

# Création de Sous-Formulaires

*Révision 1.01 du 02/01/04*

Réalisé avec : **OOo 1.1.0**

Plate-forme / Os : **Toutes**

*Distribué par le projet Fr.OpenOffice.org*

## Table des Matières

1	<a href="#">But de ce how-to</a>	3
2	<a href="#">Pré-requis</a>	3
3	<a href="#">Détail de l'exemple de cet how-to</a>	3
4	<a href="#">Création de la première table dBase avec Calc</a>	4
5	<a href="#">Création de la source de données</a>	5
6	<a href="#">Création de la deuxième table dBase depuis la source de données</a>	6
7	<a href="#">Création du formulaire principal avec l'AutoPilote</a>	7
8	<a href="#">Création du sous-formulaire manuellement</a>	8
9	<a href="#">Lier le sous-formulaire au formulaire principal</a>	9
10	<a href="#">Le formulaire et le sous-formulaire utilisent la même table (1)</a>	10
11	<a href="#">Le formulaire et le sous-formulaire utilisent la même table (2)</a>	11
12	<a href="#">Créer un sous-formulaire avec une clé sur plusieurs champs</a>	12
13	<a href="#">D'autres astuces à essayer par vous même</a>	13
14	<a href="#">Crédits</a>	14
15	<a href="#">Licence</a>	14

### **Qu'est-ce qu'un sous-formulaire ?**

La génération de formulaires s'effectue d'après une table ou une requête de base de données. Ils permettent de présenter les données et peuvent être utilisés pour saisir de nouvelles données ou pour modifier des données qui sont enregistrées dans la table ou la requête de base de données sur laquelle s'appuie le formulaire

Pour obtenir un formulaire pouvant faire référence aux données d'une table ou d'une requête et pouvant également afficher des données d'une autre table, vous devez créer un sous-formulaire. Ce sous-formulaire peut être, par exemple, une zone de texte affichant les données d'une autre table de base de données.

Un sous-formulaire est un "composant auxiliaire du formulaire principal". Le "formulaire principal" est appelé "formulaire parent". Il est créé sur la base d'une table de base de données comportant les principales données du formulaire. Toutes les autres données gérées dans d'autres tables de base de données sont créées en tant que composants auxiliaires sous forme de sous-formulaires. Les sous-formulaires sont donc utiles lorsque vous souhaitez accéder à plus d'une table dans un formulaire. Un sous-formulaire est nécessaire pour chaque table supplémentaire.

*Extrait de l'aide OOo1.1.0*

## 1 But de ce how-to

Ce how-to est destiné aux personnes désirant réaliser des sous-formulaires.

Quelques exemples d'utilisation de sous-formulaires :

- ➔ Créer, modifier ou supprimer, dans le formulaire principal, des informations sur un client et, dans une table d'un sous-formulaire, la liste de ses factures ( exemple détaillé dans cet how-to ).
- ➔ Afficher la liste des clients dans une table du formulaire principal et la fiche détaillée du client sélectionné dans le sous-formulaire pour le modifier ou le supprimer.
- ➔ Créer, modifier ou supprimer simultanément les informations provenant de deux tables ne pouvant pas être reliées par une requête SQL ( ex : tables dBase ) ou deux tables provenant de deux sources de données différentes.

## 2 Pré-requis

Ce how-to est destiné à des personnes ayant déjà des connaissances dans les domaines suivants :

- ➔ Création et utilisation des sources de données ( Outils / Sources de données ou F4 )
- ➔ Réalisation de formulaire à l'aide de l'auto-pilote ( Fichier / AutoPilote / Formulaires... )
- ➔ Un minimum de connaissances dans le langage SQL ( ex : `select * from client where code=10` )

## 3 Détail de l'exemple de cet how-to

Dans cet how-to, je vais détailler la réalisation d'un exemple qui servira de base à la compréhension de l'ensemble :

- ➔ Réalisation, avec deux méthodes différentes, de deux tables dBase.
- ➔ Création de la source de données qui servira à la réalisation des formulaires.
- ➔ Création du formulaire principal à l'aide de l'auto-pilote pour afficher le détail du client.
- ➔ Création du sous-formulaire pour afficher la liste des factures.

Le résultat final devrait ressembler à ceci :

The screenshot shows a database application window. At the top, there are three input fields: 'CODECLIENT' with the value '1001', 'CLIENT' with the value 'Client 1', and 'VILLE' with the value 'Paris'. Below these fields is a table with the following data:

	NUMFACTURE	INTITULE	MONTANT
▶	1	Essai 1	10,20
	2	Essai 2	99,95
*			

At the bottom of the window, there is a status bar showing 'Enregistrement 1 de 2' and navigation icons.

## 4 Création de la première table dBase avec Calc

J'ai choisi le format dBase car c'est la seule base de données qu'il est possible de créer directement avec OOo et qui ne nécessite pas l'installation de programmes complémentaires.

Cependant, il est parfaitement possible ( voire souhaitable ) d'utiliser une autre base de données ( ex : MySQL ) pour réaliser cet how-to.

Procédure pour créer la première table dBase :

- ➔ Créer avec la méthode de votre choix un nouveau dossier destiné à recevoir les 2 tables dBase et éventuellement votre formulaire ( ex : SousFormulaire )
- ➔ Dans une nouvelle feuille Calc, saisir les informations suivantes ( ou faire un copier / coller ) :

<b>CODECLIENT,N,6,0</b>	<b>CLIENT,C,50</b>	<b>VILLE,C,50</b>
1001	Client 1	Paris
1002	Client 2	Lyon
1003	Client 3	Dijon

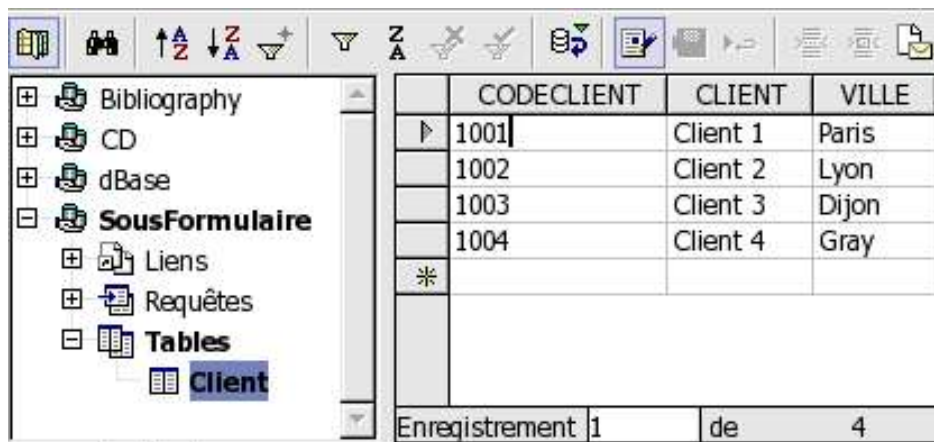
- ➔ La première ligne de ce tableau, contient la définition complète des champs de la table :
  - ➔ CODECLIENT = Nom du 1er champ de la table sur 12 caractères maximum, en majuscule, et sans espaces, caractères spéciaux ou caractères accentués.
  - ➔ ,N = Signifie que ce sera un champ de type Numérique ( ex : CODECLIENT )
  - ➔ ,C = Signifie que ce sera un champ de type Caractères ( ex : CLIENT et VILLE )
  - ➔ ,6,0 = Signifie que le champ numérique sera sur 6 chiffres avec 0 décimale ( ex : CODECLIENT )
  - ➔ ,50 = Signifie que le champ sera sur 50 caractères.
- ➔ Enregistrer le fichier avec le menu « Fichier / Enregistrer sous »
  - ➔ Choisir pour l'exemple « Client » comme nom de fichier. **ATTENTION** : Le nom du fichier ne doit pas contenir d'espaces, de caractères spéciaux et ne pas faire plus de 10 caractères.
  - ➔ Choisir « dBase » dans la zone « Type de fichier » et cliquer sur « Enregistrer »
  - ➔ Choisir dans la liste « Europe Occidentale ( Windows 1252 / WinLatin 1 ) » et cliquer sur OK. Vous pouvez aussi utiliser le jeu de caractères Europe occidentale ISO8859-15.
  - ➔ **ATTENTION** : Si vous choisissez une autre option, vous risquez de perdre les caractères spéciaux ( accents... )
- ➔ Fermer le fichier en répondant « Oui » à l'avertissement de pertes de données

## 5 Création de la source de données

Créer la nouvelle source de données qui pointera sur le nouveau dossier contenant la table :

- ➔ Menu « Outils / Sources de données... »
- ➔ Bouton « Nouvelle source de données »
- ➔ Indiquer un nom quelconque à la source de données ( ex : SousFormulaire )
- ➔ Dans « Type de BD », sélectionner « dBase »
- ➔ Cliquer sur « ... » pour aller rechercher le nouveau dossier contenant la table ( ex : SousFormulaire )
- ➔ Valider la création de la source de données en cliquant sur OK.

Normalement, vous devez pouvoir consulter et modifier les données de la nouvelle table depuis les sources de données ( Menu « Affichage / Source de données » ou touche « F4 » )

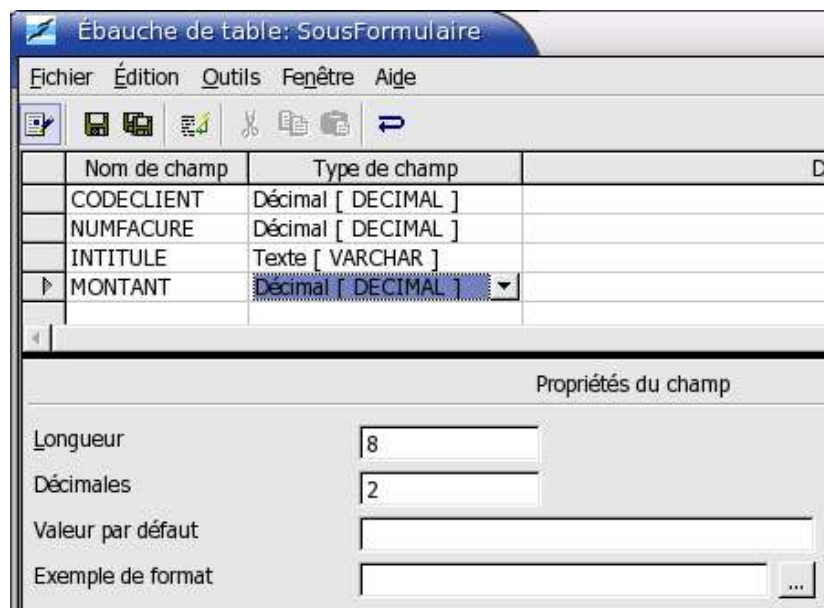


	CODECLIENT	CLIENT	VILLE
	1001	Client 1	Paris
	1002	Client 2	Lyon
	1003	Client 3	Dijon
	1004	Client 4	Gray
	*		
Enregistrement 1		de	4

## 6 Création de la deuxième table dBase depuis la source de données

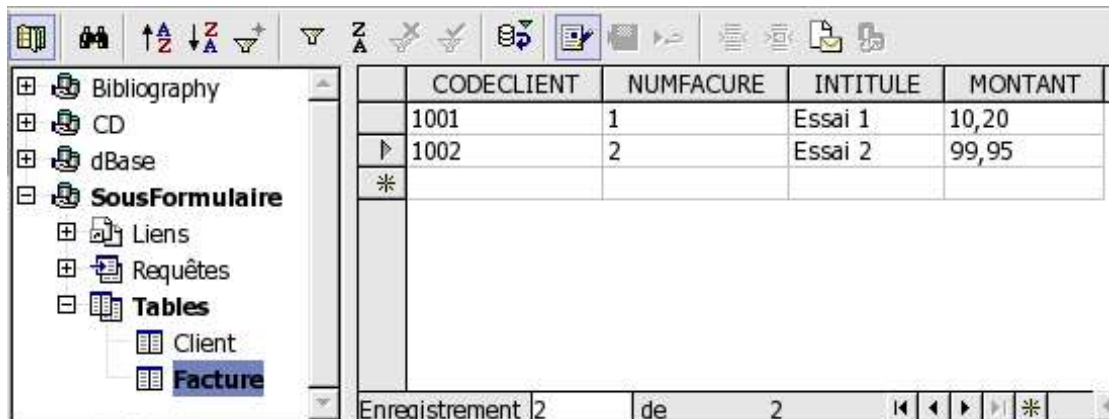
Pour créer la deuxième table, il est possible d'utiliser la même méthode que précédemment mais, dans un but purement instructif, nous allons utiliser une autre méthode :

- Afficher les sources de données ( Menu « Affichage / Source de données » ou touche « F4 » )
- Afficher la table créée précédemment en cliquant successivement sur tous les « + »
- Faire un clic droit sur « Tables » et sélectionner « Nouvelle ébauche de table »
- Saisir les informations indiquées ci-dessous :
  - CODECLIENT -> Type=Décimal, Longueur=6, Décimales=0
  - NUMFACTURE -> Type=Décimal, Longueur=6, Décimales=0
  - INTITULE -> Type=Texte, Longueur=50
  - MONTANT -> Type=Décimal, Longueur=8, Décimales=2



- Cliquer sur la disquette pour enregistrer la définition de la table et indiquer « Facture » comme nom pour cette nouvelle table
- Cliquer sur la croix en haut à droite pour fermer la fenêtre « Ébauche de table »

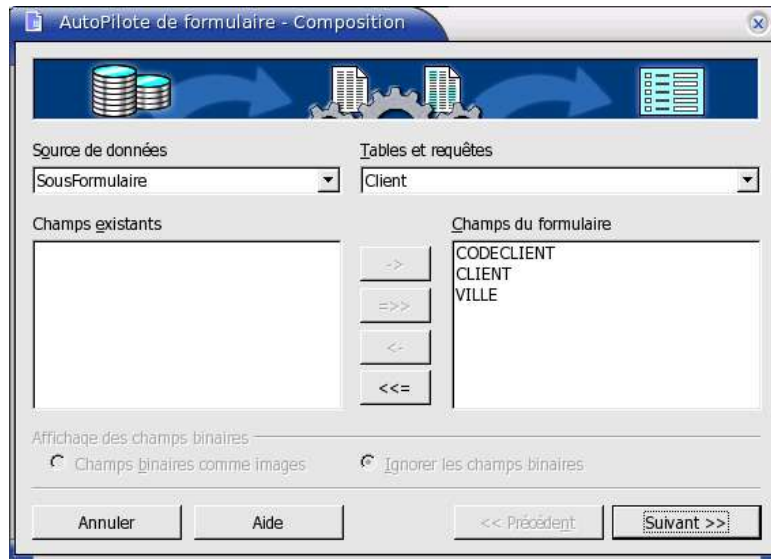
Normalement, la nouvelle table doit apparaître dans les sources de données et vous devez pouvoir ajouter des données dans cette table.



## 7 Création du formulaire principal avec l'AutoPilote

Suivez la procédure suivante pour créer le formulaire principal :

- ➔ Menu « Fichier / AutoPilote / Formulaire... »
- ➔ Sélectionner votre source de données ( ex : SousFormulaire )
- ➔ Sélectionner votre table ( ex : Client )
- ➔ Cliquer sur la flèche « =>> » pour utiliser tous les champs dans votre formulaire



- ➔ Cliquer sur le bouton « Suivant »
- ➔ Dans la zone 'disposition des champs de BD', cliquer sur le premier bouton en haut à gauche, par exemple
- ➔ Cliquer sur le bouton « Créer »
- ➔ Donner le nom et le chemin que vous voulez à votre formulaire ( pour des raisons de commodité, vous pouvez l'enregistrer au même endroit que votre base de données mais ce n'est pas obligatoire )
- ➔ Si vous avez suivi l'exemple, vous devriez avoir un écran comme ci-dessous :





CODECLIENT	1001
CLIENT	Client 1
VILLE	Paris

- ➔ Vous pouvez, dès maintenant, commencer à utiliser votre formulaire pour consulter, modifier ou créer les fiches de vos clients à l'aide de la barre de navigation située en bas du formulaire



## 8 Création du sous-formulaire manuellement

Pour créer le sous-formulaire, il n'existe pas d'AutoPilote. Nous allons donc devoir le créer manuellement.


- ➔ Afficher la barre d'outils des fonctions de formulaire  ( 5ème icône de la barre verticale si vous n'avez pas personnalisé votre barre d'instruments )
- ➔ Activer le mode conception  ( 6ème bouton en bas de la barre de fonctions de formulaire )
- ➔ Afficher le navigateur de formulaires  ( 5ème bouton en bas de la barre de fonctions de formulaire )
- ➔ Cliquer sur « Standard » qui correspond au nom du formulaire principal
- ➔ Pour une meilleure compréhension, renommer le nouveau formulaire « Standard » en « Client » en faisant un clic droit + « Renommer »
- ➔ Ajouter un sous-formulaire en faisant un clic-droit + « Nouveau / Formulaire » sur le formulaire « Client » puis saisir « Facture » pour le renommer
- ➔ Sélectionner ce nouveau sous-formulaire « Facture »
- ➔ A l'aide du bouton Contrôle de table  ( dernier bouton en haut de la barre de fonction de formulaire ), créer un contrôle de table en dessous des champs du formulaire principal. ( ce contrôle de table contiendra la liste des factures d'un client ). Pour cela :
  - ➔ Sélectionner la source de données « SousFormulaire »
  - ➔ Sélectionner la table « Facture » et cliquer sur « Suivant »
  - ➔ A l'aide de la flèche « -> » faites passer tous les champs dans la partie droite, sauf le champ « CODECLIENT »
  - ➔ Cliquer sur le bouton « Créer »



CODECLIENT	1001
CLIENT	Client 1
VILLE	Paris

	NUMFACTURE	INTITULE	MONTANT
▶	1	Essai 1	10,20
	2	Essai 2	99,95
*			

Enregistrement 1 de 2

Pour voir le résultat, désactiver le mode conception  ( 6ème bouton en bas de la barre de fonction de formulaire ).

Normalement, vous devez pouvoir naviguer dans les clients mais la liste des factures doit toujours afficher les factures de tous les clients.


**Attention** : Pour ceux qui configurent les Paramètres Linguistiques pour utiliser le point du pavé numérique comme virgule dans Calc, le formatage par défaut des colonnes d'un contrôle de table sur des données numériques induira un affichage au format date 'très spécial' à la prochaine réouverture du formulaire.

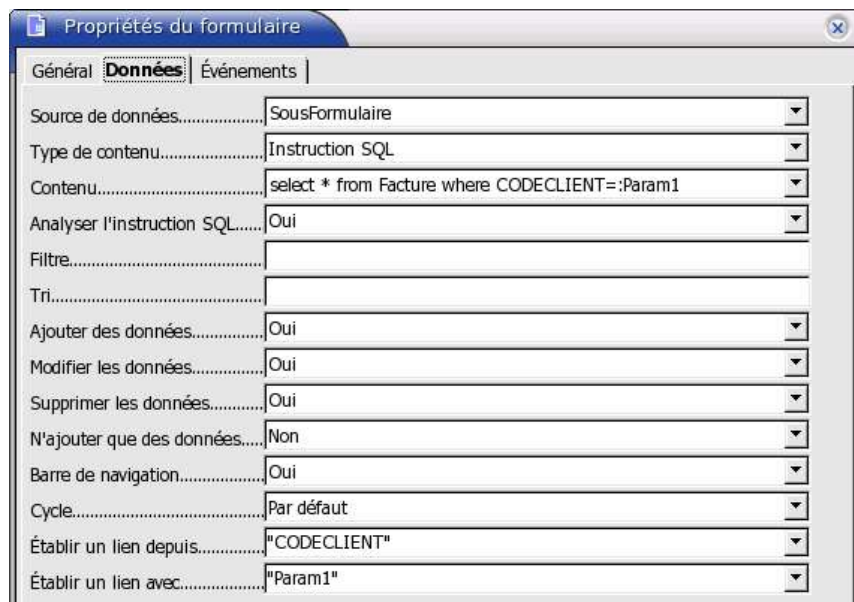
Avant de le fermer il faut donc, en mode conception, **formater toutes les colonnes numériques** du contrôle de table (clic droit/Colonne/Formatage) avec le Format Nombre désiré.



## 9 Lier le sous-formulaire au formulaire principal

Nous allons devoir écrire une requête SQL paramétrée, dans le sous-formulaire, pour arriver à lier le sous-formulaire au formulaire principal :

- ➔ Repasser en mode conception de formulaire 
- ➔ Dans le navigateur de formulaire, faire un clic droit sur le formulaire « Facture » et cliquer sur « Propriété »
  - ➔ Cliquer sur l'onglet « Données »
  - ➔ Dans la zone « Type de contenu », sélectionner « Instruction SQL »
  - ➔ Dans la zone « Contenu », saisir l'instruction paramétrée suivante :  
« **select \* from Facture where CODECLIENT=:Param1** »
  - ➔ Dans la zone « Établir un lien depuis », saisir « CODECLIENT ». Il faut indiquer, dans cette zone, le nom du champ de la table du formulaire principal servant de lien avec le sous-formulaire
  - ➔ Dans la zone « Établir un lien avec », saisir « Param1 ». Il faut indiquer dans ce champ le nom du paramètre utilisé dans la requête SQL après le signe « : ». Le signe « : », dans une requête SQL, indique que celle-ci est paramétrée et qu'il faut donc fournir un paramètre avant de l'exécuter



- ➔ Fermer la fenêtre des propriétés, en cliquant sur la croix en haut à droite
- Normalement, si tout est correctement saisi, tout doit fonctionner correctement.

Pour voir le résultat, désactiver le mode conception .

Vous devez pouvoir naviguer dans les clients et la liste des factures doit correspondre au client affiché.

Il est possible, avec ce formulaire, de consulter, de créer ou de modifier des clients ou les factures de chaque client.



**Dans les chapitres suivants, je vais vous donner d'autres idées et astuces sur les sous-formulaires. Pour les tester, il est impératif d'avoir bien compris le fonctionnement de l'exemple précédent et d'avoir de bonnes notions du langage SQL.**



## 11 Le formulaire et le sous-formulaire utilisent la même table (2)

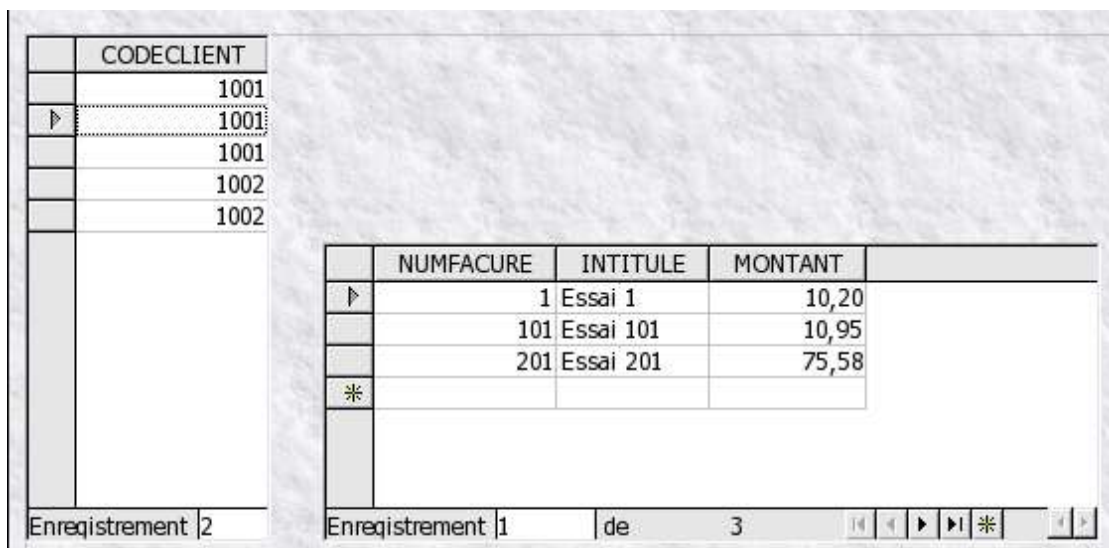
A partir de l'exemple précédent, il pourrait être intéressant de voir la liste des CODECLIENT dans un contrôle de table et d'afficher la liste des factures dans un autre contrôle de table lors de la sélection d'un CODECLIENT.

Dans ce cas, il faut remplacer le contrôle simple du formulaire principal par un contrôle de table :

- ➔ Supprimer le contrôle « CODECLIENT »
- ➔ Afficher le navigateur de formulaires 
- ➔ Cliquer sur le formulaire « Client », qui correspond au formulaire principal
- ➔ A l'aide du bouton Contrôle de table , créer un contrôle de table ( ce contrôle de table contiendra la liste des Codes Clients )
- ➔ A l'aide de la flèche « -> » faites passer le champ « CODECLIENT » à droite
- ➔ Cliquer sur le bouton « Créer »

Pour voir le résultat, désactiver le mode conception .

Vous devez pouvoir naviguer dans les clients mais la liste des clients peut comporter des doublons si vous utilisez une base de données au format dBase (cf remarque au chapitre précédent)



## 12 Créer un sous-formulaire avec une clé sur plusieurs champs.

Il arrive fréquemment que la clé primaire d'une table se compose de plusieurs champs ( ex : SOCIETE + CODECLIENT ).

Pour tester cet exemple, il faut déjà commencer par ajouter le champ SOCIETE ( ex : Texte sur 10 caractères ) dans les tables « Client » et « Facture » de cet how-to :

- ➔ Afficher les tables dans les sources de données
- ➔ Clic droit + « Éditer la table » pour modifier la structure de la table ( **attention** : ajouter les champs à la fin de la liste )

Il faut ensuite modifier les propriétés du sous-formulaire.

- ➔ Afficher la fenêtre des propriétés du sous-formulaire « Facture »

Il faut modifier l'instruction SQL en conséquence :

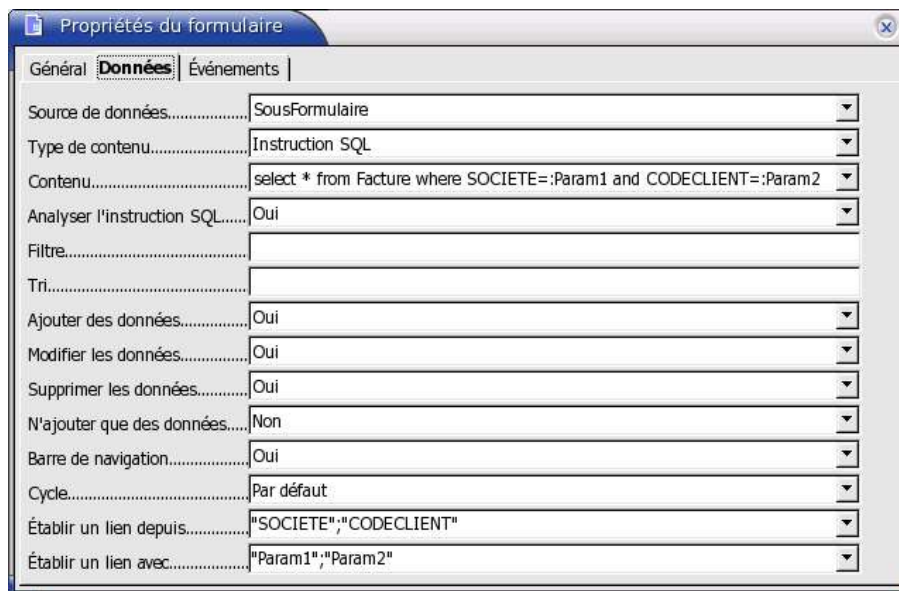
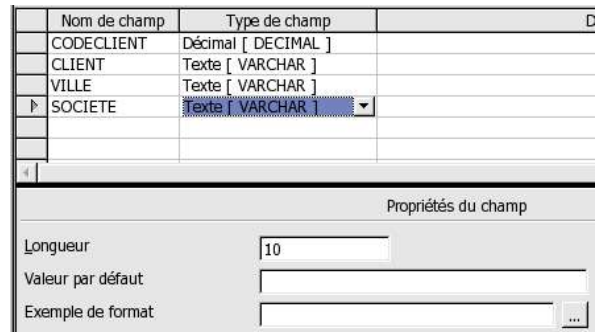
- ➔ Dans la zone « Contenu », saisir l'instruction SQL paramétrée suivante :  
« **select \* from Facture where SOCIETE=:Param1 and CODECLIENT=:Param2** »

En plus, il faut arriver à mettre deux valeurs dans les zones « Établir un lien depuis » et « Établir un lien avec ». Pour cela, la seule solution que j'ai trouvée est de saisir les deux valeurs dans deux cellules d'une feuille Calc et de faire un Copier / Coller ou « CTRL + C » / « CTRL + V »

➔ Renseigner avec les deux valeurs la zone « Établir un lien depuis » des propriétés du formulaire :

- ➔ Saisir dans une feuille Calc : « SOCIETE » en « **A1** » et « CODECLIENT » en « **A2** »
- ➔ Sélectionner et copier les deux cellules « CTRL + C »
- ➔ Dans la zone « Établir un lien depuis » des propriétés du formulaire, faire « CTRL+V »

➔ Renseigner les deux valeurs ( Param1 et Param2 ) dans la zone « Établir un lien avec » des propriétés du formulaire avec la même méthode.



Avec ce paramétrage, le sous-formulaire doit fonctionner correctement. Avec cette astuce, il est donc possible de réaliser des sous-formulaires avec des tables dont la clé primaire est sur plusieurs champs.

## 13 D'autres astuces à essayer par vous même

Avec les sous-formulaires, il est possible de réaliser d'autres opérations que je vous laisse expérimenter vous-même :

- ➔ Il est possible d'utiliser une source de données pour le formulaire principal et une autre source de données pour le sous-formulaire. Un sous-formulaire permet donc de relier deux bases de données **de type différent** ( ex : dBase et MySQL ), ce qui est impossible à faire avec une requête SQL.
- ➔ Il est possible de créer des sous-sous-formulaire ( 3 niveaux ) et même plus ( j'ai testé 4 niveaux ). On pourrait imaginer, pour notre exemple, d'avoir la liste des clients dans une table, qui donnerait la liste des factures dans une autre table d'un sous-formulaire et enfin le détail de la facture dans le sous-sous-formulaire.

Voilà, c'est la fin de cet how-to et j'espère avoir réussi à vous faire découvrir une partie des merveilles de OOo, sans vous donner mal à la tête...

Si vous avez d'autres astuces sur les sous-formulaires, n'hésitez pas à me contacter pour que je les ajoute à cet how-to dans une prochaine version.

J'aimerais également, dans un prochain how-to, expliquer comment mettre des automatismes et des contrôles de validation à l'aide de macros dans les formulaires. Mais pour le moment, mes connaissances dans le domaine sont encore trop limitées. Donc si vous avez un peu de temps et l'envie de m'aider, n'hésitez pas à me contacter également.

## 14 Crédits

Auteur : **Tony GALMICHE** ( [tony.galmiche@tiscali.fr](mailto:tony.galmiche@tiscali.fr) ou [galmichetony@openoffice.org](mailto:galmichetony@openoffice.org) )

Remerciements :

- **Merci aux auteurs et aux traducteurs du how-to « 06 - Paramètres Nommés » qui m'a servi de base dans mon apprentissage sur les sous-formulaires.**
- **Merci à Christian HARDY, pour sa relecture.**
- **Merci à Sophie GAUTIER, pour sa relecture et ses tests.**
- **Un grand merci à Christophe FICHOT, pour sa relecture, ses tests, ses commentaires et ses nombreuses améliorations.**

Intégré par : **Sophie Gautier**

Dernière modification : **02/01/04**

Contacts : **Projet Documentation OpenOffice.org - [Fr.OpenOffice.org](http://fr.openoffice.org)**

Traduction :

## 15 Licence

### Appendix

#### Public Documentation License Notice

The contents of this Documentation are subject to the Public Documentation License Version 1.0 (the "License"); you may only use this Documentation if you comply with the terms of this License. A copy of the License is available at <http://www.openoffice.org/licenses/PDL.html>.

The Original Documentation is "**creation\_de\_sous\_formulaires.sxw**". The Initial Writer of the Original Documentation is **Tony GALMICHE** Copyright (C) **2003**. All Rights Reserved (Initial Writer contact: [tony.galmiche@tiscali.fr](mailto:tony.galmiche@tiscali.fr) or [galmichetony@openoffice.org](mailto:galmichetony@openoffice.org) ).

Contributor(s): \_\_\_\_\_.

Portions created by \_\_\_\_\_ are Copyright (C) \_\_\_\_\_ [Insert year(s)]. All Rights Reserved.  
(Contributor contact(s): \_\_\_\_\_ [Insert hyperlink/alias]).

NOTE: The text of this **Appendix** may differ slightly from the text of the notices in the files of the Original Documentation. You should use the text of this **Appendix** rather than the text found in the Original Documentation for Your Modifications.