

TP - IHM BD – Macros

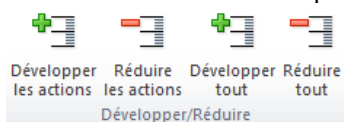
Sur la base de données ComptoirMacro.mdb à recopier depuis « Commun... » :

**Objectifs TP : L'automatisation :** Créer des macros pour prendre en charge des actions de l'utilisateur.

I. **Partie 1 – Etude**

1) Etude du formulaire "Menu Général".

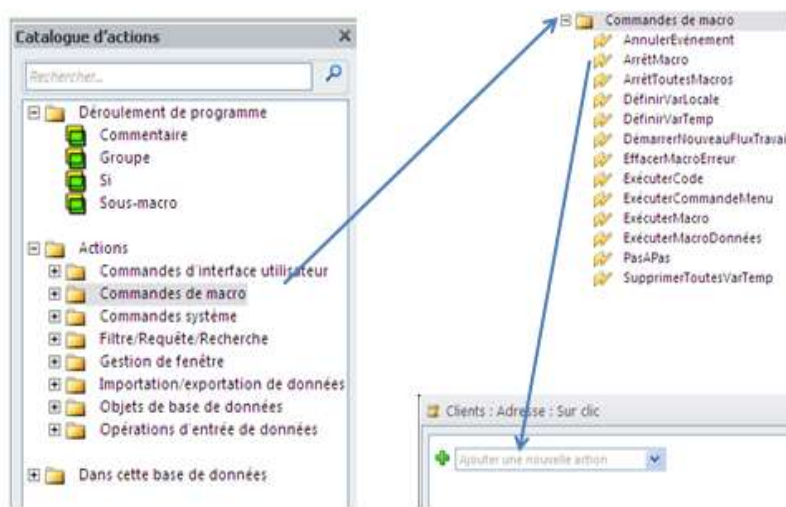
- Il propose 3 boutons de commande de noms « **Formulaires, Base de données, Quitter Microsoft Access** ».
- Chacun est associé sur sa propriété événementielle « **Sur clic** » à une **[Macro incorporée]**.
- Etudier les macros incorporées. Utiliser les boutons du ruban pour voir plus ou moins en détail :



Il existe différents types d'action de macros :

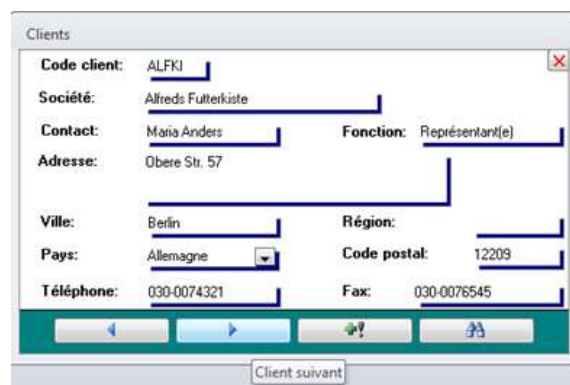
- AFFICHER DES MESSAGES (ZoneMessage, Echo)
- GERER LES OBJETS (OuvrirFormulaire, OuvrirEtat)
- EXECUTER LES OBJETS (ExecuterSQL, ...)
- GERER LES DONNEES (AtteindreControle, ...)
- ARRETER LES MACROS (Arrêtmacro, ...)

<p>Catalogue d'actions</p>	<p>Afficher toutes les actions</p>
Ouvrez le catalogue	Affichez toutes les actions



I. **Transformation du formulaire « Clients » (Utilisation de l'assistant de création des boutons de commande)**

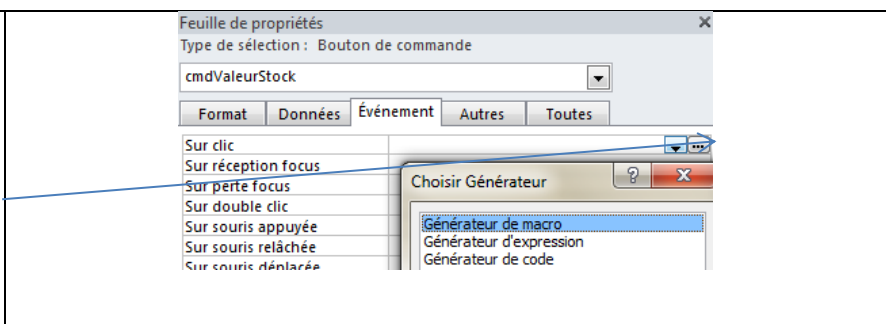
- 1) Remplacez la barre des boutons de déplacement (Précédent, Suivant, Nouveau et Recherche) par 4 boutons de commande créés à l'aide de l'assistant.
  - Nommez les boutons : *cmdPrec*, *cmdSuiv*, *cmdNouv* et *cmdRech*.
  - Mettre des textes d'info-bulle informatives.
- 2) Remplacez le bouton Fermer de la boîte de contrôle par 1 bouton de commande créé à l'aide de l'assistant.
- 3) Nommez le *cmdFermer*, renseignez l'info-bulle.
- 4) Manipulation : Testez les boutons
  - Utilisez le bouton Nouveau pour ajouter l'IUT en client.
  - Utilisez le bouton Recherche : Trouver le client de code CACTU (1), les clients de Nantes (2); les clients dont la société contient le texte « market » (3).



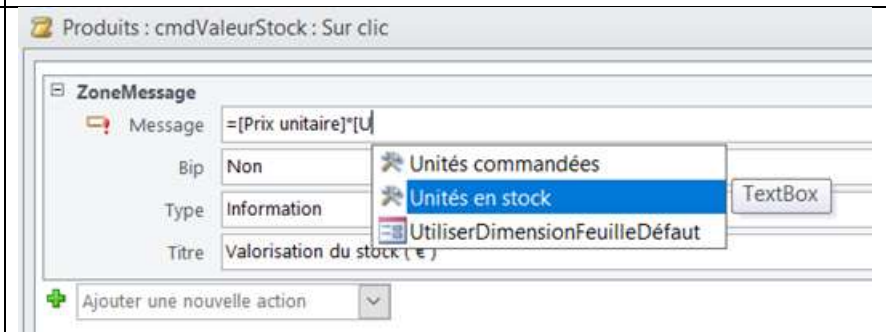
II. **Transformation du formulaire « Produits » avec un bouton de commande pour afficher la valeur du stock du produit en cours.**



- Ajoutez un bouton de commande créé sans l'aide de l'assistant.  
 - Nommez le `cmdValeurStock`, renseignez l'info-bulle.  
 - Mettez à jour sa légende :  
 Prix unitaire: 25,00 €  
 Unités en stock: 39  
 Valeur stock

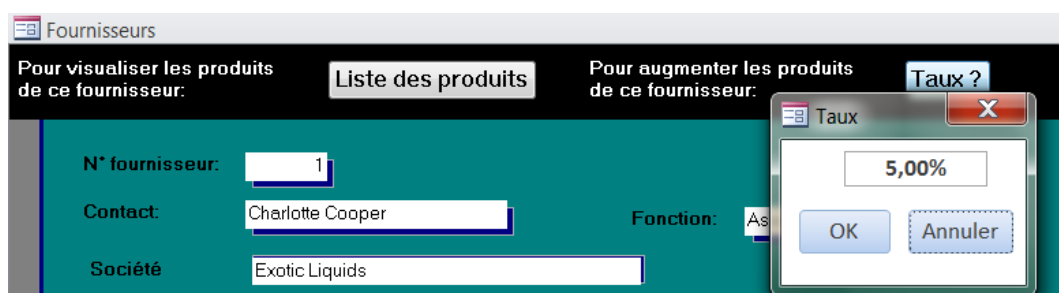


- Renseignez sa propriété événementielle **Sur clic** en **Générant du code**  
 - Vous allez créer une **Macro Incorporée** événementielle qui sera incorporée au bouton de commande et stockée dans le code associé au formulaire *Produits*.  
 - Ecrivez l'action de macro `ZoneMessage` et testez son fonctionnement



### III. Etude et transformation du formulaire « Fournisseurs Variation\_Prix » pour prendre en compte une augmentation de prix Pour un fournisseur.

On veut saisir le taux d'augmentation à appliquer à l'ensemble de ses produits dans une boîte de dialogue. Une requête SQL de Mise à jour fera le traitement désiré. Quelques contrôles seront mis en place comme vérification d'un taux saisi et si oui supérieur à 0.



#### (1) Création de la boîte de dialogue correspondante

- Créer un formulaire avec Plus de formulaires en mode Boîte de dialogue modale de nom « **Dialogue Augmentation** » (Le formulaire est en fenêtre modale : propriété **Fen modale** à **Oui**)
- Créer 1 zone de texte de nom `txtAugment` pour saisir le taux d'augmentation en pourcentage : utiliser la propriété de Format « **Format** » et les propriétés de Données « **Valide si** » et « **Message si erreur** » pour vérifier que le taux est supérieur à 0



- Nommer les 2 boutons de commande de légendes OK et Annuler : « **cmdOK** » et « **cmdAnnuler** ».

**(2) Ajout d'un bouton de commande dans le formulaire «Fournisseurs Variation\_Prix» :**

- Ajouter un bouton de commande de nom « **cmdTaux** » (sans utiliser les Assistants contrôle)
- Ajouter dans la [Macro incorporée] liée à la propriété **Sur clic** du bouton de commande « **cmdTaux** » qui :
  - Ouvre le formulaire «Dialogue Augmentation» en mode fenêtre «Standard» 'Action OuvrirFormulaire)

**(3) Fonctionnement de la boîte de dialogue:**

- Clic sur le bouton de commande Annuler « **cmdAnnuler** » >> Ferme le formulaire (Action Fermer)
- Clic sur le bouton OK « **cmdOK** » (On considère qu'il y a un taux saisi) :
  - ☞ Prévoir un message indiquant la prise en charge de l'augmentation (Action ZoneMessage)
  - ☞ Lancer la requête SQL de MAJ des prix unitaires en fonction des paramètres (Fournisseur et taux) (Action ExécuterSQL )

La requête action sera paramétrée sur le fournisseur sélectionné dans le formulaire «Fournisseurs Variation\_Prix» et le taux saisi dans la boîte de dialogue comme ci-dessous :

```
Update Produits Set [Prix unitaire] = [Prix unitaire] * (1 + [Formulaires]![Dialogue +Augmentation]![txtAugment])
Where [N° fournisseur] = [Formulaires]![Fournisseurs Variation_Prix]![ N° fournisseur];
```

- La requête peut être aussi créée avec le générateur de requête (Quand le formulaire **Fournisseurs Variation\_Prix** est ouvert) puis ensuite récupérer le code SQL.
- La requête peut être stockée dans les objets de la base puis lancée dans ce cas par l'action *OuvrirRequête* )

**(4) Améliorations possibles :**

- TEST de la présence d'un TAUX
  - **Si** il n'y a pas de taux saisi (Condition : **EstNull ([txtAugment])**)
  - >> Prévoir un message d'erreur et mettre **le focus** sur le taux (Actions *ZoneMessage* et *AtteindreContrôle*)
  - Si il y a un taux saisi >>
    - Un message indiquant la prise en charge de l'augmentation (Action *ZoneMessage*)
    - Lancer la requête SQL de MAJ des prix unitaires en fonction des paramètres (Fournisseur et taux) (Action *ExécuterSQL* )
- A l'ouverture de la boîte de dialogue mettre le focus sur la zone de saisie
- Supprimez les avertissements d'Access
- Mettre un message de confirmation
- Ouvrir la liste des produits pour vérification

Actions de macros utiles pour le TP	
<b>ZoneMessage</b> (mettre = en début de message pour une expression)	<b>DéfinirValeur</b> (affectation)
<b>OuvrirFormulaire</b> (ouvre un formulaire selon les arguments)	<b>AtteindreContrôle</b> (gestion du focus)
<b>Fermer</b> (l'objet en cours sans arguments ou l'objet spécifié)	<b>ArrêtMacro</b> (arrête la macro)
<b>SélectionnerObjet</b> (rendre un objet actif)	<b>Actualiser</b> (recalcule la source d'un objet)
<b>ExécuterSQL</b> (lance une instruction SQL LMD ( <i>Insert</i> , <i>Update</i> ou <i>Delete</i> ) de longueur max. 255 car.)	
<b>OuvrirRequête</b> (lance une requête SELECT affiche le résultat dans une feuille de données ou lance une requête SQL LMD ( <i>Insert</i> , <i>Update</i> ou <i>Delete</i> ))	
<b>ExécuterCommandeMenu</b> (exécute une commande de Menu Access)	
<b>Avertissements</b> (activation ou non des avertissements) confirmation d' Access	

**Instruction SQL-LMD Update :**  
Mise à jour des valeurs des champs d'une table, selon des critères déterminés.

**Syntaxe :**  
**UPDATE** <NomTable> **SET** <NomCol1> = <Exp1> [, <NomCol2> = <Exp2> ...] [**WHERE** <Critère>]

**Exemple :**  
**Update Produits Set [Prix unitaire]=[Prix unitaire]\*1,1 Where [Nom du produit] = "Tofu ";**  
*augmente le prix unitaire du produit de nom Tofu de 10%*