

1. Introduction	2
2. Création d'une macro autonome	2
3. Exécuter la macro pas à pas.....	5
4. Modifier une macro.....	5
5. Création d'une macro associée à un formulaire	6
6. Exécuter des actions en fonction de conditions.....	7
7. Afficher un message.....	7
8. Déplacer le curseur	8
9. Affectation de la macro à l'événement.....	9
10. Définir la valeur d'un champ dans une macro	10
11. Mettre un bouton de commande dans un formulaire.....	11
12. Ouvrir un formulaire et afficher un enregistrement	12
13. Créer une boîte de dialogue	13

LES MACRO COMMANDES

1. Introduction

La macro commande permet d'automatiser certaines tâches d'Access, elle peut simuler une suite d'actions qui auraient du être faites par l'utilisateur.

Une macro est composée d'Actions, chaque action correspond à une tâche : lorsque vous exécutez la macro, Access exécute automatiquement les actions qu'elle contient. Certaines de ces actions, plus complexes, vous permettent d'afficher des boîtes de dialogue, de tester la réponse fournie par l'utilisateur, d'afficher une barre de menus personnalisée... et de développer une application autonome sans avoir besoin de programmer des modules en Visual Basic (bien que les actions fassent référence à des instructions en Visual Basic).

La méthode de création d'une macro est liée à deux facteurs importants :

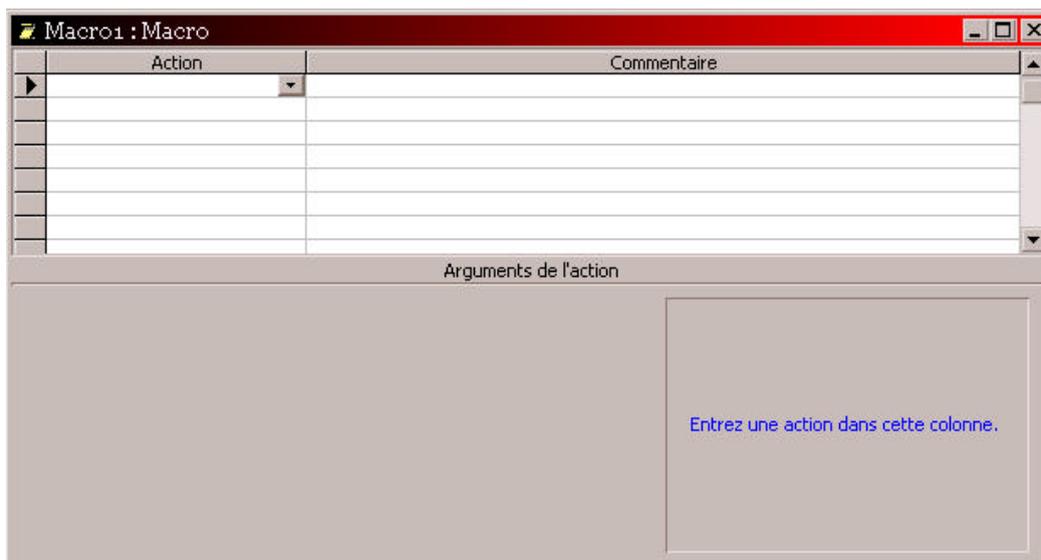
- ?? L'environnement de départ de la macro : certaines macros peuvent être exécutées quelle que soit la fenêtre active, d'autres sont liées à un objet de la base de données (formulaire, état, ...)
- ?? L'événement qui va déclencher l'exécution de la macro : dans un formulaire, ce peut être un clic sur 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.

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.

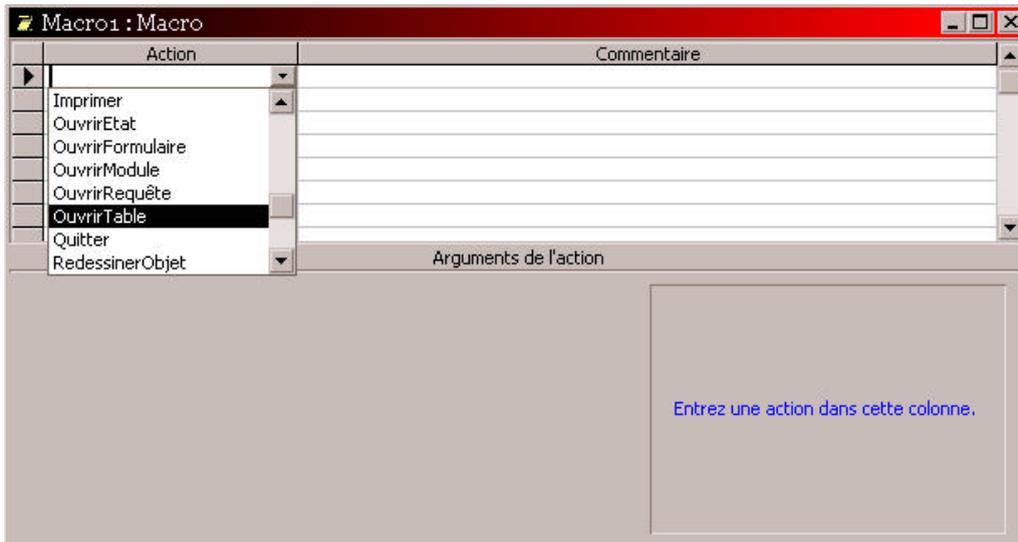
Nous allons créer une macro qui va ouvrir la table Clients et son formulaire :

1. Dans la fenêtre principale d'Access, cliquez sur l'onglet Macro et sur Nouveau :



La partie supérieure de la fenêtre (le tableau) est destinée aux différentes actions qui vont composer la macro. La plupart des actions ont des paramètres (par exemple l'action OuvrirTable qui ouvre une table demande en paramètre le nom de la table qu'elle va ouvrir), les paramètres (par exemple ici, le nom de la table) sera indiqué dans la partie inférieure de la fenêtre.

2. On choisit dans le menu déroulant l'action que l'on veut faire exécuter par la macro :



Comme vous pouvez le constater, il existe un très grand nombre d'actions possible, en fait, tout ce qu'on peut faire dans Access se retrouve dans ce menu. Pour avoir la liste des actions possibles, utilisez l'aide intégrée à Access qui décrit le fonctionnement de chaque action.

Nous voulons ouvrir une table, nous choisissons l'action "OuvrirTable".

3. Saisie des paramètres de l'action



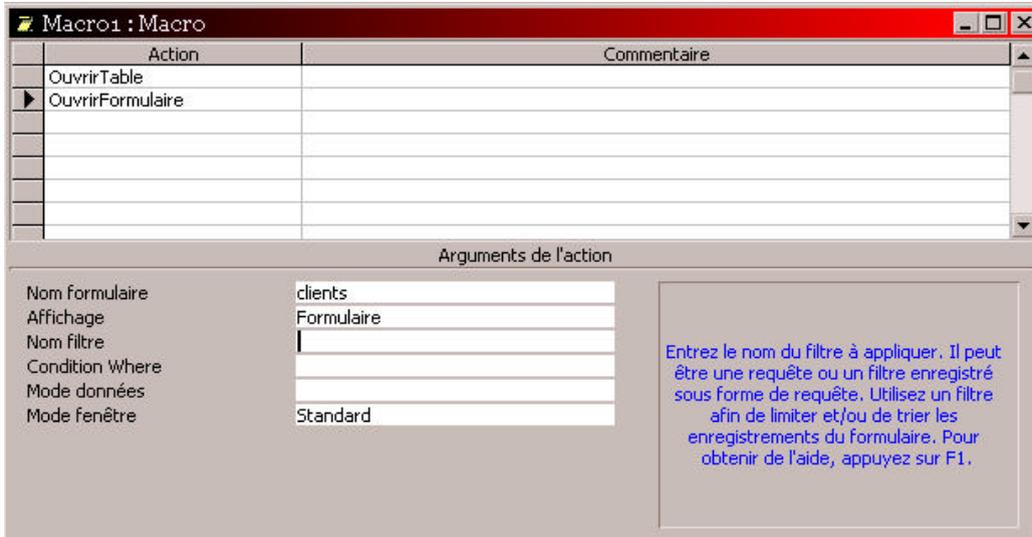
Comme vous pouvez le constater, dans le champ "Nom table" apparaissent les noms de toutes les tables de la base, on choisit "Clients".

Deux autres champs peuvent paramétrer l'action :

- ?? Affichage : détermine si la table sera ouverte en mode création ou en mode feuille de données
- ?? Mode données : Trois choix : Ajout (on peut ajouter des enregistrements à la table), Modification (on peut ajouter, modifier ou supprimer des enregistrements) et Lecture Seule (seule la consultation est permise).

4. Une deuxième action

Vous l'avez remarqué, le tableau de la fenêtre "Macros" contient plusieurs lignes, on peut exécuter plusieurs actions les unes à la suite des autres dans la même macro, maintenant, après avoir ouvert notre table Clients, nous allons ouvrir le formulaire associé à cette table :



L'Action "OuvrirFormulaire" a plus de paramètres :

- ?? Affichage : identique aux tables
- ?? Mode données : identique aux tables
- ?? Nom Filtre : on peut indiquer ici le nom d'une requête contenant les critères nécessaires à la sélection des enregistrements.
- ?? Mode fenêtre : indique comment le formulaire va être affiché : Standard (comme défini dans Affichage), Masquée (le formulaire est ouvert et automatiquement caché), Icône (le formulaire est réduit en icône), Boîte de dialogue (nous verrons ça plus tard)
- ?? Condition Where : plus tard...

5. Exécution de la macro

Sauvez votre macro.

Comme pour les requêtes, cliquez sur l'icône 'Point d'exclamation'. Access va alors exécuter toutes les lignes de la macro et s'arrêter dès qu'il va rencontrer une ligne vide, ou dès qu'il va rencontrer l'action "ArretMacro".

3. Exécuter la macro pas à pas

Pour analyser le déroulement d'une macro, on peut l'exécuter pas à pas, c'est-à-dire, action après action.
Pour cela :

1. Ouvrir la macro en mode création
2. Cliquer sur l'icône :



3. Exécuter ensuite la macro



Access affiche la première action ainsi que ses paramètres, cliquez sur :

- ?? Pas à pas : pour exécuter cette action et passer à la suivante
- ?? Arrêter : pour stopper l'exécution de la macro
- ?? Continuer : Pour poursuivre l'exécution de la macro en mode normal

4. Modifier une macro

Pour modifier une macro, ouvrez là en mode création et positionnez vous sur l'action à modifier, cliquez dessus avec le bouton droit : vous pouvez alors la supprimer ou en insérer une autre.

5. Création d'une macro associée à un formulaire

Certaines macros doivent être exécutées en réponse à un événement lié au formulaire (ouverture, fermeture, fermeture...), d'autres dépendent d'un événement lié à 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.

Il faut définir précisément quel est l'événement qui va déclencher l'exécution de la macro (quand tel événement se produit, telles actions vont se dérouler).

Prenons un exemple, nous voulons refuser l'enregistrement d'une commande si la date de cette commande n'a pas été saisie. (Nous pourrions dans les propriétés du champ date, indiquer dans la propriété Null Interdit Oui).

La question à se poser est : "Quand va-t-on vérifier que le champ date est vide ?", Il n'y a qu'une réponse : "Quand on passera à un nouvel enregistrement, que ce soit pour retourner sur un enregistrement précédent ou pour en créer un nouveau". Donc, lorsqu'on passera à un nouvel enregistrement, si la date est vide, on va afficher un message d'erreur, et positionner le curseur à nouveau sur le champ date pour pouvoir le saisir.

L'événement "passer à un nouvel enregistrement" est lié au formulaire et pas au contrôle "Date", l'événement sera donc associé au formulaire.

Voici la liste des principaux événements qui peuvent être associés à un formulaire :

Evénement	La macro doit s'exécuter
Sur Ouverture	A l'ouverture du formulaire
Sur Fermeture	A la fermeture du formulaire
Sur Activation	Lorsque Access accède à un enregistrement du formulaire
Sur Insertion	Lorsqu'on précise la valeur d'un champ d'un nouvel enregistrement
Sur Suppression	Lorsqu'on va supprimer un enregistrement
Avant MAJ	Après avoir quitté un enregistrement modifié et avant l'enregistrement des modifications
Après MAJ	Après avoir quitté un enregistrement modifié et après l'enregistrement des modifications.

En résumé, ces événements concernent l'ouverture, la fermeture du formulaire et la gestion des enregistrements.

Les événements liés aux contrôles d'un formulaire sont les suivants :

Evénement	La macro doit s'exécuter :
Sur Clic	Quand on clique sur le contrôle
Sur Double Clic	Quand on double-clique sur un contrôle
Sur Entrée	Avant d'accéder à un contrôle
Sur Sortie	Lorsqu'on quitte un contrôle
Avant MAJ	Après avoir quitté un contrôle modifié et avant qu'Access ne fasse la mise à jour
Après MAJ	Après avoir quitté un contrôle modifié et après sa mise à jour

Dans notre exemple, il faut que la macro s'exécute lorsqu'on va accéder à un nouvel enregistrement, on va donc l'affecter à l'événement "AvantMAJ" du formulaire : il faut que la vérification de la bonne saisie de la date se fasse lorsque l'on veut changer d'enregistrement mais avant qu'Access n'enregistre quoi que ce soit.

6. Exécuter des actions en fonction de conditions

1. On va créer une nouvelle macro pour vérifier la date, une fois la fenêtre de la macro ouverte, cliquez sur l'icône :



2. Une nouvelle colonne apparaît dans la fenêtre de la macro, c'est la colonne "Condition", c'est dans cette colonne qu'on va taper la condition qui décidera si oui ou non l'action de la macro doit s'exécuter.
3. On veut qu'un message d'erreur s'affiche lorsque la date de la commande est vide, la condition pour que le message s'affiche est donc : [date de la commande] Est Null. L'action qui va être déclenchée si la condition s'avère vraie va être l'affichage d'un message d'erreur.

7. Afficher un message

Pour afficher un message pendant une macro, on utilise l'action "BoîteMsg", "BoîteMsg" va afficher une boîte de dialogue Windows avec le texte et les caractéristiques qu'on va indiquer dans les paramètres de l'action.

L'action "BoîteMsg" a les paramètres suivants :

- ?? Message : texte à afficher
- ?? Bip : Emet un bip sonore lors de l'affichage du texte
- ?? Type : Icône à afficher à côté du texte (point d'exclamation, d'interrogation, etc...)
- ?? Titre : titre de la fenêtre où va s'afficher le texte

Condition	Action	Commentaire
[date de la commande] Est Null	BoîteMsg	

Arguments de l'action		
Message	<input type="text" value="Vous avez oublié de saisir la date !"/>	<p>Entrez le texte à afficher dans la barre de titre de la zone de message. Par exemple, 'Validation du code client'. Pour obtenir de l'aide, appuyez sur F1.</p>
Bip	<input type="text" value="Oui"/>	
Type	<input type="text" value="Point d'exclamation"/>	
Titre	<input type="text" value="Erreur de saisie"/>	

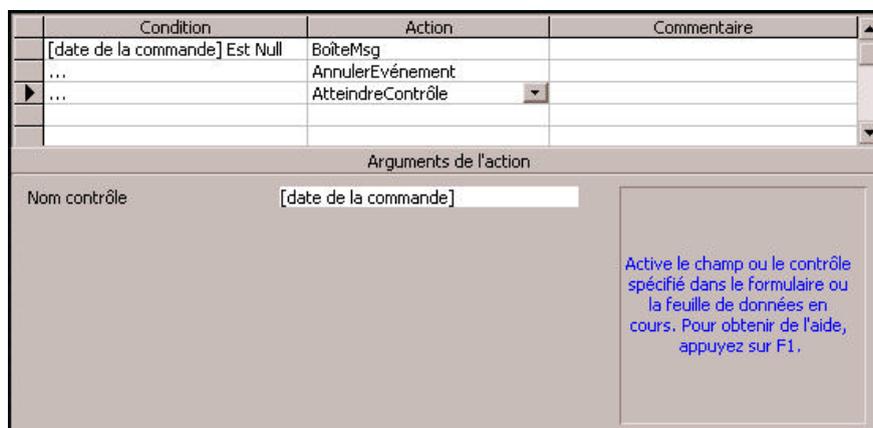
8. Déplacer le curseur

Une fois le message d'erreur affiché, il va falloir positionner le curseur automatiquement sur le contrôle Date, pour que l'utilisateur puisse le resaisir.

Il existe plusieurs actions pour déplacer le curseur :

- ?? AtteindreContrôle : Déplace le curseur sur un contrôle spécifié
- ?? AtteindrePage : Déplace le curseur sur le premier contrôle de la page spécifiée
- ?? AtteindreEnregistrement : Déplace le curseur sur l'enregistrement spécifié ou sur un nouvel enregistrement.

Nous devons déplacer le curseur dans le formulaire sur le contrôle du champ "Date", nous utiliserons donc l'action "AtteindreContrôle" :



Quoi de neuf dans cette fenêtre ?

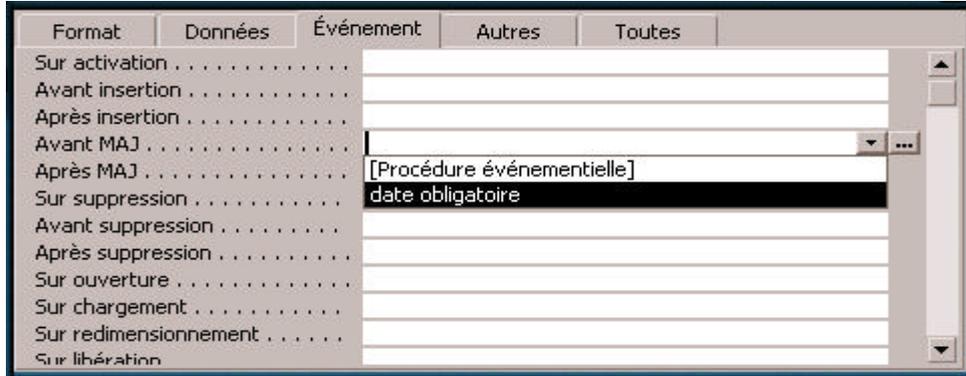
Sur la première ligne on a la condition : si la date de la commande est vide, on affiche le message d'erreur, mais après avoir affiché l'erreur que fait-on ? on va faire plusieurs choses, et pour indiquer que les autres actions que l'on va faire vont se dérouler dans le cas où l'erreur s'est produite, on va mettre comme condition "...", autrement dit, à chaque fois qu'Access va rencontrer comme condition "...", il va reprendre la condition spécifiée explicitement plus haut.

Donc, après avoir affiché l'erreur, on va exécuter l'action "AnnulerEvénement". A quoi sert cette action ? Normalement, après avoir modifié le formulaire, Access va mettre à jour les champs correspondants dans les tables liées à ce formulaire. Juste avant de faire cette mise à jour, Access va appeler la macro associée à l'évènement « AvantMAJ ». La macro va être exécutée, puis la mise à jour sera faite. Il faut qu'on annule cette mise à jour, sinon, le message indiquant que la date est vide va être affiché, puis Access va sauvegarder les modifications et sauvegarder dans la table commandes la date vide. L'action « AnnulerEvénement » annule la mise à jour qui allait être faite, ainsi, la date vide ne sera pas sauvée.

Enfin, on va déplacer le curseur sur le contrôle "Date de la commande" grâce à l'action "AtteindreContrôle", Ainsi le curseur cliquetera directement sur le contrôle où l'utilisateur devra saisir la date.

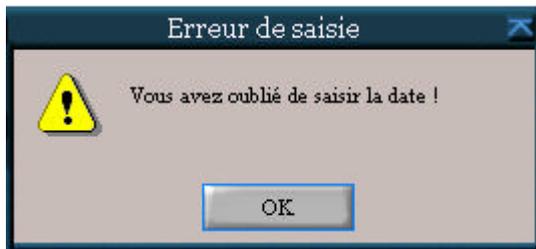
9. Affectation de la macro à l'événement

Maintenant que la macro est créée, on va l'affecter à la macro à l'événement "AvantMAJ" du formulaire "Commandes". Pour afficher les propriétés d'un formulaire, cliquez avec le bouton droit sur le carré noir situé en haut à gauche du formulaire, puis dans le menu, cliquez sur "Propriétés"



Choisissez l'événement auquel vous voulez associer la macro, dans le menu déroulant s'affichent toutes les macros existantes, on choisit "Date Obligatoire".

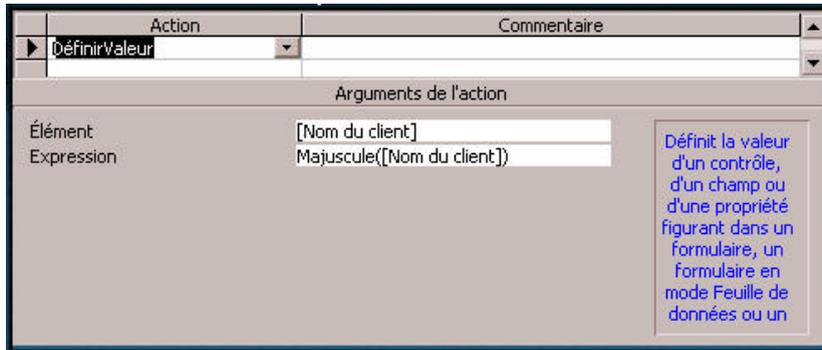
Une fois la macro associée à l'événement, il suffit d'entrer une commande dans laquelle on omet de saisir la date pour vérifier que la macro va s'exécuter (message d'erreur et repositionnement sur le contrôle) :



10. Définir la valeur d'un champ dans une macro

On peut, grâce à une macro modifier le contenu d'un champ d'une table. Imaginons que nous voulions que le nom de chaque client saisi soit en majuscule, on pourrait bien sur le faire en modifiant la propriété "Masque de saisie" du contrôle "nom du client", mais nous allons voir comment le faire à partir d'une macro. Pouvoir modifier le contenu d'un champ lors d'un événement peut s'avérer très utile.

Nous allons créer une nouvelle macro et utiliser l'action "DéfinirValeur"



Quels sont les paramètres de l'action "DéfinirValeur" ?

- ?? Élément indique qu'est-ce qui est à modifier, ce peut être comme ici le nom d'un champ, mais ce peut être aussi une propriété d'un contrôle par exemple.
- ?? Expression : Indique la nouvelle valeur de l'élément indiqué précédemment, ce peut être une valeur indiquée explicitement, le contenu d'un autre champ, le résultat d'un calcul, ou, comme ici le résultat d'une fonction. La fonction "Majuscule()" renvoie la chaîne de caractère passé en paramètre en majuscule.

Comme précédemment, on va affecter cette macro à l'événement "AvantMAJ" su formulaire.

11. Mettre un bouton de commande dans un formulaire

Nous avons vu rapidement dans les formulaires que, parmi les contrôles existants, il y avait un contrôle nommé "Bouton de Commande". Ce bouton, placé sur un formulaire, permet d'exécuter soit n'importe quelle action d'Access, soit, et c'est ce qui nous intéresse ici, une macro.

Dans notre formulaire "Bon de commande +", qui contient la commande ainsi que son détail, nous allons ajouter un bouton qui, lorsqu'on cliquera dessus, déclenchera l'impression du bon de commande.

Nous allons créer une macro qui imprimera le formulaire :



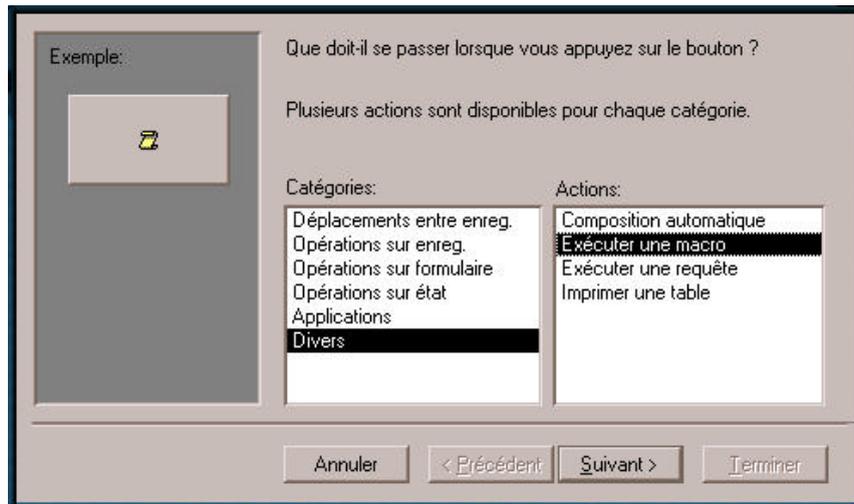
Un petit peu compliqué pour imprimer un bon de commande ?

La macro comporte trois actions, pourquoi ?

Si on n'utilise que l'action "Imprimer", Access va imprimer tous les bons de commande, il faut donc limiter l'impression au bon de commande courant, pour cela, on procède en trois étapes :

- ?? D'abord, on appelle l'action "ExecuterCommande", "ExecuterCommande" permet d'exécuter toutes les opérations possibles dans Access, on va donc choisir parmi la liste proposée, l'action "SélectionnerEnregistrement", cette action va sélectionner l'enregistrement courant. Une fois cet enregistrement sélectionné, on va pouvoir l'imprimer.
- ?? Ensuite, on appelle l'action "Imprimer", parmi les paramètres de cette commande, il y a la définition de ce qui a à imprimer : toutes les pages, un groupe, une sélection, ... on choisit la sélection.
- ?? Enfin, on appelle l'action "AtteindreContrôle", cette action va positionner le curseur sur le contrôle passé en paramètre, ce qui a pour effet de supprimer la sélection.

On insère ensuite un bouton de commande dans le formulaire, la fenêtre suivante s'affiche :



On peut choisir toutes les actions possibles, parmi celles-ci, dans la catégorie "Divers", il y a "Exécuter une macro", après avoir cliqué sur suivant, Access nous demande le nom de cette macro, on choisit celle que l'on vient de créer, puis le texte ou l'icône du bouton.

Vous pouvez essayer, ça marche !

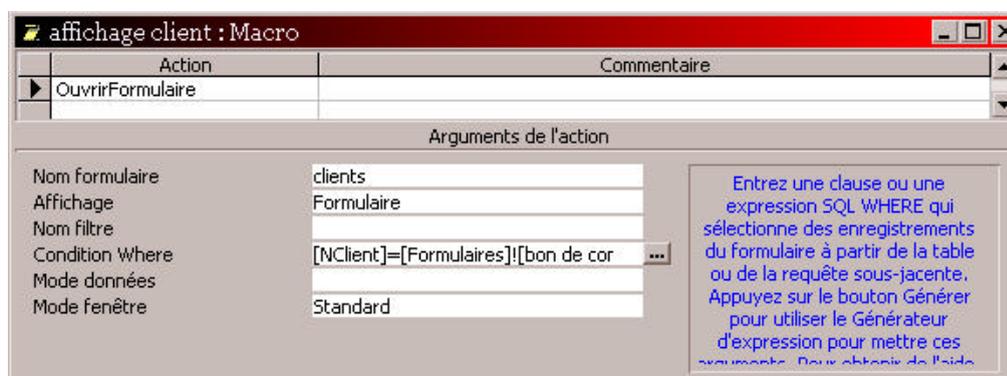
Si par la suite, vous allez regarder les propriétés du bouton qui vient d'être créé, vous verrez que la macro a été affectée à l'événement "Sur Clic", ce qui signifie, qu'à chaque fois qu'on cliquera sur le bouton, la macro se déclenchera, vous pouvez la placer ailleurs, par exemple sur l'événement "Sur Double Clic", il faudra alors double-cliquer sur le bouton pour imprimer.

12. Ouvrir un formulaire et afficher un enregistrement

Notez que cette manipulation n'est donnée qu'à titre d'exemple, on peut faire la même beaucoup plus simplement en utilisant l'assistant du bouton de contrôle.

Nous allons ajouter un bouton sur le bon de commande, ce bouton affichera les informations complètes sur le client concerné par le bon de commande :

D'abord, on crée une macro qui va ouvrir le formulaire :



On indique qu'on ouvre le formulaire "Clients", et ici, on utilise la condition Where qu'on avait vu plus tôt, on va indiquer que dans le formulaire qui va être ouvert, le champ [Nclient] (numéro de client) va être égal à [Formulaires]![bon de commande +]![Nom Client]. Ce qui signifie que le champ [Nclient] dans le formulaire "Clients" va être égal au contenu du contrôle "Nom Client" dans le formulaire "Bon de commande+".

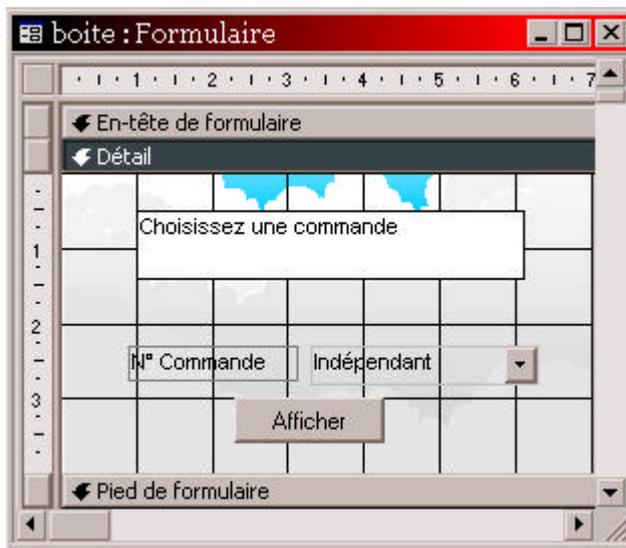
13. Créer une boîte de dialogue

Il peut être pratique de créer une boîte de dialogue pour répondre à une question par exemple, nous allons créer une boîte de dialogue dans laquelle nous allons entrer un numéro de commande, et qui affichera le formulaire correspondant à cette commande.

1. Création de la boîte de dialogue

Une boîte de dialogue est un formulaire standard qui a des propriétés particulières : Dans la fenêtre propriétés du formulaire, cliquez sur l'onglet "Toutes", et mettez la propriété "Fen Indépendante" à oui, "Fen Modale" à oui, "Auto Center" à oui, "Boîte Contrôle" à non et "Affich par défaut" à "Mode simple".

Ce qui aura pour effet de créer un formulaire qui sera centré à l'écran, qui n'affichera pas le menu système de windows en haut à gauche et qui n'affichera pas les icônes pour se déplacer sur les enregistrements.



La liste de choix contient le résultat d'une requête allant chercher les numéros de toutes les commandes.

2. On crée ensuite la macro

La macro est du même type que la précédente, la clause Where ici est :

```
[N° Commande]=[Formulaires]![boîte]![liste commandes]
```

où [N° Commande] est le nom du champ contenant le numéro de commande dans le formulaire "Bond de Commande" et [liste commandes] est le nom de la liste contenant la liste des commandes existantes.

Voilà, nous avons fait le tour rapidement des fonctionnalités offertes par l'utilisation des macros, on peut, bien sur, pousser plus loin et faire beaucoup d'autres choses avec, à vous d'expérimenter et de trouver d'autres utilisations aux macros.