

Mise à jour de champs Access après un choix dans une liste déroulante

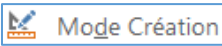
Aujourd'hui, pour l'évènement interblogueurs [Les cadeaux des blogueurs](#), je vous propose un article sur Access.


Dans ce tutoriel, nous allons faire en sorte que **les champs d'un formulaire se mettent à jour après une sélection dans une liste déroulante Access**. Nous allons pour cela utiliser une requête comme source d'enregistrement pour la liste déroulante et un peu de VBA.

Dans l'[Espace Membres](#), vous retrouverez la base de données utilisée.


Préparation du formulaire et de ses contrôles :

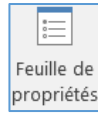
Dans la base de données que j'ai mise à votre disposition, vous constaterez que le formulaire de saisie présent est un formulaire classique c'est-à-dire qu'il a été créé à partir de la table *Clients* et qu'il contient, pour chaque champ, une source contrôle différente.

Ici, nous avons besoin de contrôles indépendants (ce n'est pas obligatoire, mais cela vous évitera d'avoir des champs remplis alors que rien n'est sélectionné dans la liste déroulante). Dans ce cas, il est préférable de réinsérer les contrôles. Nous allons donc passer en **Mode création**  et supprimer tous les contrôles présents (ce qui a pour effet de supprimer les étiquettes par la même occasion).

Dans l'onglet **Création**, vous allez insérer un contrôle *Zone de texte*  et le placer sur la partie *Détail* de votre formulaire : ce premier contrôle sera notre liste déroulante. Insérez également les autres contrôles pour la *Civilité*, le *Prénom*, le *Nom*, l'*Adresse*, le *Code postal*, la *Ville*, le *Téléphone* et l'*Email* (je n'ajoute pas la *Raison sociale*, car ce sera la donnée de la liste déroulante que nous verrons plus tard). Pour aller plus vite, vous pouvez faire un copier-coller du précédent contrôle créé (vous devrez vous retrouver avec 9 contrôles indépendants au total).

Un point très important que vous devez retenir, quand vous travaillez sur une base de données, est de toujours nommer vos contrôles et leurs étiquettes, cela se passe dans l'onglet **Autres** de

la **Feuille de propriétés**  (si elle n'est pas affichée, cliquez sur l'onglet



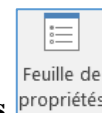
Création > Feuille de propriétés (à la ligne *Nom* (cela sera beaucoup plus facile pour quelqu'un d'autre de reprendre votre travail, mais, pour vous aussi, ce sera plus agréable de savoir à quoi chaque élément correspond).

Personnellement, je nomme toujours mon contrôle par sa *Source contrôle* et son étiquette du même nom suivi d'un underscore (_) et du mot *Étiquette* (par exemple *ID_Client* pour le contrôle et *ID_Client_Étiquette* pour son étiquette). Il faut à tout prix éviter les espaces et les caractères spéciaux, car si vous utilisez un contrôle ou son étiquette en VBA, il ou elle ne sera pas reconnu, et les accents sont à éviter, mais je ne les utilise que pour *Étiquette* puisque c'est le nom donné par défaut aux étiquettes par Access suivi par un nombre. Je vous suggère également de remplacer les tirets (-) par des underscores (_).

Étant donné que nos contrôles n'ont pas de *Source contrôle*, nous allons les nommer par la donnée qu'ils sont censés recevoir, de même que pour leurs étiquettes. Ce qui donne dans l'ordre des contrôles créés :

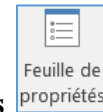
- *Client* et *Client_Étiquette*
- *Civilite* et *Civilite_Étiquette*
- *Prenom* et *Prenom_Étiquette*
- *Nom_contact* (car nommer un champ *Nom* peut poser des soucis puisque c'est un mot réservé par Access) et *Nom_contact_Étiquette*
- *Adresse* et *Adresse_Étiquette*
- *CP* et *CP_Étiquette*
- *Ville* et *Ville_Étiquette*
- *Telephone* et *Telephone_Étiquette*
- *Email* et *Email_Étiquette*

Profitez-en pour légender vos étiquettes, c'est-à-dire remplacer *Texte* avec le nombre que vous




avez par un titre dans l'onglet **Format** de la **Feuille de propriétés**, à la ligne *Légende*, ou en double-cliquant directement sur l'étiquette. Par exemple, dans l'ordre, *Raison sociale*, *Civilité*, *Prénom*, *Nom*, *Adresse*, *CP*, *Ville*, *Téléphone* et *Email*. Vous pouvez également appliquer un peu de mise en forme à votre formulaire.

Étant donné que nos contrôles vont être indépendants, cela ne sert à rien de leur laisser la possibilité d'être modifiés. Sélectionnez donc tous les contrôles, sauf le premier avec la liste




déroulante, puis dans l'onglet **Données** de la **Feuille de propriétés**, à la ligne *Activé*, choisissez *Non* dans la liste déroulante ou faites un double-clic : les champs sont grisés.


Pour terminer la préparation du formulaire, faites un clic droit sur le premier contrôle

Remplacer par > *Zone de liste déroulante*  *Zone de liste déroulante*.

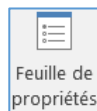
Saisie du contenu de la liste déroulante :

Sélectionnez le contrôle *Client* et, toujours dans l'onglet **Données**, à la ligne *Contenu*, cliquez sur les 3 petits points  : cela vous ouvre le **Générateur de requêtes**. Sélectionnez la table *Clients* qui contient les données affichées dans la liste déroulante ainsi que dans les autres champs, puis double-cliquez sur chacun des champs de la table, sauf *Date_demarrage* pour les

ajouter à la requête (si vous voulez vérifier que tous les champs sont présents, vous pouvez cliquer sur l'image du bouton **Affichage**  pour aller directement en **Mode feuille de**

données, puis recliquez dessus pour revenir en **Mode création** ).

Une fois que tout est correct, faites un clic droit sur l'onglet du **Générateur de requête** > *Fermer*. On vous demande si vous souhaitez enregistrer les modifications apportées à l'instruction SQL, cliquez sur **Oui**.

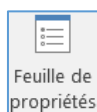


De retour sur la **Feuille de propriétés**, vous constaterez que le *Contenu* a été traduit en langage SQL.

Pour l'instant, si j'utilise mon formulaire, il ne se passe strictement rien, je n'ai que les numéros clients, je ne sais pas à qui ils correspondent et rien ne se passe après sa mise à jour.

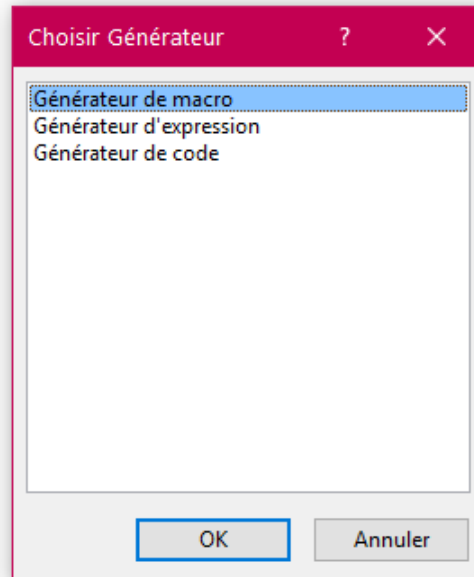
Saisie du code VBA pour mettre à jour les champs suite au choix dans la liste déroulante et paramétrage de la liste :

Gardez sélectionné le contrôle *Client* et allez dans l'onglet **Évènement** de la **Feuille de**



propriétés, cliquez sur les 3 petits points  de la ligne *Après MAJ*. Une fenêtre s'ouvre

en vous demandant de choisir le *Générateur de macro*, *Générateur d'expression* ou *Générateur de code* :



Choisissez le **Générateur de code** en double cliquant dessus ou en le choisissant et en cliquant sur **OK**.

Une fenêtre VBA s'ouvre et vous n'avez plus qu'à saisir le code désiré entre *Private Sub* et *End Sub*. C'est ici que les noms que vous avez donnés à vos différents contrôles vont être très importants.

Concernant le premier champ qui doit se mettre à jour suite au choix de la liste déroulante (il s'agit de *Civilité*), je vais donc écrire :

```
Civilite = Client.column(2)
```

J'ai saisi le chiffre 2 entre parenthèses, car la *Civilité* se trouve en 3^e position et que la "numérotation" commence à zéro dans l'instruction SQL. Si j'avais souhaité que le champ *Civilité* récupère les raisons sociales, j'aurais saisi 1 entre parenthèses, car ce champ se trouve en 2^e position. Il est très important que votre champ soit une liste déroulante, car, si en saisissant le mot *column*, la saisie semi-automatique ne vous le suggère pas, c'est qu'il doit y avoir un problème de ce côté-là (c'est sûrement ce qui a dû se passer pour les commentateurs du [tutoriel sur la création de factures avec Access](#)).

Je continue sur le même principe pour les autres champs :

Prenom = *Client.Column(3)*

Nom_contact = *Client.Column(4)*


Adresse = *Client.Column(5)*



CP = *Client.Column(6)*

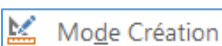
Ville = *Client.Column(7)*

Telephone = *Client.Column(8)*

Email = *Client.Column(9)*

Refermez la fenêtre VBA, essayez votre formulaire et voyez ce qu'il se passe : il ne se passe toujours rien, car nous avons indiqué dans notre code de sélectionner la colonne 2, 3, 4 etc. dans une liste qui n'a qu'une seule colonne. Pour remédier à cela, repassez en **Mode création** , toujours sur le contrôle *Client*, à l'onglet **Format** de la **Feuille de propriétés**

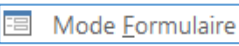
 et à la ligne *Nbre colonnes*, remplacez le 1 par 10 (car j'ai 10 champs dans mon instruction SQL). En revenant au **Mode formulaire**  et en sélectionnant un client au hasard, vous verrez que les champs se mettent à jour, mais vous ne devez pas y voir grand-chose.


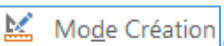

Retournez de nouveau sur l'onglet **Format** du contrôle *Client* en **Mode création**  et, à la ligne *Largeurs colonnes*, saisissez 3 cm pour toutes les colonnes (vous pouvez saisir 3;3;3;3;3;3;3;3;3;3 et Access le réécrira lorsque vous changerez de ligne).

Si vous le souhaitez, vous pouvez afficher les en-têtes de colonnes (c'est-à-dire afficher le titre de vos champs en haut de la liste) en changeant le *Non* par *Oui* à la ligne *En-têtes colonnes* et le nombre de lignes affichées dans la liste, sans avoir à utiliser la scrollbar, en modifiant le champ *Lignes affichées* qui est par défaut à 16.

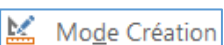
Modifiez ensuite la largeur de la liste (ligne *Largeur liste*) qui est par défaut à *Auto* à 30 pour commencer (notez que si vous mettez une largeur de liste plus grande que le total des largeurs colonnes, le dernier champ récupérera le surplus). Ne confondez pas la *Largeur liste* et la

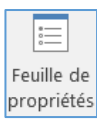
Largeur : la première est la largeur de la liste lorsque vous cliquez dessus et la seconde est la largeur de base.

Retournez en **Mode formulaire**  et choisissez un client : vous verrez que le champ client conserve le numéro client, car c'est son premier champ affiché. Dans ce cas, peu nous importe le numéro client, nous allons donc le masquer. Retournez dans l'onglet **Format**

de la **Feuille de propriétés**  en **Mode création**  et, à la ligne *Largeurs colonnes*, remplacez le premier 3 par 0. Désormais, en **Mode formulaire** , la *Raison sociale* reste dans le champ *Client*. Il ne vous reste plus qu'à modifier la propriété *Largeur colonnes* en fonction des données qui doivent s'afficher dans votre liste déroulante.

Concaténer deux ou plusieurs champs en code VBA :

Si je souhaite concaténer les champs *Adresse*, *Code postal* et *Ville* en un seul champ, il me suffit de l'écrire en VBA. Pour cela, je peux déjà supprimer les contrôles *CP* et *Ville* ainsi que leurs étiquettes en **Mode création**  et agrandir en hauteur et en largeur le contrôle *Adresse* (8 cm de largeur et 1,5 cm de hauteur suffisent).

Puis, à l'onglet **Évènement** de la **Feuille de propriétés**  pour le contrôle *Client*, à la ligne *Après MAJ*, je me place à la suite de *Adresse = Client.column(5)* et je supprime le saut de ligne et *CP =* afin de ramener *Client.column(6)* sur la même ligne. Entre ces deux expressions, je saisis une esperluette (&). Je fais de même avec la ligne *Ville*.

Votre code doit ressembler à cela :

```
Private Sub Client_AfterUpdate()
```

```
  Civilite = Client.Column(2)
```

```
  Prenom = Client.Column(3)
```

```
  Nom_contact = Client.Column(4)
```

```
  Adresse = Client.Column(5) & Client.Column(6) & Client.Column(7)
```

```
  Telephone = Client.Column(8)
```

```
  Email = Client.Column(9)
```

```
End Sub
```

Si je laisse mon code tel quel, les trois champs vont être collés (sans espaces). Je vais donc ajouter une espace entre chaque champ en remplaçant la ligne de code de l'adresse par :


```
Adresse = Client.Column(5) & " " & Client.Column(6) & " " & Client.Column(7)
```

Si vous souhaitez aller plus loin en ajoutant du texte entre chaque champ, il vous suffit d'utiliser la même syntaxe que pour les espaces (n'oubliez jamais que les appels aux colonnes doivent toujours être encadrés d'esperluettes auquel cas vous aurez droit à une erreur de compilation).

Pour terminer, si vous souhaitez ajouter des retours à la ligne, voire même des tirets, voici une syntaxe que vous pouvez utiliser :

```
Adresse = Client.Column(5) & Chr(13) + Chr(10) & Client.Column(6) & " - " &  
Client.Column(7)
```

Pour obtenir ce résultat :



```
2, rue de l'Église  
95690 - NESLES-LA-VALLÉE
```

Vous savez désormais comment mettre à jour des champs en VBA pour Access après une sélection dans une liste déroulante.

Note : ce tutoriel vous permet de rechercher des informations dans un formulaire basé sur 2 ou plusieurs tables, mais, pour un formulaire n'ayant qu'une seule table comme source, l'article

Rechercher un enregistrement avec une liste déroulante dans un formulaire Access sera plus simple et rapide.

Tutoriel réalisé avec Access 2016

[Voir la version vidéo de cet article](#)