'autres sont liées à un objet de la base de données (formulaire, état, ...)
- **L'événement** qui va déclencher l'exécution de la macro : dans un formulaire, ce peut être un clic sur un bouton, ce peut être l'ouverture du formulaire, la valeur du contenu d'un contrôle, ou encore en fonction de la mise en page d'un état.

## VII.1. Création d'un formulaire indépendant et utilisation de l'assistant de bouton de commande

Un formulaire indépendant est un formulaire qui ne sert pas à saisir les données dans une table. On distingue deux types de formulaires indépendants :

- Les formulaires de menu
  - Les formulaires de paramétrage
- **Créer un formulaire à l'aide de l'assistant basé sur la table Entreprise**
  - **Y mettre tous les attributs et choisir le format « justifié » et un style sobre.**
  - **Le nommer « frmEntrepri1 »**
  - **Créer un nouveau formulaire en mode création. Ce formulaire servira de menu. Il n'est donc basé sur aucune table/requête.**
  - **A partir de la fenêtre « propriétés » associée à ce formulaire, désactiver le sélecteur, les boutons de déplacement et les diviseurs d'enregistrements**
  - **À l'aide de la boîte à outils, ajouter à ce formulaire une étiquette « Application de gestion de stage » et le logo de l'EMAC.**

- Activer l'option « assistants contrôle » de la boîte à outils et insérer ensuite un bouton de commande dans ce formulaire. Une fenêtre « assistant de bouton de commande » s'ouvre
- Choisir la catégorie « opérations sur formulaire » et l'action « ouvrir un formulaire »
- Choisir le formulaire « frmEntreprise1 » afin d'afficher tous les enregistrements liés à ce dernier.
- Enregistrer le formulaire « menu » sous le nom « FrmMenu » et vérifier le bon fonctionnement
- 
- Ouvrir le formulaire « frmEntreprise1 » en mode création
- Ajouter dans le pied de ce formulaire un bouton qui permet de le fermer en utilisant l'assistant de bouton de commande.
- A partir de la fenêtre « propriétés » associée à ce formulaire, désactiver le sélecteur, les boutons de déplacement et les diviseurs d'enregistrements et les boutons « MinMax » et « Fermer »
- Créer dans l'entête de ce formulaire la barre de navigation suivante en utilisant l'assistant



- Vérifier le bon fonctionnement

## VII.2. Création d'une macro autonome

Une macro autonome n'est pas liée à un événement spécifique et peut être exécutée quelle que soit la fenêtre active.

- Nous voudrions créer une macro qui ouvre le formulaire « frmEntreprise1 » en mode lecture seule
  - Dans la fenêtre Objets d'Access, cliquez sur l'onglet Macro et sur Nouveau
  - Choisir dans le menu déroulant l'action ouvrir formulaire
  - Dans le volet « Arguments de l'action », choisir « frmEntreprise1 » comme Nom de formulaire et « lecture seule » comme mode de données
  - Sauvegarder la macro sous le nom « mc\_consulter\_Entr »
  - Exécuter la macro pour la tester en cliquant sur l'icône « point d'exclamation »
- Access exécutera alors toutes les actions de la macro et s'arrêtera dès qu'il va rencontrer une ligne vide, ou dès qu'il va rencontrer l'action "ArretMacro »

Pour modifier une macro, ouvrez là en mode création et se positionner sur l'action à modifier, cliquer dessus avec le bouton droit : on peut alors la supprimer ou en insérer une autre.

- Créer un formulaire « FrmEntreprise\_Ajout » basé sur la table entreprise
- Créer une macro « mc\_Ajout\_ent » qui permet d'ouvrir ce formulaire et de positionner le curseur sur le contrôle « NomEntreprise » (choisir l'action « atteindreContrôle » et préciser le nom du contrôle de les arguments de l'action).
- Tester la macro



### VII.3. Exécution d'une macro en réponse à un événement

Généralement les macros doivent être exécutées en réponse à un événement. Ces événements peuvent être générés par un formulaire (ex. ouverture, fermeture, chargement,...), ou par un contrôle du formulaire (clic, valeur du contrôle, etc.). Dans le premier cas, la macro doit être insérée dans la feuille de propriétés du formulaire, dans le second cas, elle doit apparaître dans la feuille de propriétés du contrôle. On choisira alors précisément l'événement qui va déclencher l'exécution de la macro (quand tel événement se produit, telles actions vont se dérouler).

Une macro peut également être exécutée à partir d'une autre macro ( action ExécuterMacro).

- Nous désirons ouvrir le formulaire « frmEntreprise1 » en mode consultation lorsqu'on clique sur le bouton « Liste d'entreprises »
  - **Ouvrir le formulaire « FrmMenu » en mode création**
  - **Désactiver l'option « assistants contrôle » de la boîte à outils et supprimer le bouton insérer ensuite un bouton de commande dans ce formulaire. (vous pouvez supprimer l'ancien). Nommer la légende de ce bouton « Liste d'entreprise »**
  - **Dans l'onglet « Événement » de la fenêtre « propriétés » liée à ce bouton de commande, associer la macro « mc\_consulter\_Entr » à l'événement « Sur clic » afin d'ouvrir le formulaire « frmEntreprise1 » en mode consultation lorsqu'on clique sur ce bouton.**
  - **Sauvegarder et vérifier le bon fonctionnement du formulaire**
  
- **Ouvrir le formulaire « frmEntreprise1 » en mode création**
- **Ajouter un nouveau bouton de commande nommé « Ajout d'une nouvelle entreprise » dans le pied de ce formulaire**
- **Lui associer la macro « mc\_Ajout\_ent » à son événement « sur clic »**
- **Vérifier le résultat**
  
- **Modifier le formulaire «FrmEntreprise\_Ajout » en ajoutant dans le pied de ce dernier un bouton qui permet d'enregistrer dans la base l'entreprise ainsi créée et de fermer ensuite le formulaire. Nommer cette nouvelle macro « mc\_MAJ »**
- **Vérifier le résultat dans la table « Entreprise » en insérant une nouvelle entreprise à partir de ce formulaire.**
  
- *Le contenu dans un formulaire est chargé par Access lors de l'ouverture du formulaire. Si le contenu évolue pendant que le formulaire reste ouvert (exemple : on ajoute ou supprime un enregistrement à la table lue par le formulaire), il n'est pas rafraîchi automatiquement. L'utilisateur ne peut pas donc constater les dernières mises à jour de données sans fermer et rouvrir le formulaire.*
  - **Afin d'actualiser le contenu du formulaire « frmEntreprise1 », associer une macro constituée de l'action « Actualiser » à l'événement « Sur activation » liée au contrôle formulaire.**
  - **Vérifier le résultat**
  - 
  - **Modifier le formulaire «FrmEntreprise1» afin de permettre à l'utilisateur d'imprimer la liste des entreprises en cliquant sur un bouton de commande; Penser à créer d'abord l'état permettant de générer la liste d'entreprise.**
  - **Nous désirons accéder directement à un enregistrement à partir d'un formulaire**

- **Créer une zone de texte en en-tête du formulaire «FrmEntreprise1» permettant d'atteindre directement un enregistrement donné :**
- **Donner « Zt\_N\_ref » comme nom à cette zone de texte**
- **Transcrire « Atteindre l'entreprise N°: » dans la légende de l'étiquette associée à cette zone sur l'événement « Après Mise à Jour » de la zone de texte, associer la macro suivante:**

Action	Arguments de l'action	Valeur à écrire
Atteindre contrôle	Nom_contrôle	Nom du contrôle qui contient la clé primaire (ici, RefEntreprise )
Trouverenregistrement	Rechercher	=nom_liste (ici =[Zt_N_ref ])

- *si vous oubliez le signe « = » dans l'argument rechercher, la macro ne fonctionnera pas.*
- **Afin d'éviter des erreurs de saisi, remplacer la zone de texte par une liste indépendante (Vu que la liste déroulante ne va remplir aucun champ de la table Entreprise, on ne spécifiera rien dans la propriété « source contrôle »).**
- **Paramétrer le contenu de la liste en mettant en colonne liée le champ qui servira à la recherche (en général la clé primaire : ici , RefEntrepise).**

## VII.4. Exécution d'une macro en fonction de conditions

Généralement, l'affichage d'une liste sous forme d'un état (voire dans certains cas sous forme de formulaire) se fait après avoir spécifié certaines conditions : contrainte sur une période, sur un tuteur, sur une entreprise, sur type de stage,...

La solution la plus utilisée consiste à créer un formulaire de paramétrage qui va contenir les conditions à respecter. Il faut ensuite spécifier ces derniers dans l'argument « condition where » de la macro qui sera associée à l'événement « sur\_Clic » du bouton « Afficher résultat ».

L'argument condition where attend généralement la syntaxe suivante :


- « nom d'un champ de la requête » = « valeur du contrôle du formulaire de paramétrage »
- Nous désirons imprimer la liste des stages associée à un tuteur donné.
  - **Concevoir l'état « Et\_stage »**
  - **Créer un formulaire de paramétrage « frm\_stage\_tuteur » qui demande à l'utilisateur de rentrer le nom de tuteur afin d'afficher, sous forme d'état, les stages associés à ce tuteur.**  
*comme c'est un formulaire de paramétrage, il n'y aura aucune table ou requête source (formulaire indépendant). Il est donc inutile de garder les boutons de déplacement sur enregistrements (mettre la propriété à " Non " dans le formulaire).*
  - **Nommer la zone de texte prévue pour renseigner le nom du tuteur « Zt\_Tut »**
  - **Créer une macro qui permet d'ouvrir l'état « Et-stage »**
  - **Dans l'argument « condition where » associé à l'action « ouvrir état », ajouter la condition suivante :**
    - [Tuteur]=[Formulaires]![frm\_stage\_tuteur]![Zt\_tut]
  - **Vous pouvez utiliser le générateur d'expression pour éditer cette conditio, toutefois éviter de mettre le nom du contrôle de l'état de tuteur dans la condition where (Etats!nom\_état!tuteur = Formulaires![formulaire de paramétrage]![zl\_service])**

- **Associer cette macro à l'événement « Sur clic » du bouton de commande « Afficher stage » du formulaire de paramétrage « frm\_stage\_tuteur ».**
- **Vérifier le résultat**

Dans le cas où il n'y a aucun enregistrement associé à la requête (on aurait un état qui n'affiche aucune donnée), on doit être capable d'annuler l'ouverture de l'état et d'afficher un message à l'utilisateur.

- **Sur l'événement « Sur aucune donnée » de l'état, associer une macro qui contient les 2 actions suivantes :**
  1. Boîte message (message = « il n'existe aucune donnée correspondant aux critères saisis »)
  2. Arrêt toutes macros (permet d'annuler la macro d'ouverture de l'état. Celui-ci n'est donc pas affiché).

Certaines actions d'une macro peuvent être déclenchées selon la véracité d'une condition donnée.

- Nous voudrions, lors d'un ajout d'un nouveau stagiaire, afficher un message d'erreur lorsque l'utilisateur laisse le champ « nomStagiaire » du formulaire «FrmStagiaire» vide.
  - **Créer une nouvelle macro « mc\_Verif » pour vérifier ce champ. Une fois la fenêtre de la macro ouverte cliquer sur l'icône **
  - Une nouvelle colonne "Condition", apparaît dans la fenêtre de la macro, c'est dans cette dernière qu'on saisira la condition qui décidera si oui ou non l'action de la macro doit s'exécuter.
    - **Choisir l'action « AnnulerÉvénement » afin d'annuler l'action de mise à jour si le nom de stagiaire est vide et ajouter ensuite l'action « BoîteMsg » pour qu'elle affiche le message « Vous avez oublié de saisir le nom du stagiaire »**
    - **Préciser que [NomStagiaire] Est Null (càd nom du stagiaire est vide) dans le champ « Condition » de ces deux actions**
    - **Affecter cette macro à l'événement « avant MAJ » du formulaire « FrmStagiaire »**
    - **Vérifier le résultat**

## VII.5. Lancement automatique d'une macro lors de l'ouverture d'une base de données

La macro spéciale appelée « **AutoExec** » permet d'exécuter une action ou une série d'actions lorsque la base de données est ouverte » pour la première fois. Elle est lancée automatiquement par Access lorsqu'on ouvre le fichier « .mdb »

- **Créer une macro « Autoexec » qui permet d'agrandir et ouvrir le formulaire menu**

## VIII. Compléments

- **Rajouter dans le formulaire « frm\_stage\_tuteur » la possibilité de saisir les dates de la période voulue.**
- **Afficher les critères de sélection sur l'état « Et\_stage »**

- **Modifier le formulaire « Liste des Entreprises » de telle manière quand on double-clique sur une ligne, il l'ouvre en présentant directement les stages proposés par cette entreprise cliquée.**
- **Tester la cohérence des dates de fin et début d'un stage avant de valider son insertion ou sa mise à jour dans la table.**
- **Créer un état récapitulatif donnant le détail de tous les stages, comme indiqué dans la fiche suivante**

FICHE STAGIAIRE
<b>Code Stagiaire:</b> 124 (automatique) <b>Nom :</b> Dupont <b>Prénom :</b> Nadine <b>Date de naissance :</b> 03/03/1984
<b>Adresse :</b> 18 rue des bons enfants <b>Code postal :</b> 81000 <b>Ville:</b> Albi <b>Téléphone Domicile:</b> 05 63 49 33 22 <b>Téléphone portable:</b> 06 63 41 19 67
<b>Sujet du stage :</b> Analyse de la gestion des flux <b>Type de stage :</b> Stage d'ingénieur <b>Début de Stage :</b> 9/05/2006 <b>Fin du Stage :</b> 9/05/2006 <b>Tuteur :</b> Mr. Lamine
<b>Entreprise :</b> Clinique Toulouse-Lautrec <b>Téléphone :</b> 05 63 48 45 45 <b>Adresse :</b> Val de Caussels <b>Code postal :</b> 81000 <b>Ville:</b> Albi <b>Correspondant :</b> Mr. Thary