LibreOffice

Guide Base

Chapitre 1
Introduction à Base

Droits d’auteur

Ce document est protégé par Copyright © 2020 par l’Équipe de Documentation de LibreOffice. Les contributeurs sont nommés ci-dessous. Vous pouvez le distribuer et/ou le modifier sous les termes de la Licence Publique Générale GNU (https://www.gnu.org/licenses/gpl.html), version 3 ou ultérieure, ou de la Licence Creative Commons Attribution (https://creativecommons.org/licenses/by/4.0/), version 4.0 ou ultérieure.

Toutes les marques déposées citées dans ce guide appartiennent à leurs légitimes propriétaires.

Contributeurs

De cette édition

Pulkit Krishna
Dan Lewis

Des éditions précédentes

Pulkit Krishna
Jean Hollis Weber
Dan Lewis
Peter Scholfield

Traduction

Jean-Michel Coste

Relecteurs

Philippe Clement
Patrick Auclair
Francis Lecher

Retour d’information

Veuillez adresser tout commentaire ou suggestion concernant ce document à la liste de diffusion de l’Équipe de Documentation : doc@fr.libreoffice.org

Note :

Tout ce que vous envoyez à la liste de diffusion, y compris votre adresse mail et toute autre information personnelle incluse dans le message, est archivé publiquement et ne peut pas être effacé.

Date de publication et version du logiciel

Publié en 2021. Basé sur LibreOffice .

Table des matières

Droits d’auteur....2
Contributeurs....2
De cette édition....2
Des éditions précédentes....2
Traduction....2
Relecteurs....2
Retour d'information....2
Introduction....4
Base - un conteneur pour les éléments de la base de données....4
Saisie de données à l'aide de formulaires....6
Entrée directe de données dans une table - principes de base pour la saisie de données....7
Requêtes - obtenir des informations à partir de données dans des tables....10
Rapports - présentation des données....11
Manipulation sûre d'un fichier Base....13
Une base de données simple - exemple de test en détail....13
Créer des tables....14
Créer un formulaire de saisie de données....22
Tabulation dans le sous-formulaire....30
Activer la barre de navigation du formulaire principal dans le sous-formulaire....31
Restreindre la saisie dans un contrôle....31
Créer une requête....33
Créer un rapport....38
Définition des distances entre les champs du rapport....43
Influencer le contenu d'un champ de texte par une formule....43
Modifier la mise en forme d'un champ de texte....44
Déplacement des boîtes dans le générateur de rapports....44
Extensions de la base de données exemple....45

Introduction

Dans les opérations quotidiennes de bureau, les feuilles de calcul sont régulièrement utilisées pour agréger des ensembles de données et pour effectuer des analyses sur ces derniers. Comme les données d'une feuille de calcul sont présentées dans une vue de tableau, clairement visibles et peuvent être modifiées ou ajoutées, de nombreux utilisateurs se demandent pourquoi ils devraient utiliser une base de données au lieu d'une feuille de calcul. Ce manuel explique les différences entre les deux.

Note

Dans le langage technique, le "Fichier base de données" est utilisé pour une base de données à partir d'une seule interface et le "système de base de données" englobe le système de gestion de base de données (SGBD) et la base de données proprement dite.

Base offre un accès à divers systèmes de base de données via une interface utilisateur graphique et fonctionne par défaut avec le moteur de base de données intégré HSQLDB.

Ce chapitre présente deux exemples de bases de données et tout ce manuel est construit autour d'eux : Media_sans_macros.odb et Media_avec_macros.odb, étendus avec l'inclusion de macros. Les deux bases de données sont destinées à l'exploitation d'une bibliothèque : acquisition de média, emprunt par l'utilisateur, location de média et tout ce qui y est lié, comme le rappel pour les lecteurs.

Un exemple plus détaillé est donné plus loin dans ce chapitre (à partir de la page 13). Il utilise la base de données Exemple_Sport.odb pour organiser une compétition sportive.

Note

Comme tout logiciel, LibreOffice Base n'est pas totalement exempt d'erreurs. Particulièrement ennuyeuses sont les régressions, qui réintroduisent un bogue d'une ancienne version dans la version actuelle. Le lien suivant mène aux régressions actuellement en suspens :

https://bugs.documentfoundation.org/buglist.cgi?bug_status=UNCONFIRMED&bug_status=NEW&bug_status=REOPENED&bug_status=VERIFIED&bug_status=NEEDINFO&component=Base&keywords=regression&keywords_type=allwords&list_id=1162928&product=LibreOffice&query_format=advanced

Un regard sur la liste des bogues peut donc vous aider à comprendre les différences entre la documentation et sa propre version du programme.

Base – un conteneur pour les éléments de la base de données

Un fichier Base est un dossier compressé contenant des informations sur les différentes zones de travail (composants) de Base. En utilisation quotidienne, Base s'ouvre initialement avec la vue illustrée dans la Figure 1.

L'environnement de base contient quatre zones de travail : Tables, Requêtes, Formulaires et Rapports. En fonction de la zone sélectionnée, diverses tâches (création de nouveau contenu ou appel d'éléments existants) peuvent être effectuées.

Dans les zones de travail Formulaires et Rapports, les éléments respectifs sont organisés dans une structure de dossiers (Figure 2). Cela se fait soit directement lors de l'enregistrement dans la boîte de dialogue Enregistrer, soit par la création de nouveaux dossiers en utilisant **Insertion > Dossier**

Bien que le fondement d'une base de données soit constituée de tables, similaires à des tableaux, Base commence par le mode Formulaire, car ils sont les éléments les plus couramment utilisés lors de l'utilisation de bases de données. Avec les formulaires, vous pouvez effectuer des entrées dans les tableaux et analyser le contenu des tableaux.

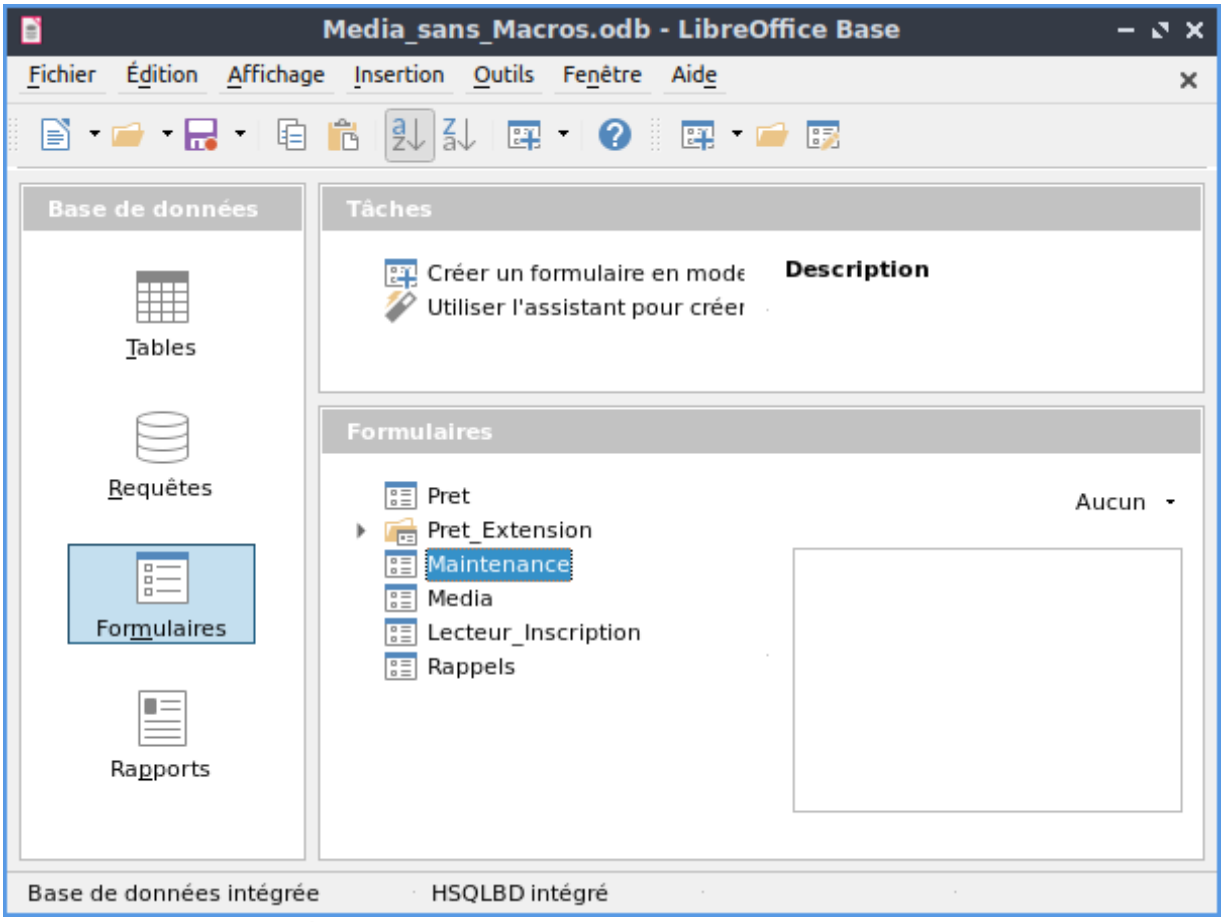


Figure 1: Vue de Base à l'ouverture

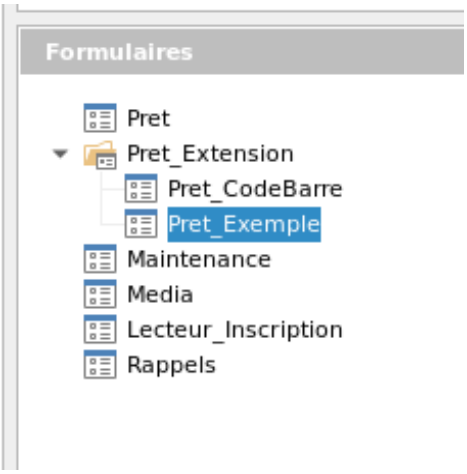


Figure 2: Structure de répertoire dans la zone de travail formulaires

Saisie de données à l’aide de formulaires

Les formulaires simples affichent un seul tableau, comme dans la partie supérieure du formulaire de prêt. Le formulaire de prêt (Figure 3) a été étendu pour afficher des informations supplémentaires : L’éventail des personnes affichées peut être filtré sur le nom de famille pour limiter les détails affichés. Si un utilisateur entre la lettre “G” dans le champ Filtre (nom de famille) à droite du tableau des prêts, seules les personnes dont le nom commence par “G” seront affichées[1].

- Les informations sur les nouveaux emprunteurs peuvent être saisies directement dans les champs du tableau du formulaire.
- Les détails des éléments à emprunter sont saisis et affichés dans la zone centrale du formulaire. Le nom de l’utilisateur est également clairement souligné. Si un article déjà emprunté est en retard et doit être retourné, cette zone est bloquée (aucune saisie possible) et le titre indiquera “Prêt temporairement verrouillé !”. Les articles en prêt sont affichés dans la partie inférieure du formulaire.
- La date d’emprunt est définie comme date du jour. Dans le champ déroulant à gauche du bouton Actualiser se trouvent les éléments multimédias qui peuvent être empruntés. Les articles déjà prêtés à l’emprunteur sélectionné ne peuvent pas être sélectionnés.
- Les éléments multimédias sélectionnés pour le prêt sont ajoutés aux détails du prêt en cours en cliquant sur le bouton Actualiser.
- Dans la partie inférieure du formulaire (Retours), il n’est pas possible de supprimer une ligne de données. Seuls les champs Date de retour et Extension peuvent être modifiés. Si un emprunteur a été précédemment verrouillé et a ensuite renvoyé le ou les articles en retard, la zone de prêt peut être déverrouillée en cliquant sur le bouton Actualiser.

Toutes ces fonctions peuvent être effectuées sans utiliser de macros, lorsque le formulaire est configuré et rempli de la manière décrite.

1 Ce formulaire est visible dans le fichier exemple Media_sans_Macros.odb

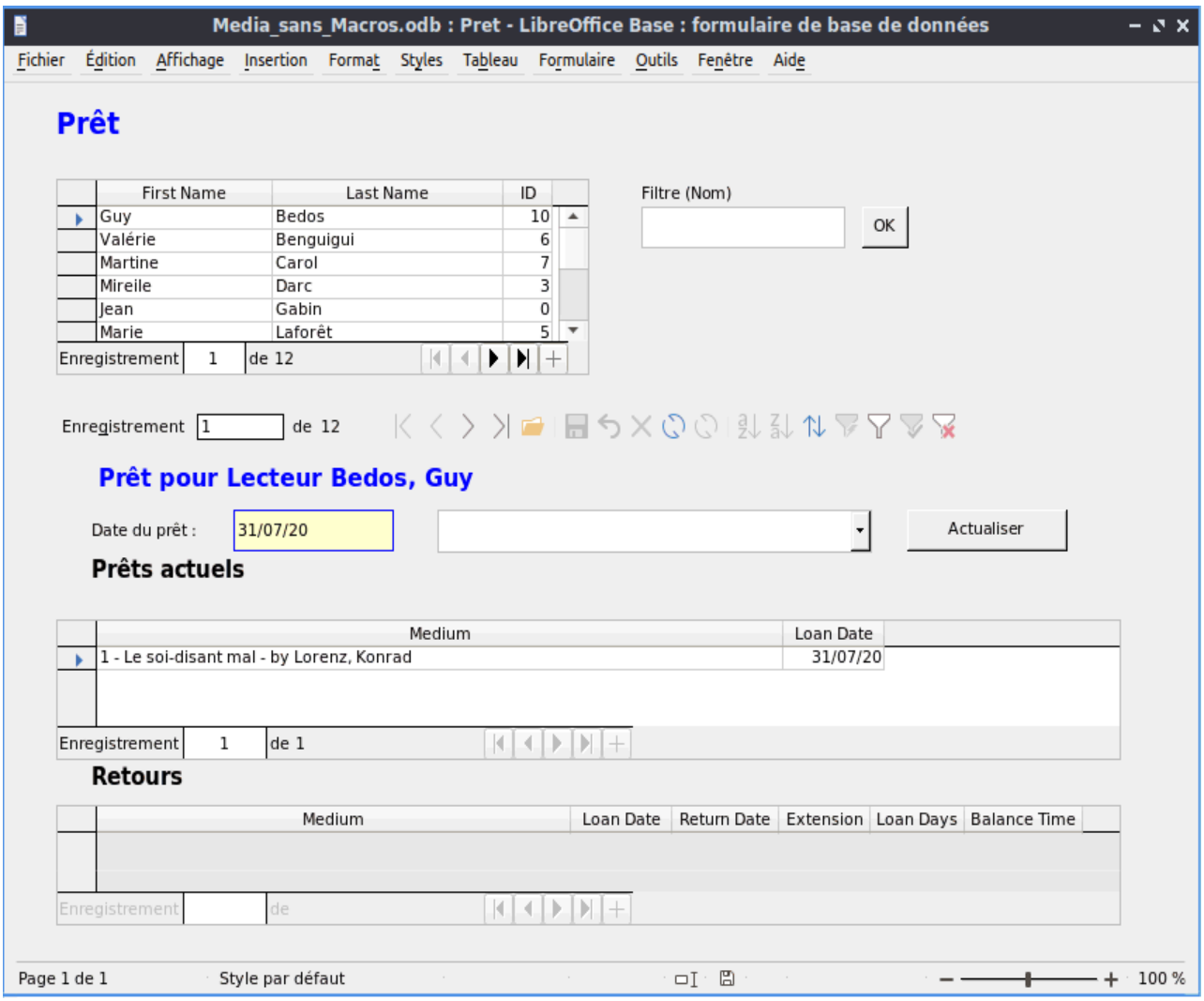


Figure 3: Le formulaire de prêt

Entrée directe de données dans une table – principes de base pour la saisie de données

Les tables d'une base de données sont liées un peu comme dans un réseau. Une table reçoit des informations d'autres tables ou leur en fournit. Ceci est appelé la relation et est indiqué par une ligne entre les tables reliant un champ de chacune. Ces relations sont visibles dans **Outils>Relations**.

Note

La table Lecteurs a une relation avec une autre table qui implique ID_Genre. De même, la table Medias a une relation avec quatre autres tables, chacune impliquant l'un de ces champs : ID_Categorie, ID_TypeMedia, ID_Ville et ID_Editeur.

La table des prêts est directement liée aux tables Media et Lecteurs, comme illustré à la figure 4.

Lorsqu'un livre est emprunté, au lieu que son titre soit enregistré dans la table Prets, un seul numéro est enregistré dans le champ ID_Media. Le champ ID de la table Media stocke l'identifiant unique de chaque enregistrement de cette table. Ce champ est un champ clé de la table Media : la clé primaire.

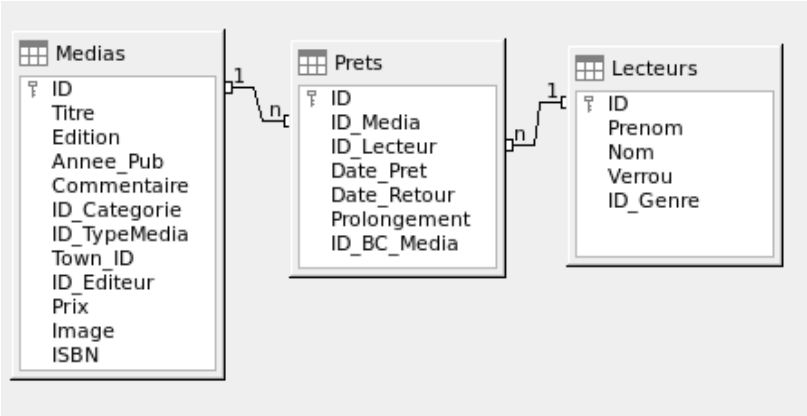


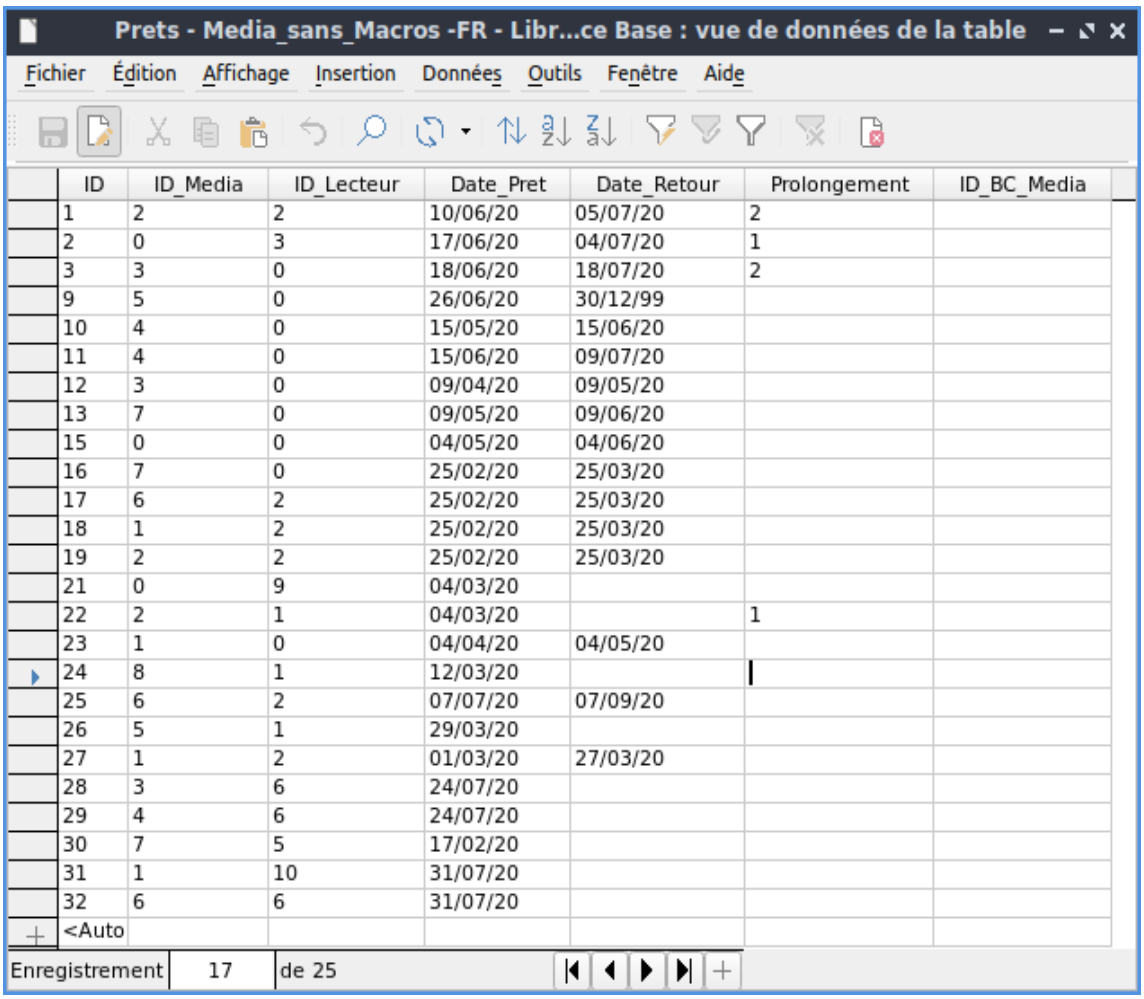
Figure 4: Relation entre la table Prets et la table Lecteurs

Conseil

La clé primaire détermine de manière unique les valeurs de chaque champ dans chaque enregistrement d'une table. Ainsi, lorsqu'un élément est emprunté, le numéro entré dans le champ ID-Media correspond au numéro du champ ID de la table Media qui identifie l'enregistrement contenant les informations sur l'élément emprunté.

Le nom du lecteur n'est pas saisi à chaque fois dans la table des prêts. Ces informations sont enregistrées dans la table Lecteurs. Il a également un champ de clé primaire qui identifie chaque personne qui emprunte un article. La valeur de ce champ peut ensuite être entrée dans la table Prets avec le champ ID_Lecteur identifiant la personne spécifique.

Les relations entre les tables présentent l'avantage de réduire considérablement le travail de bureau sur le formulaire. Au lieu d'avoir à entrer le titre du média ainsi que le prénom et le nom sans aucune erreur, ceux-ci peuvent être entrés en sélectionnant les bons numéros pour les champs ID_Media et ID_Lecteur, ce qui permet de sélectionner les bons éléments multimédias ainsi que les nom et prénom. Enfin, le même support peut être emprunté à nouveau plus tard et le même lecteur peut emprunter plusieurs supports supplémentaires lors de toute action de prêt.



ID	ID_Media	ID_Lecteur	Date_Pret	Date_Retour	Prolongement	ID_BC_Media
1	2	2	10/06/20	05/07/20	2	
2	0	3	17/06/20	04/07/20	1	
3	3	0	18/06/20	18/07/20	2	
9	5	0	26/06/20	30/12/99		
10	4	0	15/05/20	15/06/20		
11	4	0	15/06/20	09/07/20		
12	3	0	09/04/20	09/05/20		
13	7	0	09/05/20	09/06/20		
15	0	0	04/05/20	04/06/20		
16	7	0	25/02/20	25/03/20		
17	6	2	25/02/20	25/03/20		
18	1	2	25/02/20	25/03/20		
19	2	2	25/02/20	25/03/20		
21	0	9	04/03/20			
22	2	1	04/03/20		1	
23	1	0	04/04/20	04/05/20		
24	8	1	12/03/20			
25	6	2	07/07/20	07/09/20		
26	5	1	29/03/20			
27	1	2	01/03/20	27/03/20		
28	3	6	24/07/20			
29	4	6	24/07/20			
30	7	5	17/02/20			
31	1	10	31/07/20			
32	6	6	31/07/20			
<Auto						

Figure 5: Table Prets, structure et données

La structure de table pour un tel formulaire est relativement basique et facile à mettre en place. Dans le tableau illustré à la Figure 5, (table prêt) les mêmes données peuvent être entrées directement dans les lignes et les colonnes de la table que lors de l'utilisation du formulaire, à condition de connaître les ID Media et lecteur correspondants. Les relations de cette table avec les autres tables de la base de données sont utilisées dans le formulaire.

- Le champ le plus important, la clé primaire (ID) qui est ajoutée automatiquement, montre le contenu unique et indispensable pour la plupart des bases de données. Pour plus d'informations sur ce sujet, reportez-vous à la section "Relations entre les tables" du chapitre 3, Tables.
- Le deuxième champ, ID_Media, stocke les valeurs de la clé primaire de la table Medias. Il fait référence au numéro dans le champ correspondant, ID, dans la table Media. Une telle référence à une clé primaire est appelée clé étrangère. Dans le formulaire, le titre et l'auteur seront affichés à la place de la clé étrangère dans une zone de liste. La zone de liste transmet la valeur en arrière-plan à la clé étrangère de la table.
- Le troisième champ, ID_Lecteur, stocke les valeurs de clé primaire de la table Lecteurs. Dans cet exemple, cette clé est uniquement un nombre qui fait référence au lecteur qui emprunte des éléments multimédias. Dans le formulaire, le nom et le prénom du lecteur sont indiqués. Comme le montre le tableau, le lecteur avec le numéro de clé primaire "0" a emprunté beaucoup de supports. La table peut enregistrer plusieurs fois la clé primaire unique de la table Lecteurs en tant que clé étrangère ID_Lecteur. Mais en aucun cas un lecteur, qui est répertorié dans la clé étrangère de la table Prets, ne peut être supprimé dans la table Lecteur. Sinon, il ne serait plus possible de savoir qui a emprunté des médias à un instant "t". La base de données définit les paramètres par défaut de sorte qu'une

suppression est impossible. Le terme technique pour cela est l'exigence d'intégrité référentielle.

- La date du prêt est stockée dans le quatrième champ. Si cette date est présente et est postérieure à la date actuelle, l'ensemble des données correspondant pour le lecteur est affiché dans le tableau du bas du formulaire sous le bouton Retour. Le prêt est différé.
- Le champ Prolongement contient des informations sur les extensions du prêt pour un article. La signification des valeurs 1, 2… est détaillée ci-après. La base de données contient une table séparée avec le nom Preferences.

L'entrée de ces données permet la gestion d'une bibliothèque simple.

Requêtes – obtenir des informations à partir de données dans des tables

Requêtes affiche une vue des tables. Elles rassemblent le contenu de plusieurs tables dans une vue d'ensemble. Les requêtes sont stockées uniquement dans le langage de requête SQL. Ce ne sont donc pas des tables, même si dans Base elles nous semblent apparaître comme des tables.

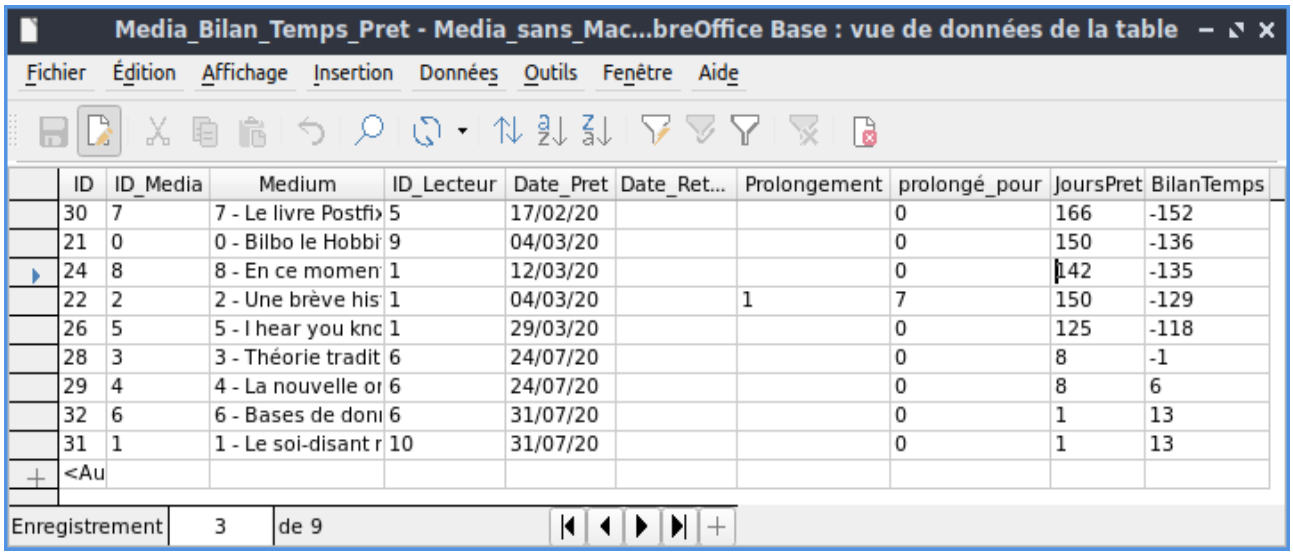


Figure 6: Exemple de requête

La requête illustrée à la figure 6 répertorie tous les médias actuellement prêtés. Elle calcule la durée du prêt de chaque article et le solde de la période de prêt. Lorsque ID_Media, le champ de clé étrangère, lit la clé primaire de la table Medias, le titre et l'auteur seront combinés en un seul texte dans le champ Medium. Ce champ sera nécessaire dans le formulaire sous le sous-titre "Retour". Les champs combinés de la requête servent également de champs de connexion entre le formulaire de prêt réel et la table de prêt, à savoir via les champs ID_ Media et ID_Lecteur.

- Tous les supports sont répertoriés pour lesquels la date de retour n'est pas entrée dans la table Prets. À titre de présentation supplémentaire, le nom du média est inclus dans la requête avec le champ ID_Media.
- La référence au lecteur est établie avec la clé primaire de la table Lecteurs.
- La différence en jours entre la date du prêt (Date_Pret) et la date actuelle est spécifiée sous la forme de JoursPret
- Le nombre de jours de prêt est soustrait de la date du jour pour donner le nombre de jours restants dans la période de prêt. La durée du prêt peut varier selon les types de supports.
- Dans la table Parametres, une valeur de "1" pour le champ Prolongement correspond à une extension de la période de prêt de 7 jours. Dans l'ensemble de données ci-dessus, la ligne avec ID_Media '2' montre un prolongement de 7 jours.

Rapports – présentation des données

Dans les rapports, les données sont traitées de manière à pouvoir être imprimées dans un format utile. Les formulaires tels que celui de la figure 7 ne conviennent pas pour une lettre convenablement formatée.

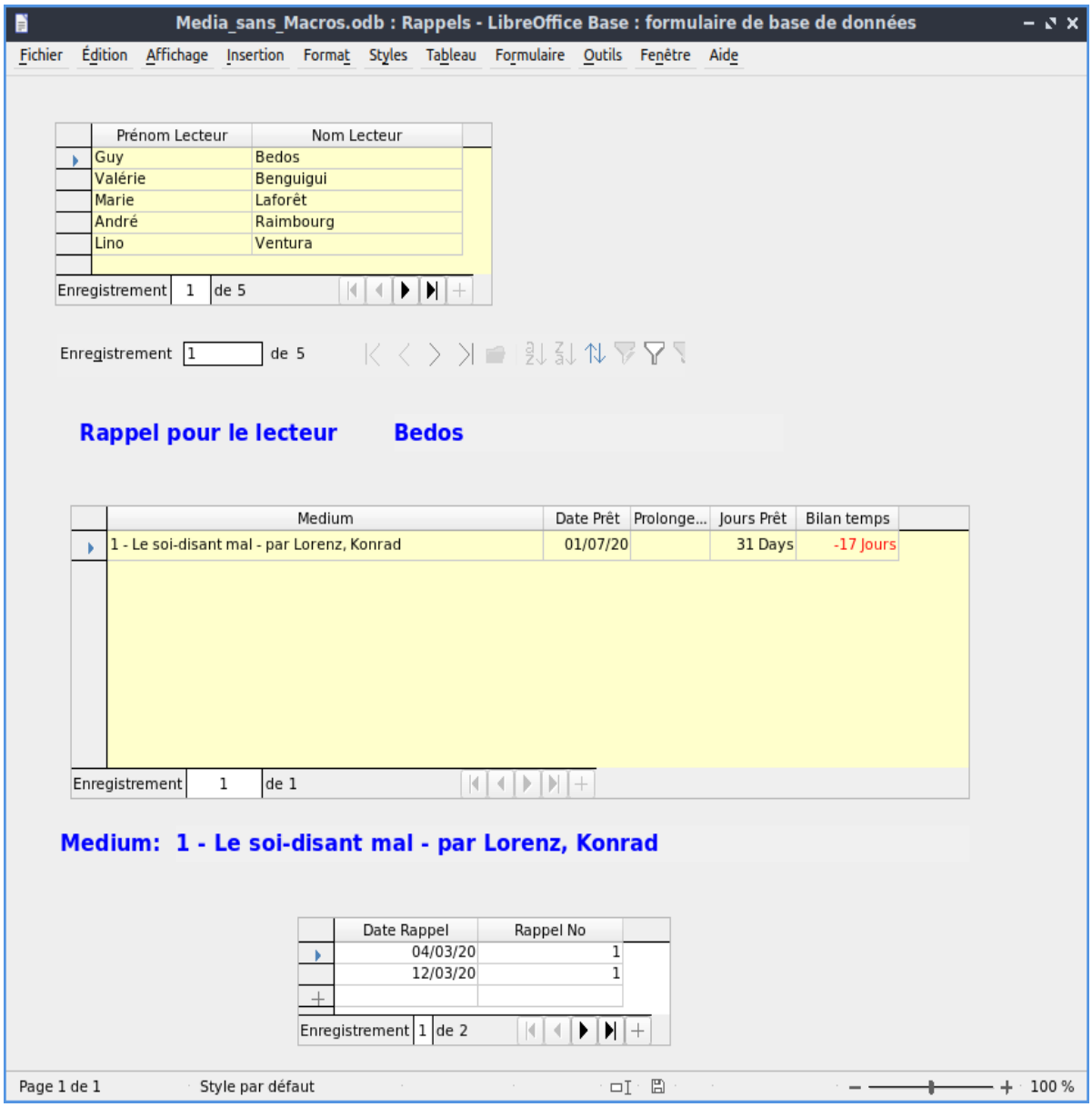


Figure 7: Formulaire contenant les informations pour un avis de rappel

Avant qu'un rapport réel, sous la forme d'un avis de rappel, puisse être imprimé, les informations doivent être saisies dans le formulaire de rappel. Le tableau du formulaire montre toutes les personnes qui ont emprunté des articles avec une durée de prêt résiduelle négative.

La date de rappel et le numéro de l'avis de rappel sont saisis pour chaque élément multimédia à rappeler. Il peut s'agir de la date actuelle de traitement des avertissements. La date de rappel est par défaut la date actuelle. Le numéro de rappel est un entier incrémenté de 1 à chaque avis de rappel successif pour un emprunt/média particulier.

Ce formulaire, dans l'exemple de base de données actuel sans macros, nécessite l'entrée de l'utilisateur pour créer des avis de rappel. Dans la version macro, la date est automatiquement saisie et l'avis de rappel imprimé.
L'avis de rappel (Figure 8) est généré au moyen d'une requête à partir des données

précédemment entrées. L'utilisateur de la base de données n'a qu'à sélectionner le rapport de rappel et une lettre de rappel peut être imprimée et envoyée à toutes les personnes qui ont une entrée de rappel effectuée dans le formulaire de la page précédente.

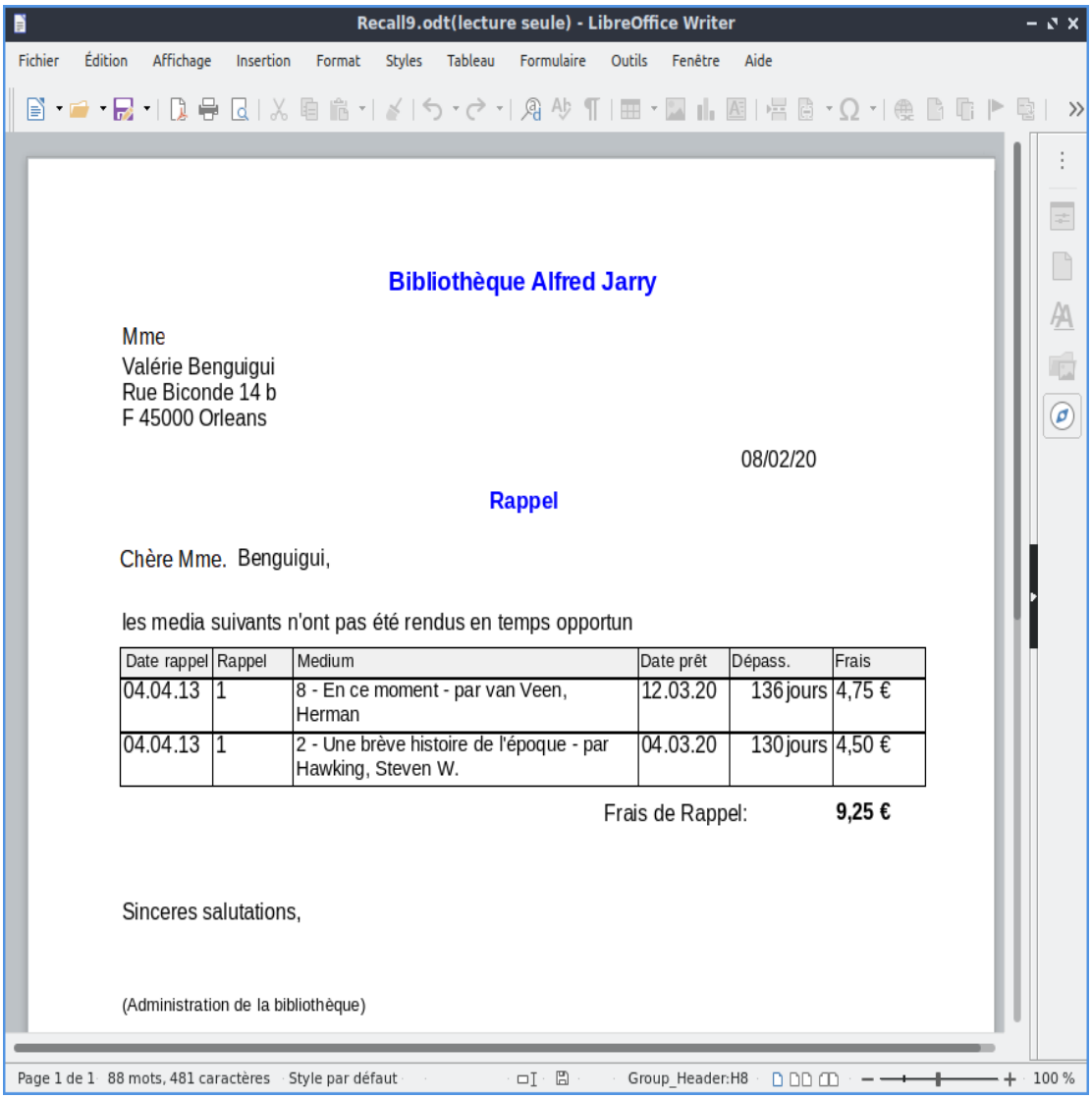


Figure 8: Exemple d'avis de rappel

Dans un tel rapport, il peut y avoir plusieurs entrées (éléments en retard) pour une personne en particulier. Si le tableau contenant les éléments pour cette personne dépasse l'espace d'une page, il est étendu pour inclure une page suivante.

Un tel rapport est plus complet qu'une lettre de publipostage produite avec Writer. Il rassemble automatiquement les ensembles de données pour l'impression et organise le texte d'accompagnement nécessaire en conséquence.

Une lettre similaire à celle de la figure ci-dessus ne peut être implémentée qu'avec des macros, ou le générateur de rapports, comme décrit dans "Créer un rapport" à la page 39.

Manipulation sûre d'un fichier Base

Les tables, requêtes, formulaires et rapports de la base de données interne HSQLDB sont stockés dans un fichier Base. Étant donné que le fichier de base de données est écrit en mémoire, les multiples objets qu'il contient vous obligent à le traiter avec soin. Les rapports de bogue indiquent clairement qu'un fichier de base de données nécessite un traitement un peu plus soigneux que, par exemple, un fichier texte écrit dans Writer.

Les instructions suivantes doivent donc être prises en compte lors du traitement d'un fichier de base :

- Un fichier de base de données ouvert ne doit pas être enregistré sous un nom différent en utilisant **Enregistrer sous**. Lorsqu'il n'y a pas d'autre choix, les tables, requêtes, formulaires et états doivent d'abord être fermés. Il est préférable de fermer le fichier de base de données et de créer ensuite une copie du fichier.
- Le Générateur de rapports est un module complémentaire. Bien qu'il ne soit désormais plus visible en tant qu'extension distincte, il fonctionne en grande partie indépendamment du fichier de base de données. Renommer le fichier supprime le Générateur de rapports de sa fondation.
- Lorsqu'une table, une requête, un formulaire ou un rapport est enregistré, il ne s'ensuit pas que le fichier de base de données entier a été enregistré. Cette sauvegarde doit être effectuée séparément. Lorsqu'un objet (table, requête, formulaire, rapport) est enregistré, les informations sont écrites dans le fichier de base de données en mémoire. Lorsque le fichier de base de données est enregistré, tout ce qui est contenu dans le fichier de base de données en mémoire est écrit dans le fichier *.odb.
- Ce comportement de mémoire est particulièrement vrai pour l'utilisation du Générateur de rapports. La préparation d'un rapport est toujours le composant le plus instable du fichier Base. Par conséquent, après chaque étape, le rapport et le fichier *.odb doivent tous deux être enregistrés. Une fois le rapport créé, il fonctionne par lui-même sans problème particulier.
- Une fois qu'une action est terminée, les données ajoutées à la base de données sont uniquement écrites dans le fichier de base de données en mémoire, mais pas dans le fichier *.odb. Ce n'est que lorsque vous fermez le fichier *.odb que vous y enregistrez les données. Le contenu de la base de données HSQLDB sera réécrit dans le fichier. Un crash à ce stade peut entraîner une perte de données. Par conséquent, une stratégie doit être développée pour que les copies de sauvegarde soient effectuées à temps. Le chapitre 9, Macros, comprend une macro pour effectuer une copie de sauvegarde lorsque vous ouvrez un fichier de base de données. De même, un moyen est montré pendant que le fichier Base est ouvert aussi bien que possible.

  Un niveau de sécurité nettement plus élevé peut enfin être atteint en utilisant des bases de données de serveurs externes comme MySQL / MariaDB ou PostgreSQL. Pour cela, Base peut alors servir de frontal avec les requêtes, formulaires et états de la base de données.

Une base de données simple – exemple de test en détail

La création d'une base de données est abordée au chapitre 2, Création d'une base de données. L'exemple suivant est basé sur la base de données par défaut HSQLDB qui est installée avec LibreOffice en tant que base de données interne. Il s'agit donc d'une base de données embarquée créée au préalable sans aucune inscription dans LibreOffice. Divers autres systèmes de bases de données peuvent être connectés en plus de la HSQLDB interne.
La première étape à faire après avoir trouvé un emplacement pour le fichier de base de données et l'avoir enregistré consiste à fermer l'assistant Base de données en cliquant sur le bouton **Terminer**[2].

La base de données est destinée à organiser une compétition sportive dans différentes disciplines. Par conséquent, Exemple_Sport a été choisi comme nom du fichier.

2 Cet exemple est réalisé dans le fichier Exemple_Sport.odb, disponible avec ce guide

Créer des tables

Dès que la base de données a été enregistrée, la fenêtre principale de la base de données apparaît. Par défaut, Tables est sélectionné dans la section Base de données sur le côté gauche de la fenêtre (Figure 9). Les tables sont le stockage central des données ; sans tables, il n'y a pas de base de données.

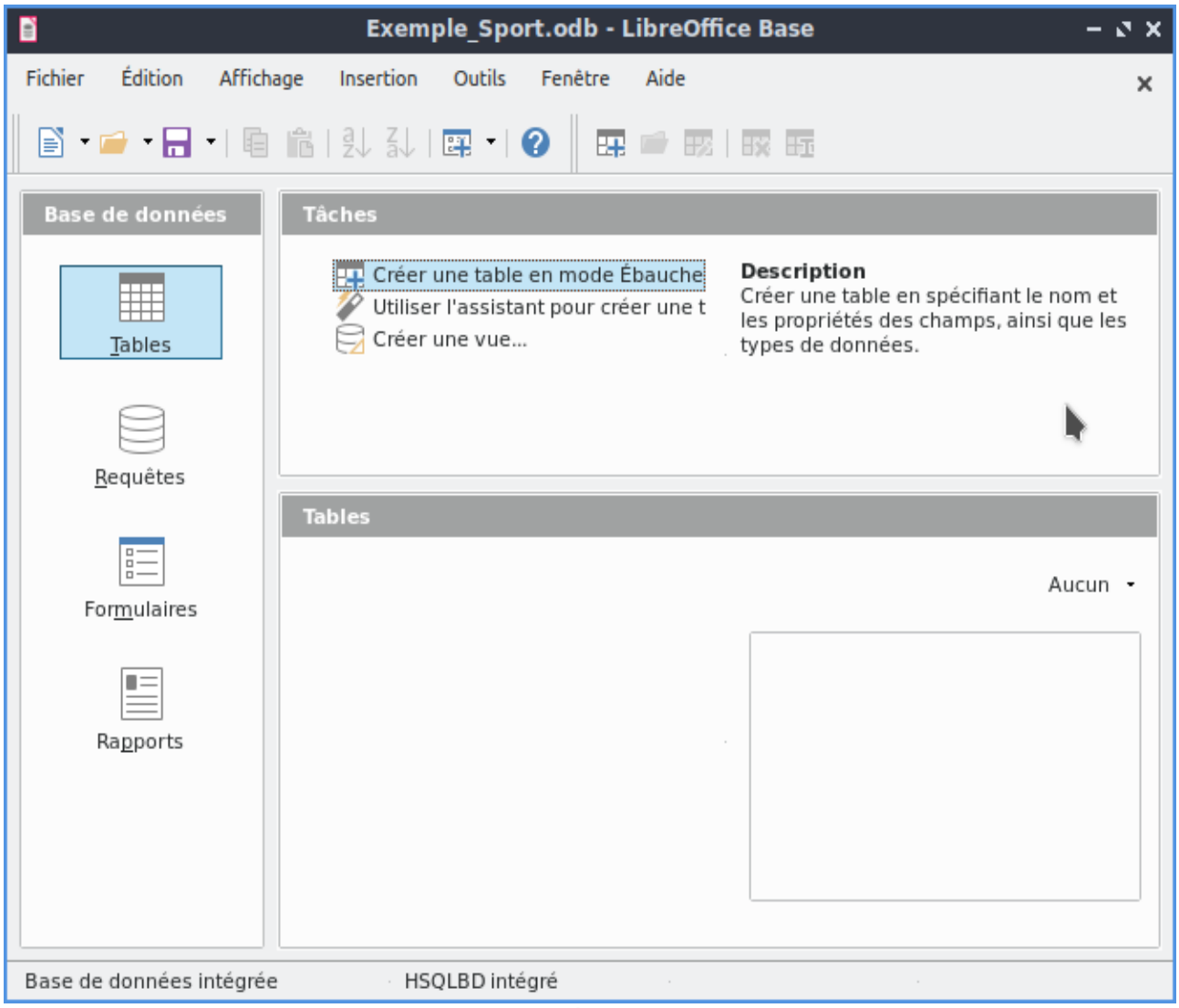


Figure 9: Fenêtre principale de la base de données exemple Exemple_Sport.odb

Cliquez sur **Créer une table en mode Ébauche** pour ouvrir la fenêtre illustrée à la Figure 10.

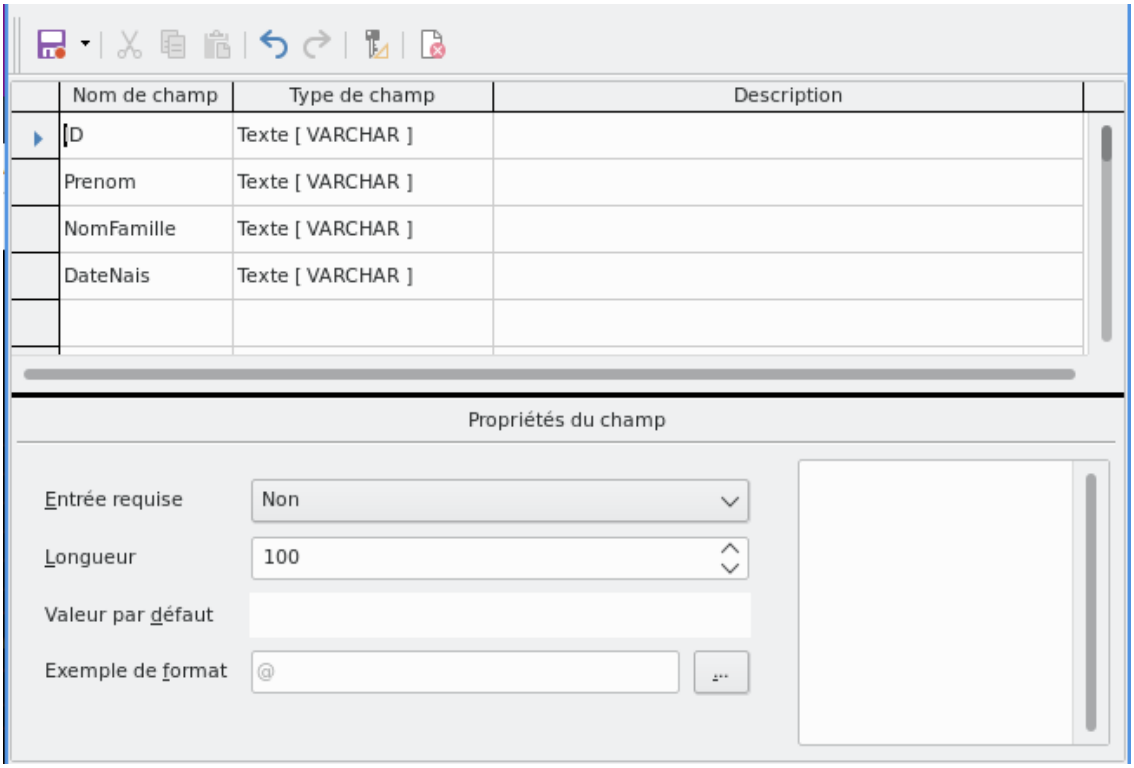
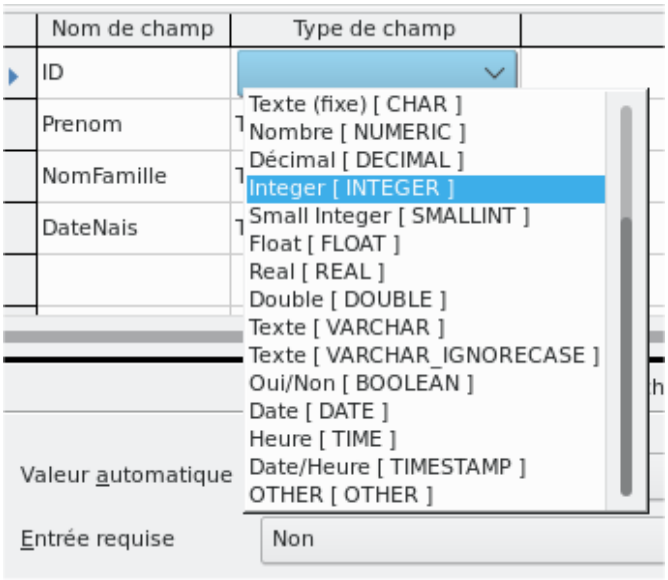


Figure 10: Vue en mode ébauche d'une table

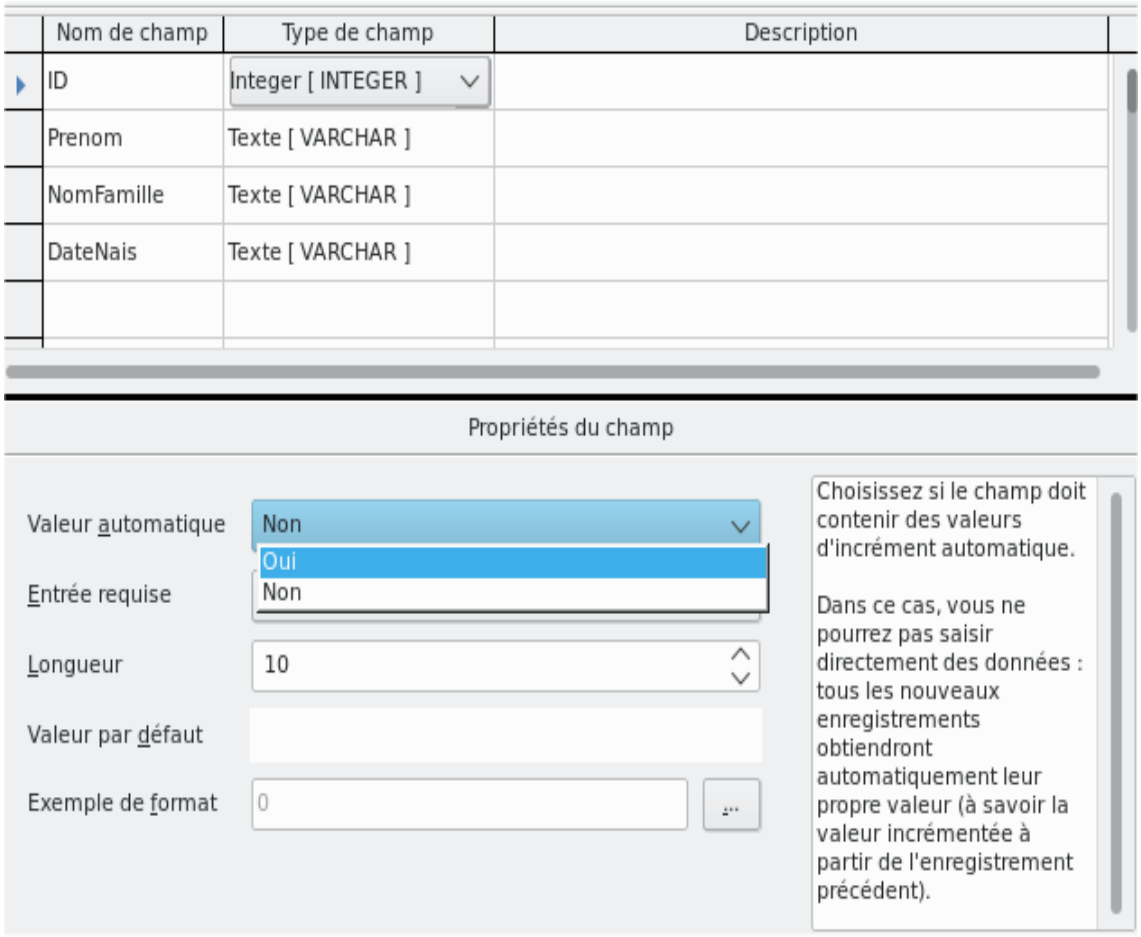
Les noms des champs de la première table seront saisis ici. Le tableau devrait inclure les candidats hommes et femmes. Ici, les noms de champ sont d'abord réduits aux composants clés.

Les noms de champ prénom (Prenom), nom (NomFamille) et date de naissance (DateNais) sont susceptibles d'être sans ambiguïté. De plus, un champ appelé ID a été ajouté. Ce champ prendra plus tard une valeur unique pour chaque enregistrement. Un champ clé unique est nécessaire pour la base de données intégrée. Sinon, aucun enregistrement ne peut être saisi dans la table. Ce champ de clé est appelé *clé primaire* dans les bases de données.

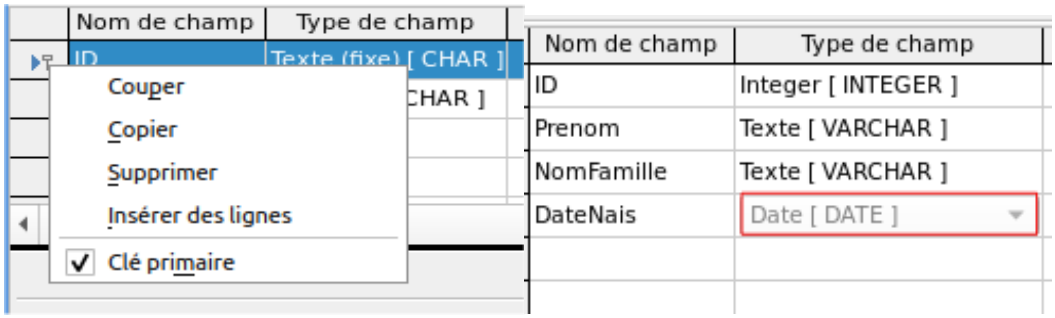
Un autre champ pourrait être utilisé pour cette propriété. Cependant, si par exemple le nom de famille était utilisé seul pour cela, deux personnes portant le même nom de famille ne pouvaient pas être enregistrées. Dans ce cas, il peut être utile de déclarer deux champs ensembles dans une clé primaire partagée. Il n'y a aucune garantie que cela fonctionne à long terme, ce qui n'est pas le cas. Ici, la version simple est préférée.



Dans la deuxième étape, sélectionnez les types de champs pour les champs déjà nommés dans les listes. Définissez le champ ID sur le type de champ Integer. Ce type de champ a l'avantage de pouvoir être automatiquement fourni par la HSQLDB intégrée avec l'entier supérieur suivant à chaque nouvel enregistrement.



Modifiez les propriétés de champ ID. Pour ce champ, activez l'incrémentation automatique des valeurs numériques croissantes : **Propriétés du champ > Valeur automatique > Oui**.



Après avoir sélectionné **Valeur automatique**, ou indiqué que ce champ était une clé primaire,une icône en forme de clé apparaît sur l'en-tête de ligne lorsque vous quittez la sélection du type de champ. Cela indique que ce champ est la clé primaire de la table. Si **Valeur automatique** n'est pas sélectionné, la clé primaire peut également être sélectionnée dans le menu contextuel (clic droit> clé primaire).

Sélectionnez le type de champ Date pour **DateNais** . Cela garantit que seules les entrées de date valides sont ajoutées. Il est également utilisé pour trier les dates ou, par exemple, calculer l'âge.

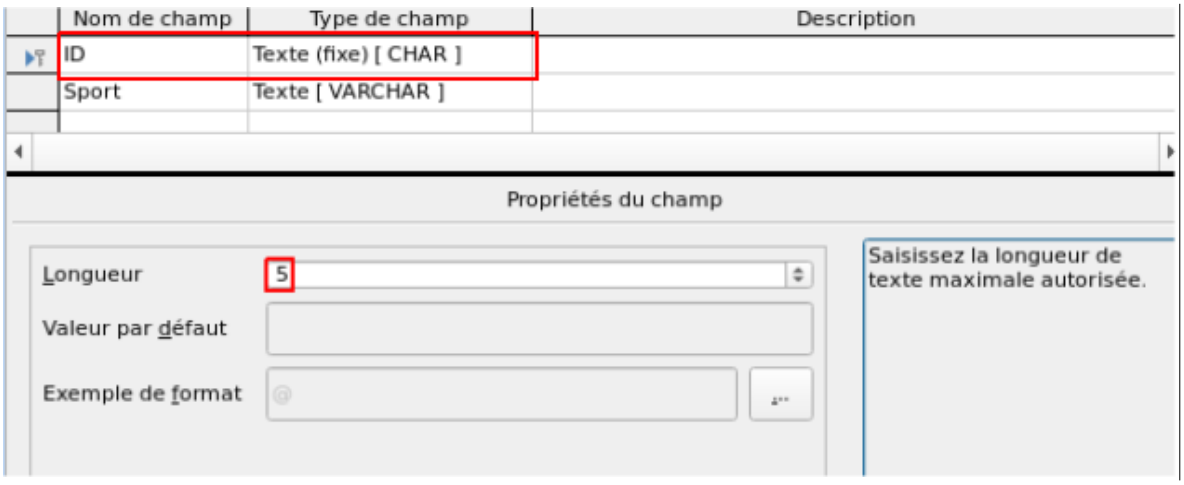
La table peut maintenant être enregistrée sous le nom **Candidats**. Par la suite, les données peuvent être saisies. La saisie dans le champ ID n'est pas nécessaire. Cela se fait automatiquement lorsque vous sauvegardez l'enregistrement

Note

Le fichier de base de données est un dossier compressé de fichiers individuels. Le stockage d'un seul objet comme table n'est donc pas directement écrit dans le fichier de base de données lui-même. C'est pourquoi il faut cliquer sur le bouton Enregistrer du fichier de base de données lui-même, même après la création de tables, requêtes, formulaires et rapports. Ce n'est qu'en quittant la ligne de données d'une table que les données saisies sont enregistrées automatiquement.

Les candidats (athlètes) peuvent désormais être inscrits. Cependant, les informations suivantes manquent à première vue :

- Une liste de sports dans lesquels les candidats veulent concourir.
- Pour les compétitions, la distinction entre les candidats masculins et féminins.

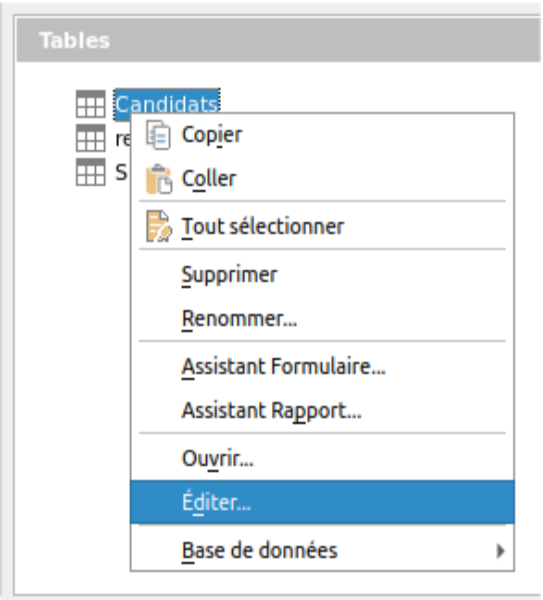


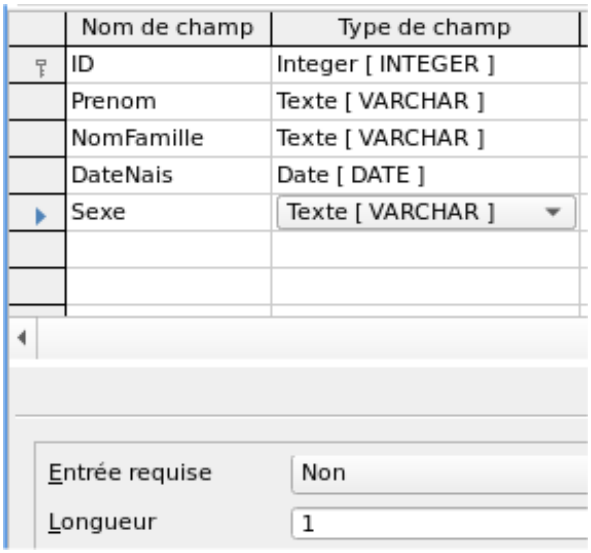
Créez une table Sports avec 2 champs comme indiqué sur la figure. Comme il n'y a pas beaucoup de sports différents, Valeur automatique n'est pas sélectionné pour la clé primaire. Au lieu de cela, le type de champ (champ ID) est laissé sous forme de texte, mais limité à 5 caractères. Les 5 caractères suffisent, afin de trouver une abréviation appropriée pour les différents sports.

Ouvrez à nouveau la table Candidats pour la modifier (**Éditer**), à l'aide du menu contextuel de la table.

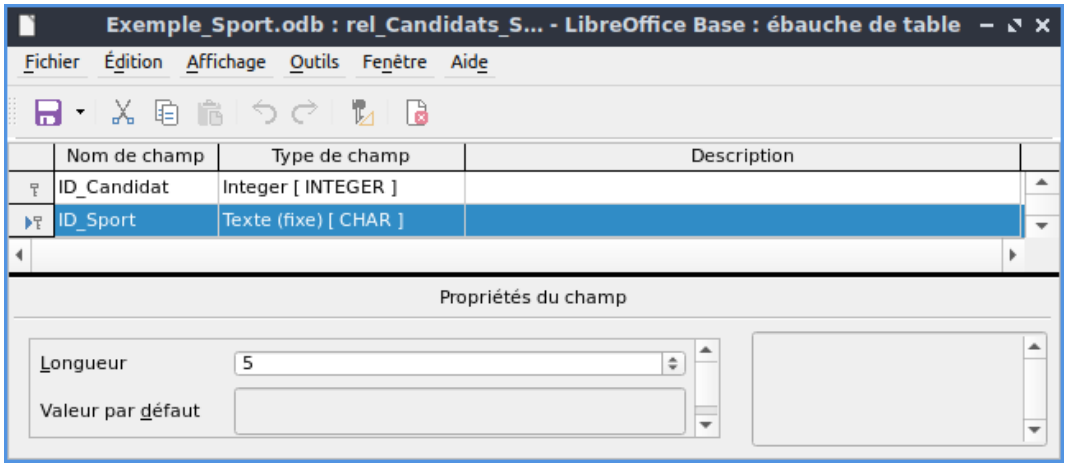
Ajoutez le champ "Sexe" à la table. Un nouveau champ peut être ajouté à la boîte de dialogue de conception de la table, uniquement à la fin de la définition de la table. Il est également possible d'utiliser SQL pour ajouter de nouveaux champs à certaines positions.
La longueur du texte dans ce champ est limitée à un caractère, suffisant pour "m" et "f" comme entrée.

D'une manière ou d'une autre, les deux tables doivent être liées afin que chaque Candidat puisse être inscrit dans plusieurs sports et que plus de candidats puissent être inscrits pour n'importe quel sport. Cela se fait via une table dans laquelle les valeurs des deux clés primaires des tables Candidats et Sport sont enregistrées. Étant donné que seule la combinaison de ces champs sera enregistrée ensemble, ces champs sont la clé primaire de cette table.



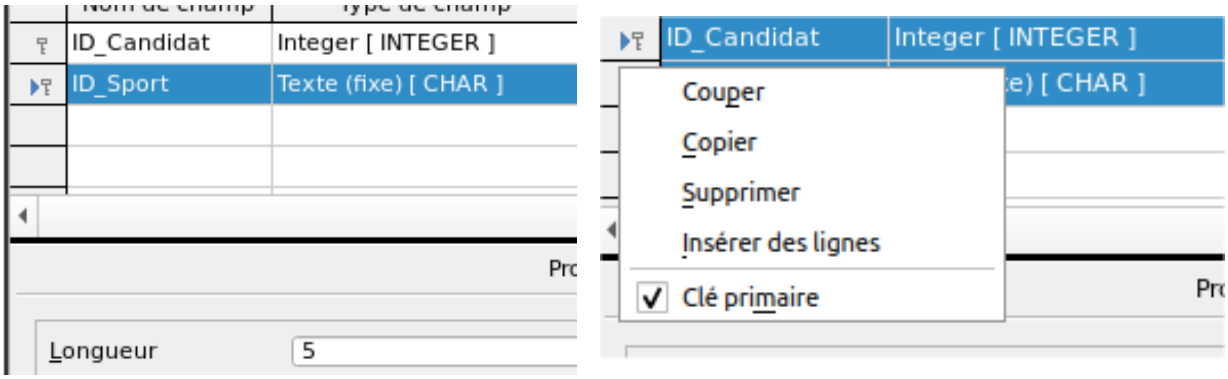


Nous allons créer cette table sous le nom rel_Candidat_Sport, avec deux champs, ID_Candidat et ID_Sport, ***du même type*** que les champs ID des tables Candidats et Sports.



Les valeurs appropriées peuvent être extraites de Candidats et Sports, car les champs doivent correspondre exactement aux types de champs que vous souhaitez enregistrer. ID_Candidat doit donc avoir le type de champ Integer. ID_sport a le type de champ Texte et est également limité à 5 caractères, comme le champ ID de la table Sport.

Pour attribuer la clé primaire aux deux champs, cliquez sur l'en-tête de ligne du premier champ, puis sur Maj - clic sur l'en-tête de ligne du deuxième champ ; ceci sélectionne les deux champs. Cliquez avec le bouton droit sur l'un des en-têtes de ligne, puis cliquez sur Clé primaire dans le menu contextuel pour spécifier la clé primaire.



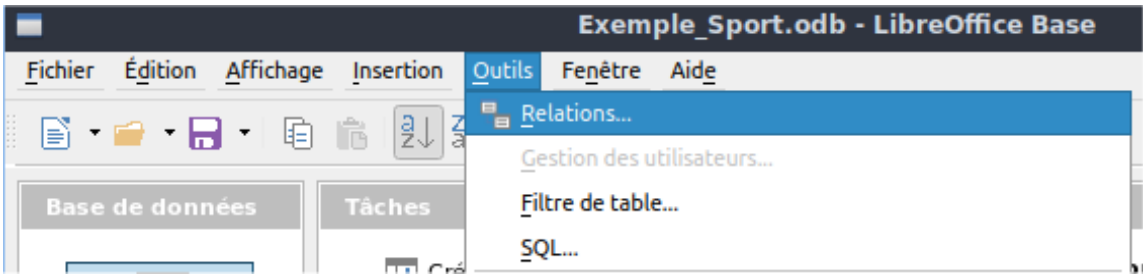
Enregistrez la table sous le nom **rel_Candidats__Sports.**

Conseil

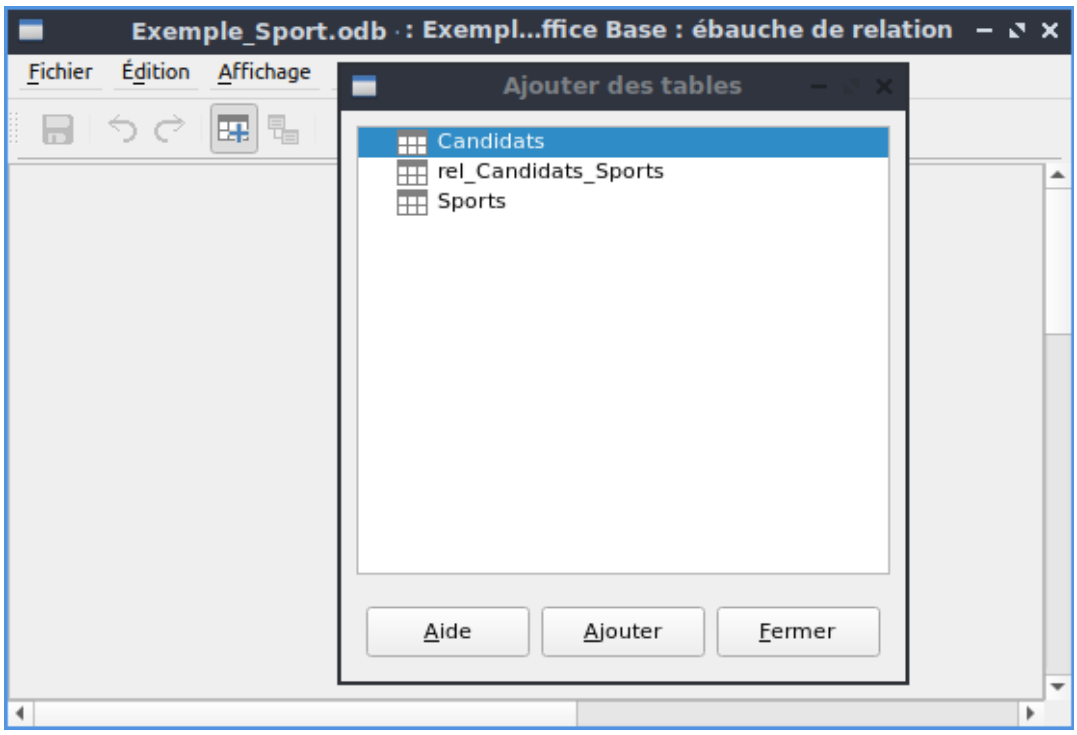
Les résultats d'un concours pourraient également être inclus dans ce tableau. Cependant, si plusieurs compétitions sont organisées, une date de course doit être attachée à la clé primaire commune.

Au fur et à mesure que les tables sont complétées, une relation entre les tables doit être définie. Cela peut empêcher un numéro pour un Candidat d'apparaître dans la table rel_Candidats_Sports qui n'est pas répertorié dans la table Candidats, par exemple.

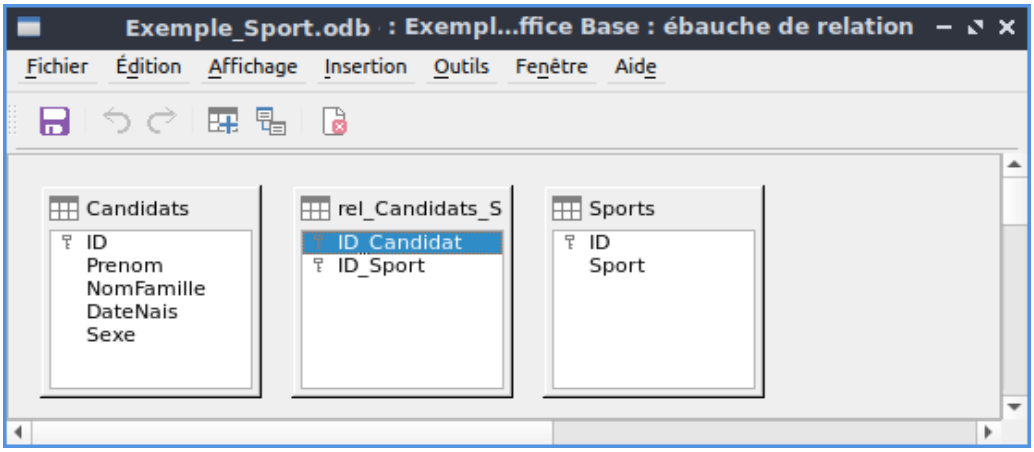
Outils > Relations ouvre la fenêtre de définition des relations



Toutes les tables créées jusqu'à présent sont nécessaires pour la définition de la relation. Cliquez sur chaque table individuelle et cliquez sur le bouton **Ajouter** (où double-cliquer) pour les ajouter à la conception de relations. Fermez ensuite la boîte de dialogue **Ajouter des tables**.

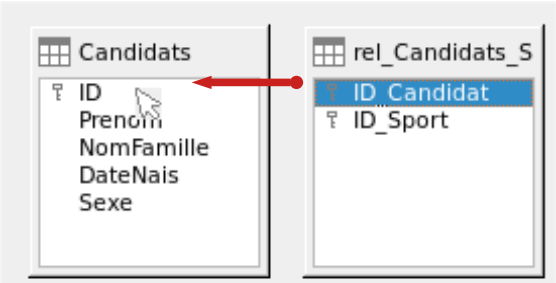


Tous les champs sont répertoriés dans chacune des tables ajoutées. Les champs de clé primaire sont marqués d'un symbole de clé. Les rectangles des tables peuvent être déplacés et redimensionnés.

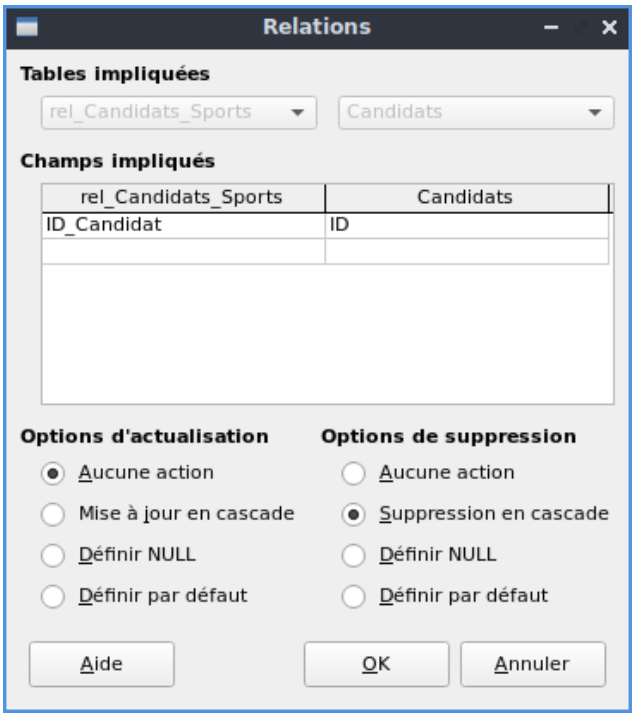


Note

Dorénavant, pour distinguer les champs en fonction des tables auxquels ils appartiennent, nous les désignerons par "Table"."Champ" (Syntaxe SQL dans HSQL).



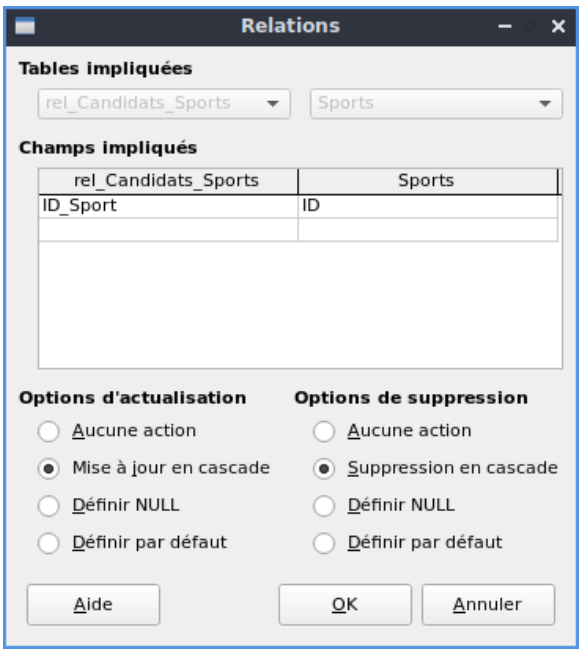
Nous allons créer une relation entre les tables. Faites un clic gauche sur “Candidats”. “ID” (table Candidats). Maintenez le bouton de la souris enfoncé et placez le pointeur sur “rel_Candidats_Sports”. “ID_Candidat”. Le curseur indique un lien. Relâchez le bouton de la souris. La boîte de dialogue suivante apparaîtra pour définir la relation.



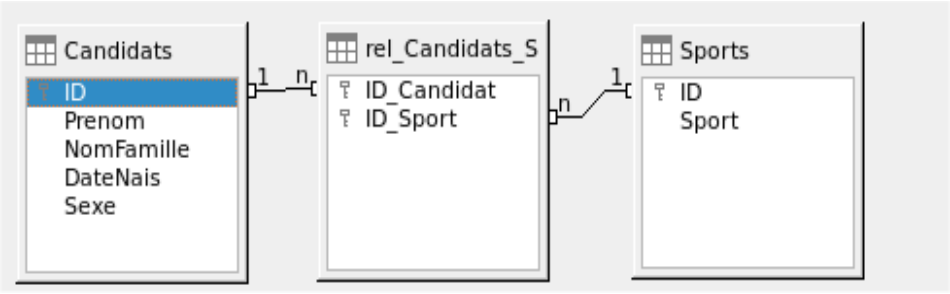
Le champ “rel_Candidats_Sports”.“ID_Sport” ne doit pas être modifié lorsque “Candidats”.“ID” est modifié. C’est la valeur par défaut. “ID” n’est pas modifié, car il s’agit d’un champ auto-incrémenté et aucune entrée n’est nécessaire.

L’enregistrement dans la table rel_Candidats_Sports doit être supprimé lorsque “ID_Candidat” est égal à “Candidats”.“ID” et “Candidats”.“ID” est supprimé. Donc, si un candidat est supprimé de la table Candidats, tous les enregistrements pertinents de la table rel_Candidats_Sports seront supprimés. Cette procédure s’appelle **Suppressions en cascade**.

À l’étape suivante “Sports”. “ID” et “rel_Candidats_Sports”.“ID_sport” sont connectés en faisant glisser la souris depuis rel_Candidats_Sports vers Sports, tout en maintenant le bouton gauche de la souris. Ici aussi, un enregistrement sera supprimé lorsque le sport correspondant sera supprimé.



Cependant, pour un changement de données dans “Sports”.“ID”, une autre variante a été sélectionnée. Si “Sports”.“ID” est modifié, “rel_Candidats_Sports”.“ID_sport” le sera également. Ainsi, l’abréviation d’un sport de 5 caractères maximum peut être facilement ajustée bien qu’il existe déjà de nombreux enregistrements dans la table rel_Candidats_Sports. Cette procédure s’appelle **Mise à jour en cascade**.

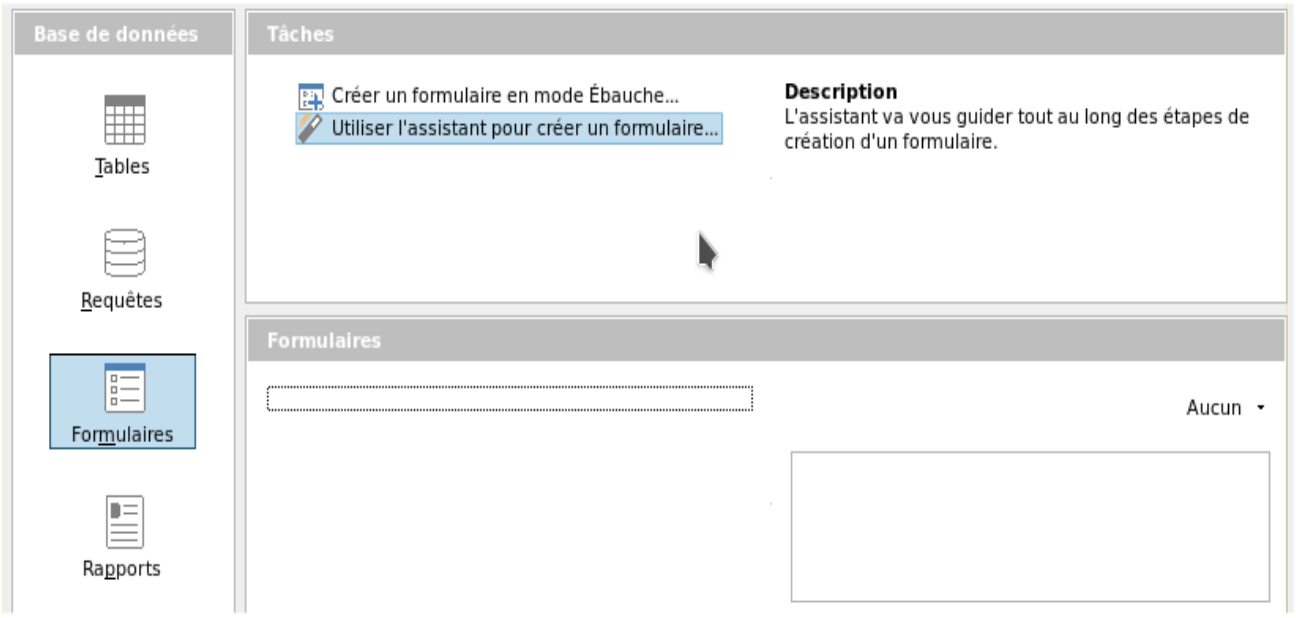


Les champs sont maintenant complètement connectés. Chaque 1:n apparaît aux extrémités des connexions. Un candidat peut apparaître à plusieurs reprises dans la table rel_Candidats_sport. Un sport peut également apparaître à plusieurs reprises dans la table rel_Candidats_Sports. Une combinaison donnée de Candidat et Sport ne peut apparaître dans la table qu’une seule fois. À partir de deux relations 1:n, une relation n:m existe désormais entre Candidats et Sport via la table rel_Candidats_Sports intermédiaire.

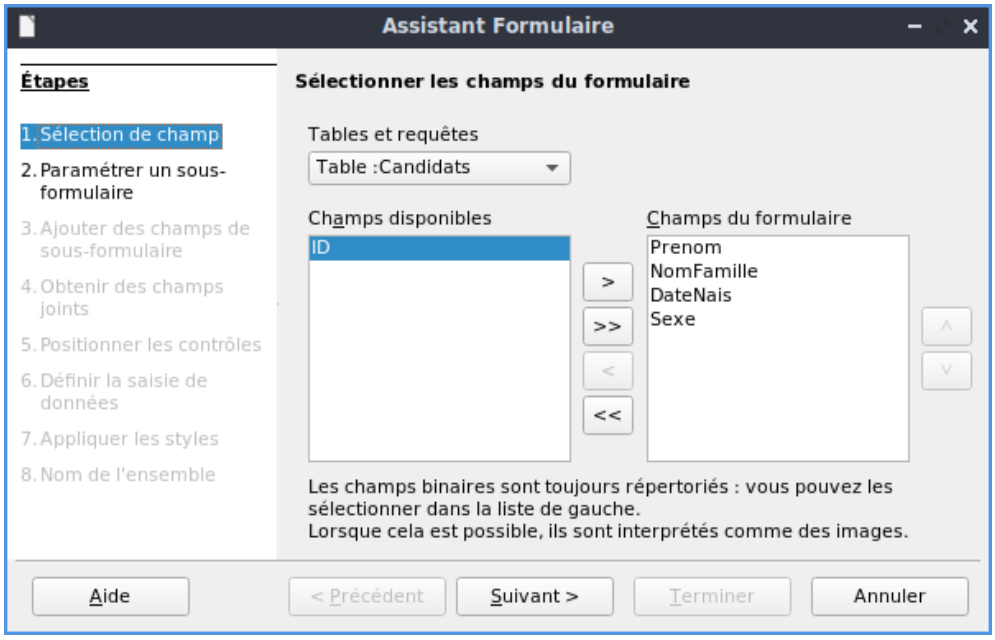
Une telle conception de table peut être mauvaise lorsqu’elle est remplie en tapant du contenu directement dans des tables. Cela nécessite que les trois tables soient ouvertes lorsqu’un candidat est affecté à un sport. “Candidats”.“ID” doit être recherché dans la table Candidats et transmis à “rel_Candidats_Sports”. “ID_Candidat” et “Sports”.“ID” doit être recherché dans la table Sports et transmis à “rel_Candidats_Sports”. C’est trop compliqué. Un formulaire résout ce problème de manière plus élégante.

Créer un formulaire de saisie de données

Les formulaires peuvent être créés directement dans la vue du mode ébauche ou à l’aide de l’assistant. Même les personnes expérimentées peuvent utiliser rapidement l’assistant, puis personnaliser ce qu’il crée pour produire ce qu’elles veulent. C’est souvent le moyen le plus rapide.



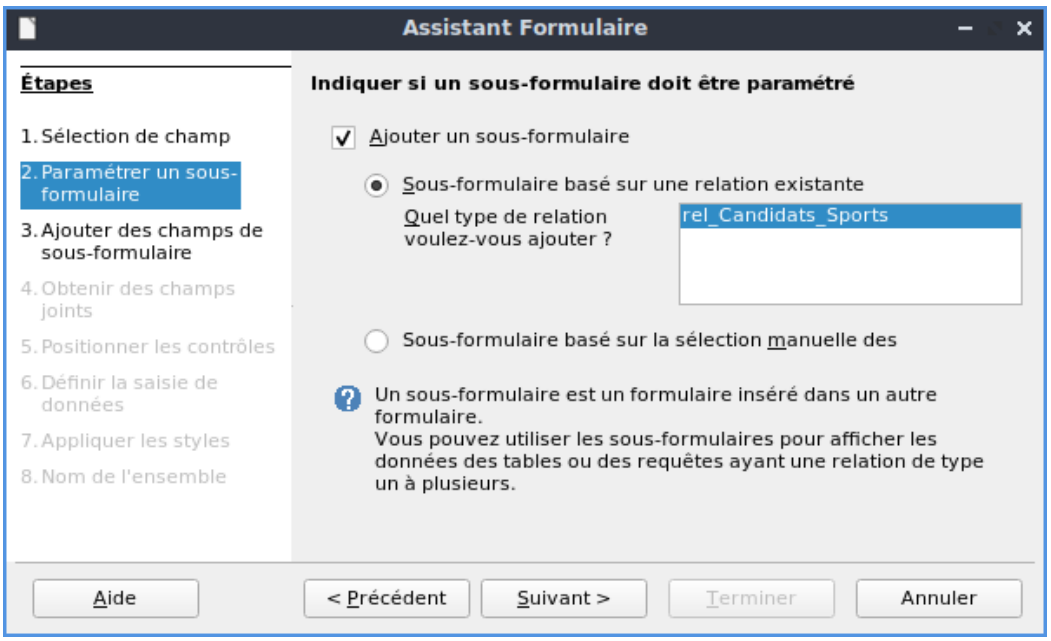
Sélectionnez d'abord Formulaires dans la section Base de données. Ensuite, dans la section **Tâches**, sélectionnez **Utiliser l'assistant pour créer un formulaire**.



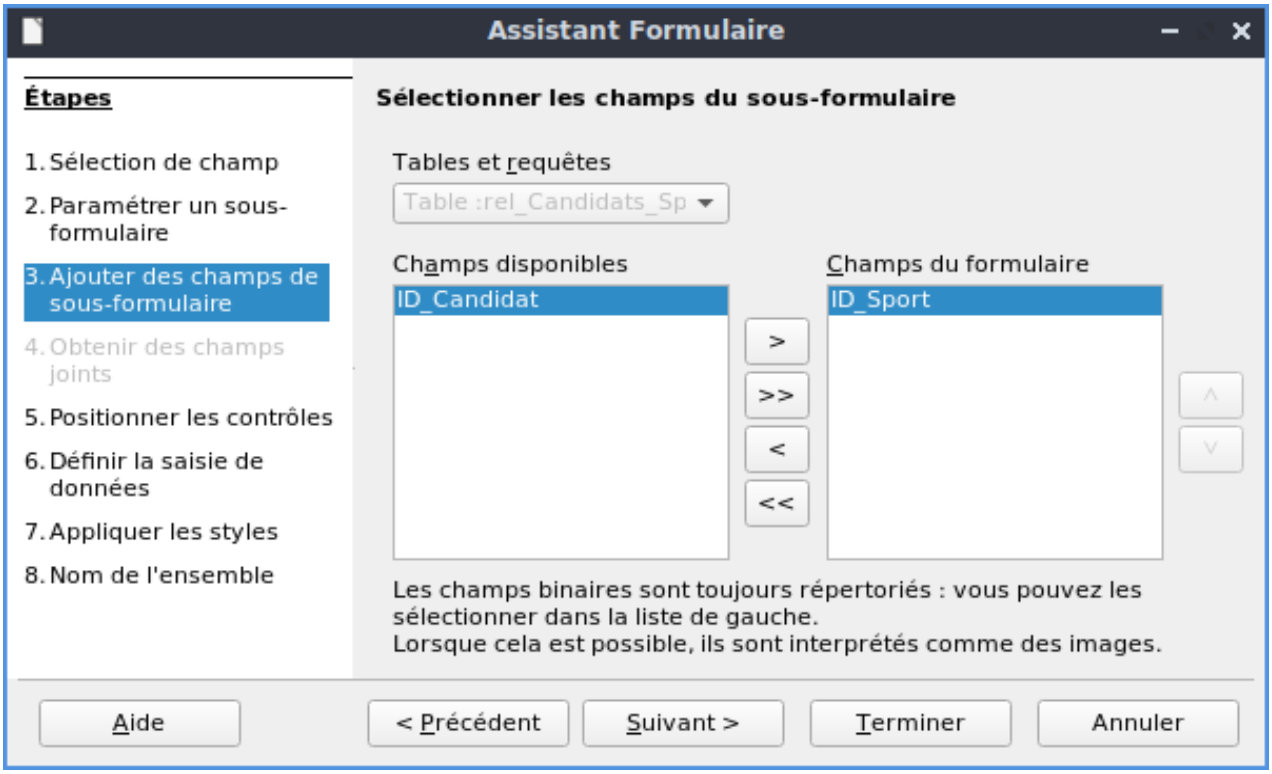
Les données de la table Candidats doivent être écrites dans le formulaire principal. Les données du tableau Sport sont chargées directement avec les quelques sports nécessaires et seront rarement mises à jour.

Tous les champs à l'exception du champ de clé primaire "ID" sont nécessaires à partir de la table Candidats. Le champ de clé primaire est rempli automatiquement avec une valeur distinctive correspondante.

Sélectionnez les champs dans la liste *Champs disponibles* et utilisez les boutons fléchés pour les déplacer vers la liste des *Champs du formulaire*. Cliquez sur **Suivant**.

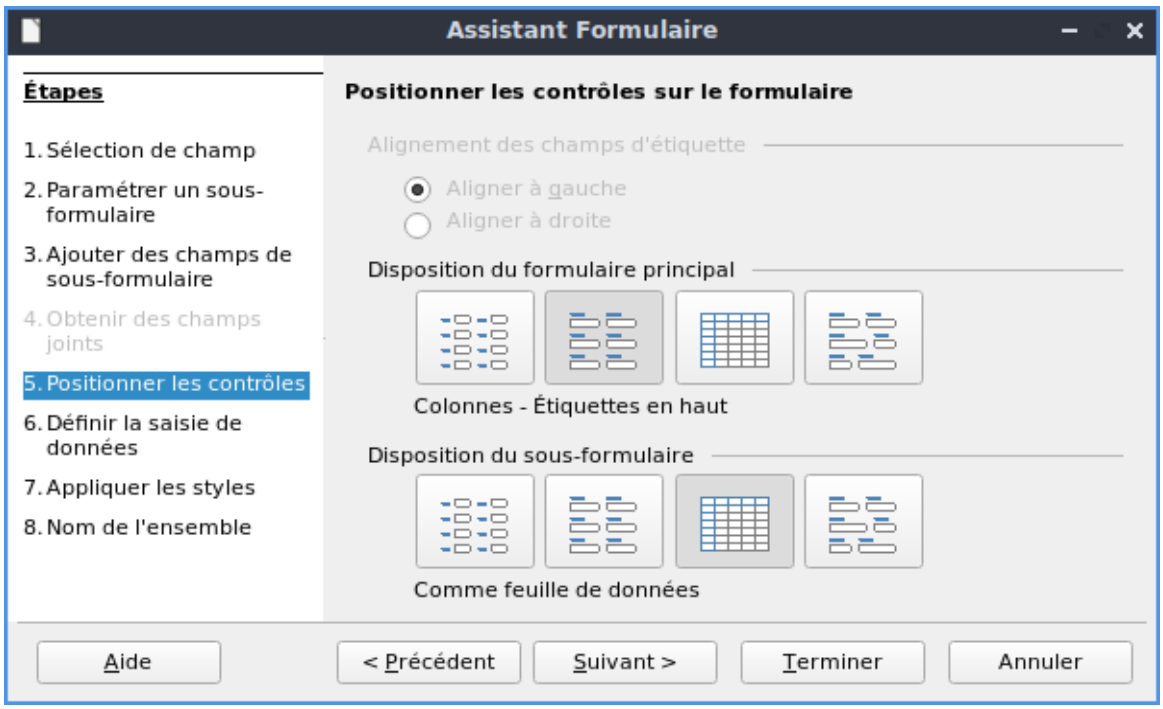


Un sous-formulaire doit être mis en place où un sport peut être attribué à un Candidat. À l'étape 2, sélectionnez Ajouter un sous-formulaire en fonction de la relation existante. Pour confirmer la relation précédemment définie, sélectionnez rel_Candidat_Sports. Cliquez sur **Suivant**.



Seul le champ ID_Sport est requis dans la table rel_Candidats_Sports. La clé primaire dans la table Candidats fournit la valeur du champ ID_Sport pour l'enregistrement en cours par la connexion du formulaire principal au sous-formulaire.

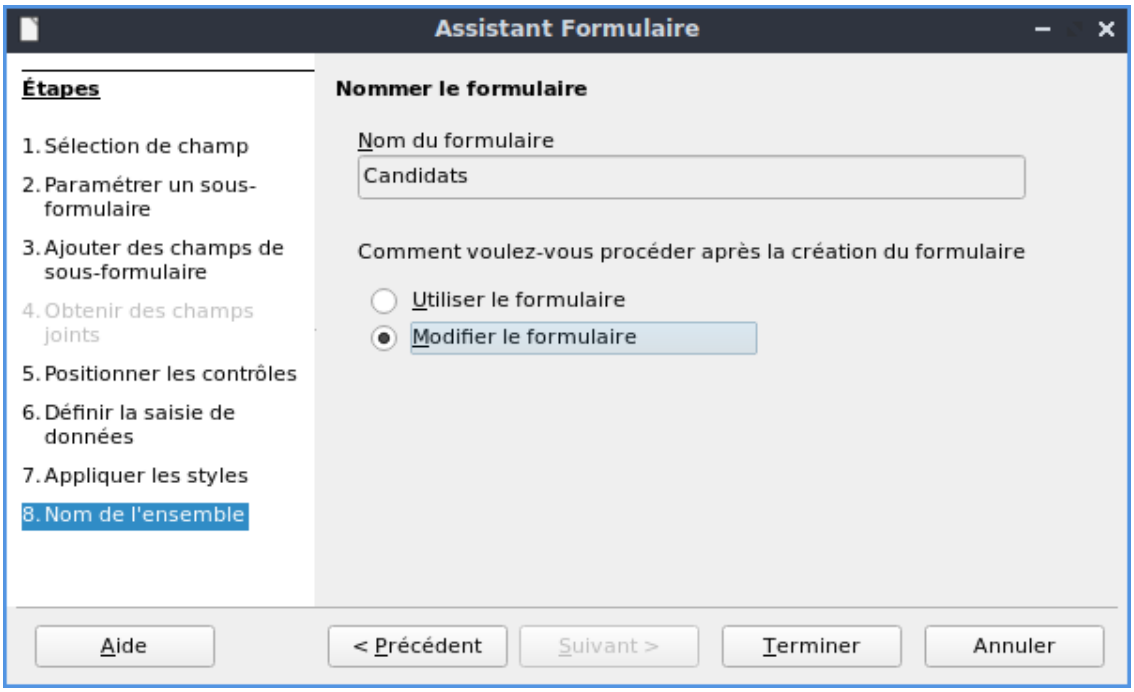
Cette connexion est déjà réglée, car le sous-formulaire est basé sur une relation existante. L'étape 4 de l'assistant : *Obtenir les champs joints* est donc inactif. Si le sous-formulaire est basé sur la sélection manuelle des champs, l'étape 4 est alors active et permet de réaliser la connexion manuellement (choix à déconseiller dans le présent exemple).



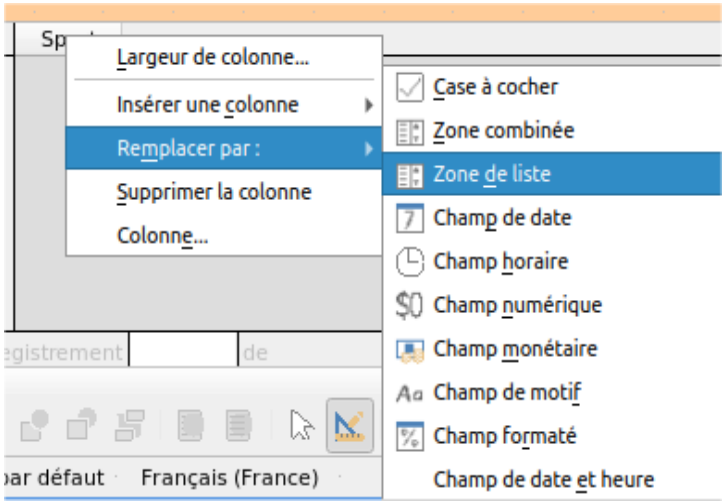
La manière dont les éléments du formulaire principal et du sous-formulaire sont organisés n'a pas d'importance en fin de compte. Cependant, l'attribution des données doit également être simple pour les inexpérimentés. Dans notre exemple, **Comme feuille de données** ne doit pas être sélectionné pour le formulaire principal, à la place choisissez **Colonnes – Étiquettes en haut**. Les champs du sous-formulaire afficheront plus tard tous les sports des Candidats, il est donc préférable de laisser la disposition du sous-formulaire telle quelle : **Comme feuille de données**.

Étape 6 (Définir la saisie des données) doit rester telle quelle : **Le formulaire doit afficher toutes les données**. Ainsi, une nouvelle entrée est possible, ainsi qu'une modification des données existantes.

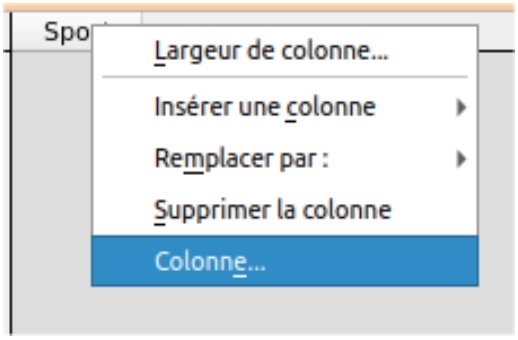
Étape 7 (Appliquer les styles). C'est une question de goût. Attention : certains styles impliquent des images inattendues à faible contraste, en particulier dans les champs de contrôle de table. Ici, la couleur de police des champs de la feuille de données doit être réajustée si nécessaire.



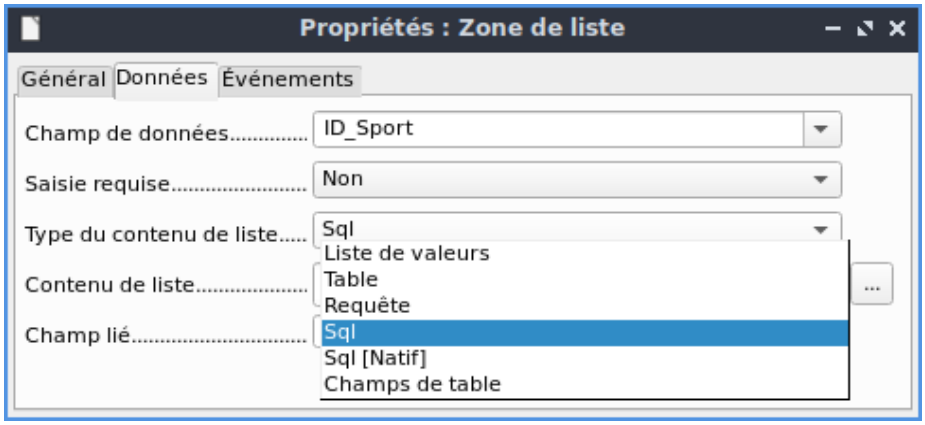
Le formulaire ne fonctionnera pas encore, car le sous-formulaire vous oblige toujours à entrer l'abréviation pour les sports. Ce qu'il faut, c'est une sélection de sports après le nom complet souhaité. Par conséquent, à l'Étape 8 (Définir le nom), nommez le formulaire **Candidats** et sélectionnez **Modifier le formulaire**. Cliquez sur **Terminer**.



Sur la feuille de données, cliquez droit sur l'en-tête du tableau, ID_Sport. Dans le menu contextuel, sélectionnez **Remplacer par > Zone de liste**.

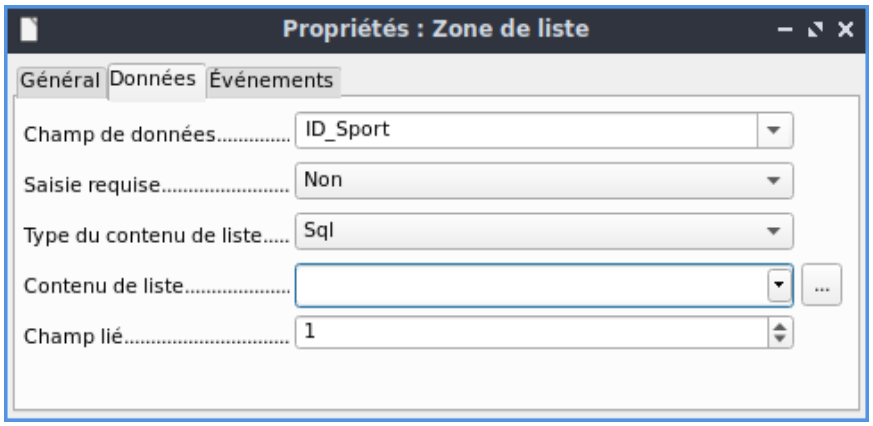


Ensuite, la liste doit être traitée afin qu'elle puisse également afficher les données correspondantes. Cliquez à nouveau avec le bouton droit sur ID_Sport et sélectionnez **Colonne**.



Cela ouvre les propriétés de la zone de liste sélectionnée. Sélectionnez **Données > Type du contenu de la liste > Sql**. Avec l'aide de SQL (Structured Query Language – langage de requête standard pour les bases de données), le champ doit obtenir son contenu de la table Sport.

Lorsque Sql est sélectionné, le *Contenu de liste* a un bouton d'ellipse sur la droite (…). Cliquez sur ce bouton pour ouvrir l'éditeur de création de requêtes. La requête en cours de création sera rassemblée et finalement enregistrée dans la zone de liste elle-même.



Dans la boîte de dialogue Ajouter une table ou une requête, sélectionnez **Tables > Sports**, puis cliquez sur **Ajouter** et **Fermer**.

Dans la première colonne de la partie inférieure de l'éditeur de requêtes, cliquez dans la case en regard de Champ et sélectionnez Sports.Sport dans la liste déroulante.

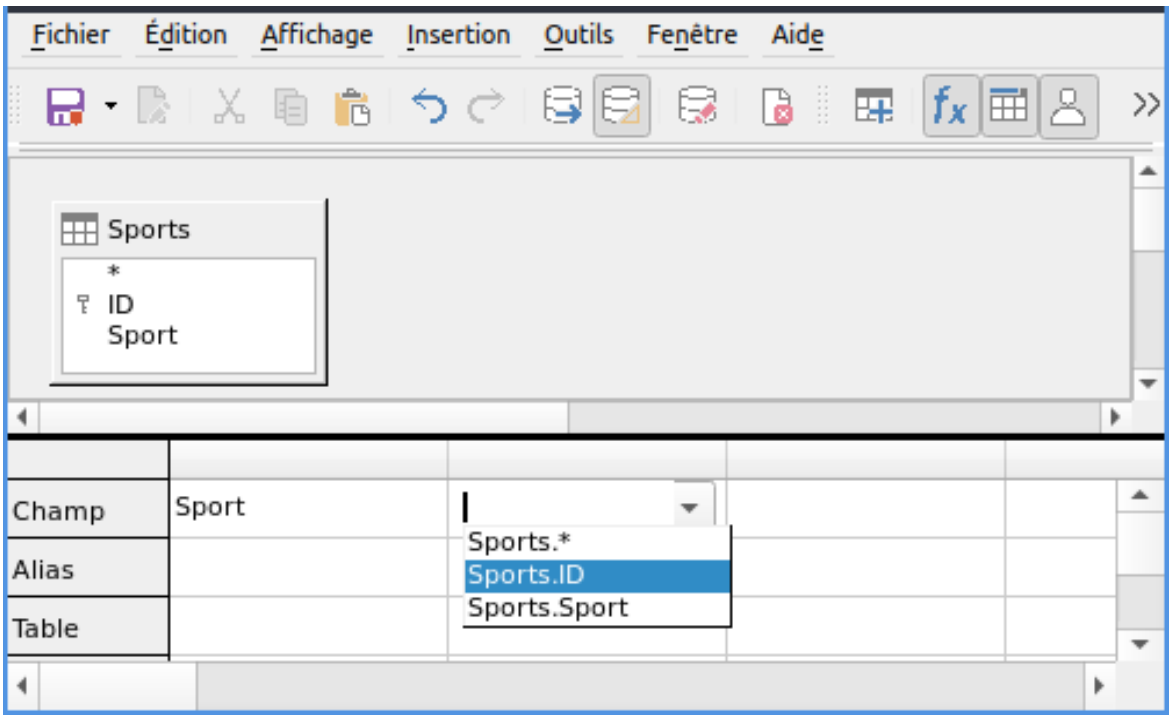
Dans la deuxième colonne, sélectionnez le champ "Sports"."ID". Ce champ transmet ensuite sa valeur à la table, qui est la source de données du sous-formulaire. Les mots inscrits sont donc affichés et les raccourcis appropriés stockés.

Enregistrez la requête (Icône disquette), qui est ensuite transmise aux propriétés de la zone de liste. Fermez l'éditeur de requêtes.

Le champ *Contenu de liste* de la boîte de dialogue **Propriétés** affiche maintenant le code SQL qui a été créé dans l'éditeur de requête :

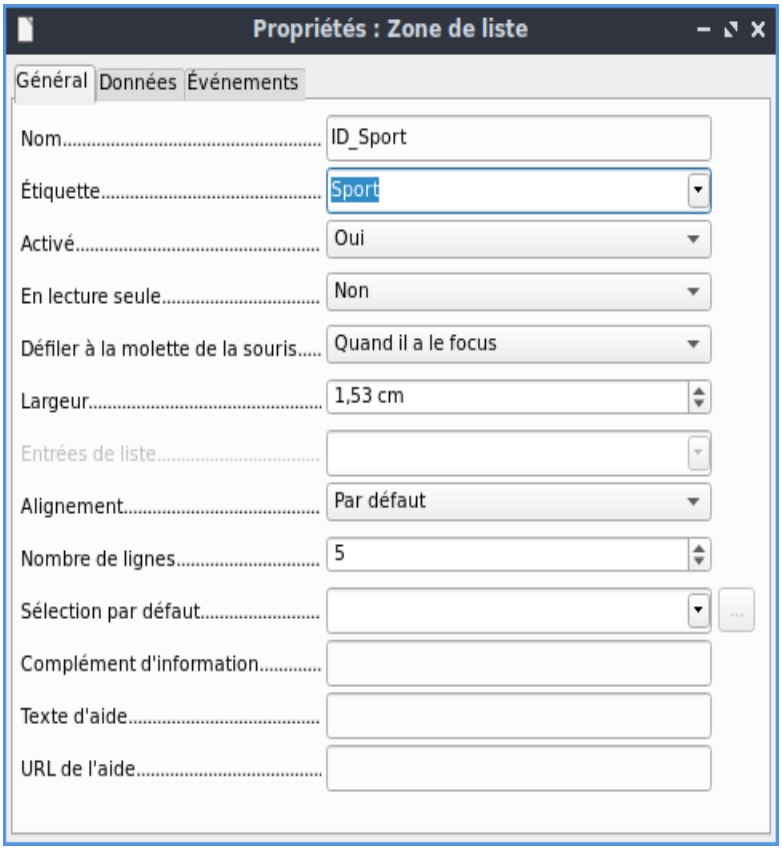
```
SELECT "Sport","ID" FROM "Sports"
```

Ce code dit : Dans la table "Sports", sélectionner le champ "Sport" et la valeur clé associée "Sports"."ID".

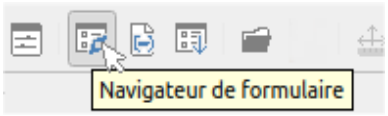


Cette requête illustre le minimum qui doit être sélectionné. Bien entendu, le tri pourrait être intégré. La sauvegarde des abréviations avec une sélection habile donne une liste utile des sports stockés dans “ID”. Si les enregistrements ne sont pas triés d’une manière spécifiée, le tri est toujours effectué par le champ de clé primaire. Afin de voir les sports plus tard dans la zone de liste, ce contenu doit être entré dans la table Sports en conséquence.

Nous devons maintenant changer le libellé du champ ID_Sport. Il est toujours stocké sous ID_Sport, mais nous voulons qu’il soit visible en tant que sport. Par conséquent : **Général > Étiquette > Sport**. Fermez la boîte de dialogue Propriétés.



Si vous souhaitez modifier les noms d’autres champs, il est préférable d’utiliser le Navigateur de formulaires. Si vous cliquez sur des champs, non seulement les champs, mais également les étiquettes sont sélectionnées. Grâce à l’assistant, ils ont été regroupés. Cela nécessite ensuite une action supplémentaire du menu contextuel de la sélection.



Le Navigateur de formulaire est lancé à partir de la barre d’outils de conception de formulaire en bas de la fenêtre.

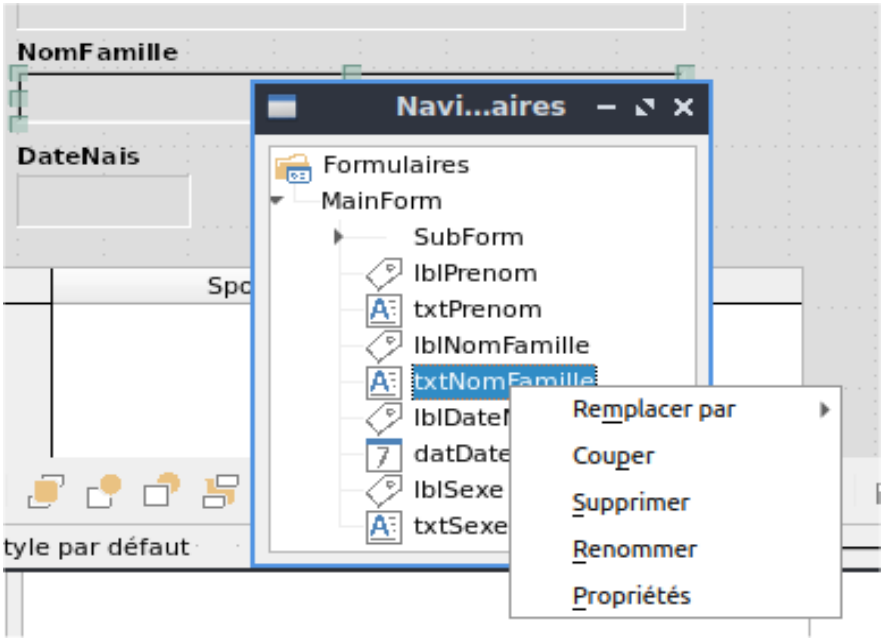
Conseil

Parfois, la barre d’outils ne s’ouvre pas correctement pour créer des formulaires lors du traitement des formulaires. Le navigateur de formulaire est introuvable dans ce cas.

Pour afficher les barres d’outils, choisissez **Affichage > Barres d’outils > Ébauche de formulaire** et **Contrôles de formulaire**. S’ils sont ensuite visibles lors de la saisie des données, la vue doit à nouveau être modifiée en conséquence.

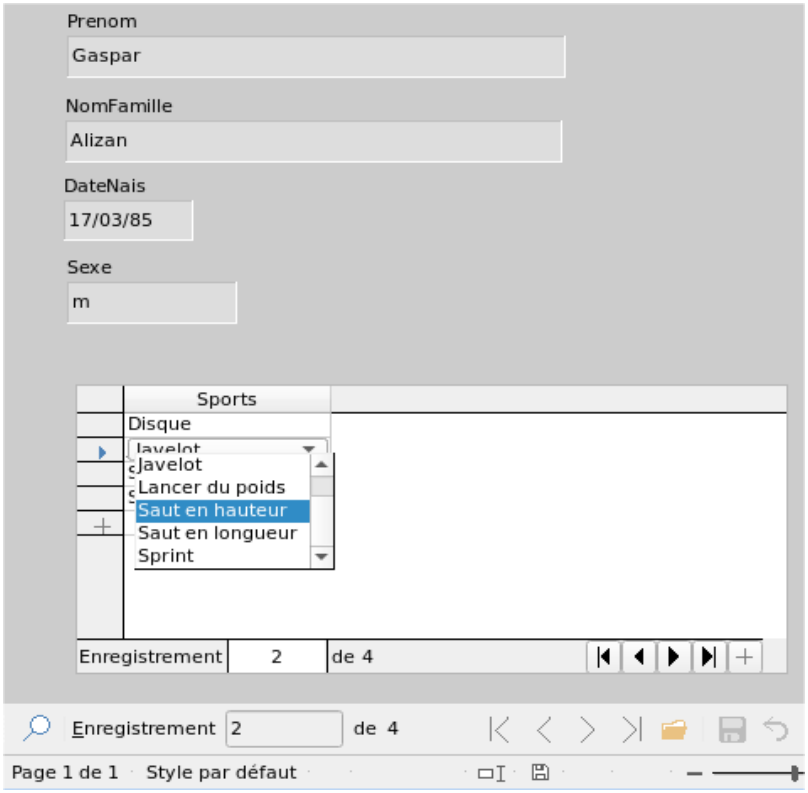
L’usage du Navigateur de Formulaires est fortement conseillé car, car il permet de naviguer facilement entre les contrôles et d’accéder aux paramètres des formulaires.

Chaque champ peut être examiné individuellement avec le Navigateur de formulaires. Les propriétés du champ sont alors accessibles dans le menu contextuel. Une entrée de propriété est automatiquement enregistrée après avoir accédé à une autre propriété. Il est possible de passer d’un champ à un autre même lorsque la boîte de dialogue des propriétés est ouverte. Ici aussi, le niveau intermédiaire respectif est stocké.



Si la conception est maintenant terminée, enregistrez et fermez le formulaire. Puis enregistrez à nouveau le fichier de base.

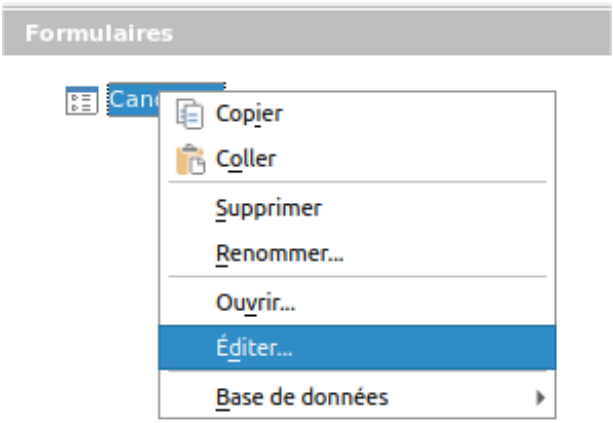
Maintenant, si vous souhaitez saisir des données dans le formulaire, cela peut ressembler à ceci. Dans le formulaire suivant, un enregistrement a été saisi pour le test. Après avoir entré le sexe, cliquez sur le bouton **Enregistrer**. Bien entendu, le tri pourrait être intégré.



Lors de l'utilisation du formulaire, vous remarquerez quelques inconvénients :

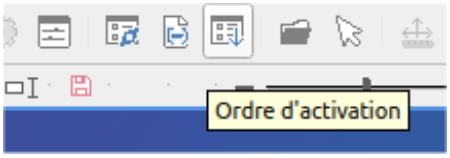
- Le sous-formulaire avec Sports n'est pas directement accessible. Si vous naviguez avec la touche Tab, après avoir entré le sexe, l'onglet passe directement au début de l'enregistrement suivant.
- La barre de navigation, si on se trouve dans le sous-formulaire, affiche le numéro d'enregistrement du sous-formulaire. Vous devriez naviguer uniquement dans le formulaire principal.
- Le sexe est entré en fonction de la préférence de l'utilisateur trouvée en mémoire. Dans le tableau, la longueur du champ a été limitée à un seul caractère. Donc, ici, une entrée plus sûre doit être garantie.

Pour résoudre ces problèmes, ouvrez le formulaire pour le modifier, non pour la saisie de données.

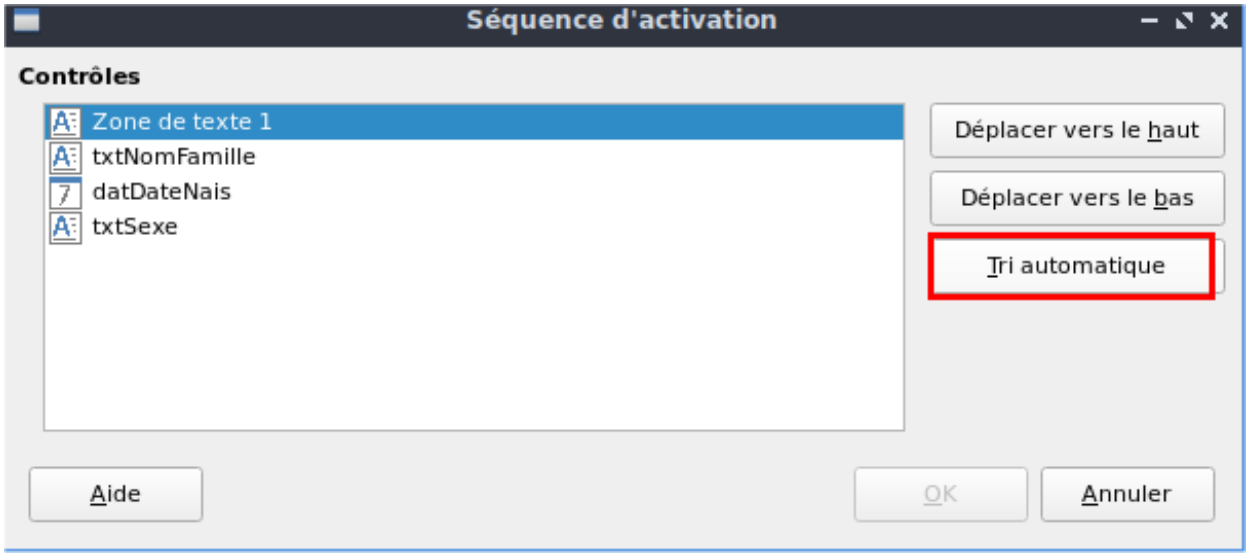


Tabulation dans le sous-formulaire

Afin de ne pas passer directement à l'enregistrement de démarrage suivant en utilisant la touche Tab après avoir saisi le sexe, il est nécessaire de modifier la séquence d'activation. Cliquez sur Ordre d'activation dans la barre d'outils Ébauche de formulaire.

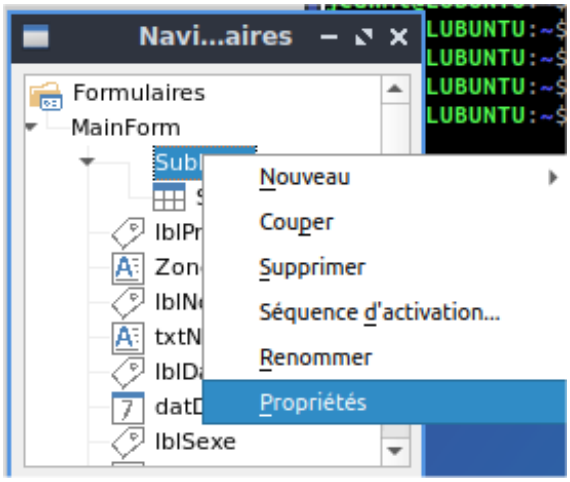


Dans la boîte de dialogue Ordre de tabulation, sélectionnez Tri automatique, qui affecte non seulement le tri des contrôles affichés, mais également la redirection automatique dans le sous-formulaire. Bien que cela ne soit pas visible dans la boîte de dialogue, il est réglé de cette manière en arrière-plan.

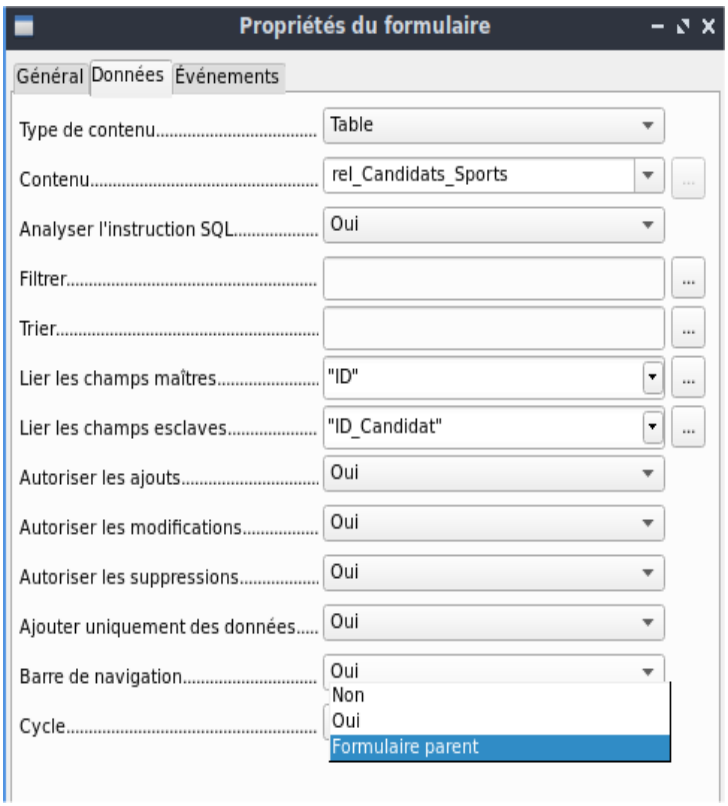


Activer la barre de navigation du formulaire principal dans le sous-formulaire

Ouvrez le navigateur de formulaires pour afficher les propriétés du sous-formulaire.



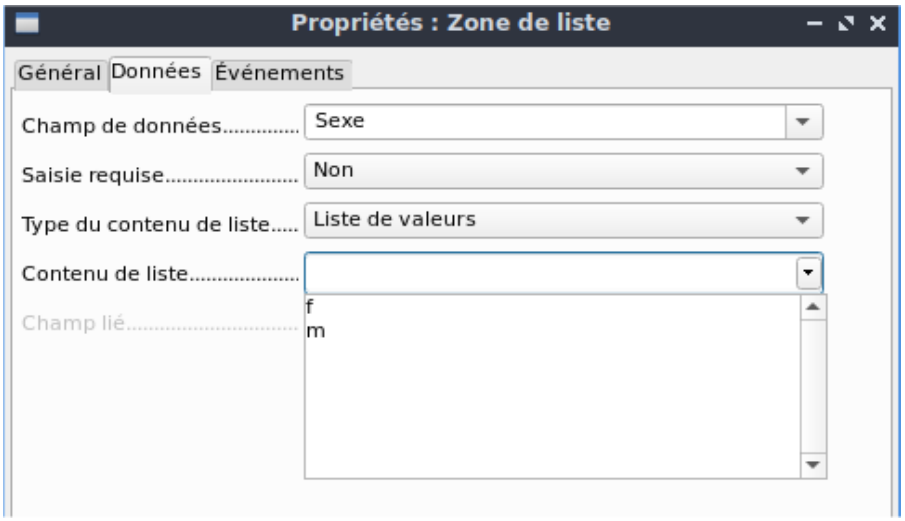
Sous l'onglet **Données > Barre de navigation,** modifiez la valeur de *Oui* à *Formulaire parent*. Désormais, la barre d'outils de navigation affiche toujours le numéro de l'enregistrement de la table Candidats.



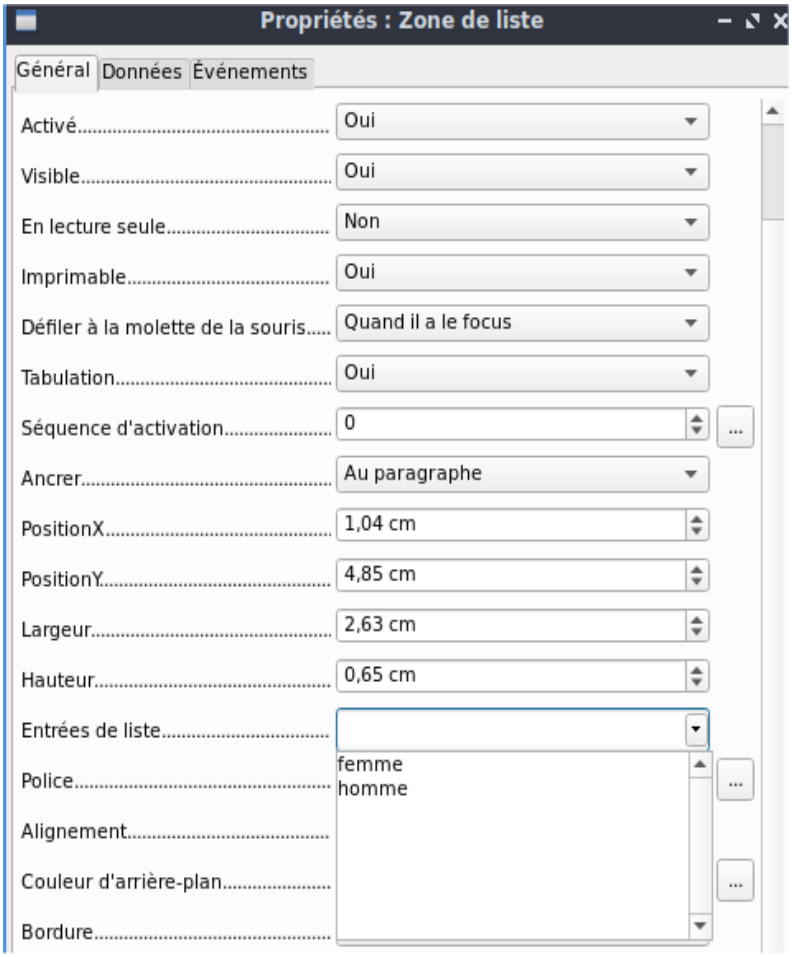
Restreindre la saisie dans un contrôle

Pour limiter l'entrée aux valeurs spécifiées, le contrôle ne peut pas être une simple zone de texte. Dans le formulaire principal, la solution peut être trouvée en saisissant le sexe avec des boutons radio dans une boîte de groupe. Une autre solution consiste à présenter les choix dans une zone de liste.

Tout d'abord, mettez en surbrillance le champ Sexe sur le navigateur. Cliquez avec le bouton droit sur ce champ pour ouvrir le menu contextuel. Sélectionnez **Remplacer par> Zone de liste**. Les propriétés seront sélectionnées à l'aide du menu contextuel.



Onglet **Données > Type de contenu de liste > Liste de valeurs** est par défaut dans cette propriété. Dans **Données > Contenu de liste**, entrez les caractères f et m l'un en dessous de l'autre (en utilisant Maj + Entrée). Ces abréviations sont les données qui seront affectées à la table Candidats, ou récupérées dans cette table pendant la lecture des enregistrements.



Dans l'onglet **Général > Entrées de liste**, spécifiez ce qui sera affiché dans la zone de liste. Cela peut également être f et m, ou il peut s'agir de femme et homme comme indiqué sur l'illustration. Il doit en tout cas avoir le même ordre que les éléments de l'onglet **Données**.

Sélectionnez ensuite **Oui** pour la propriété **Général > Liste déroulante**. Regardez vers le bas de l'onglet Général pour cette propriété.

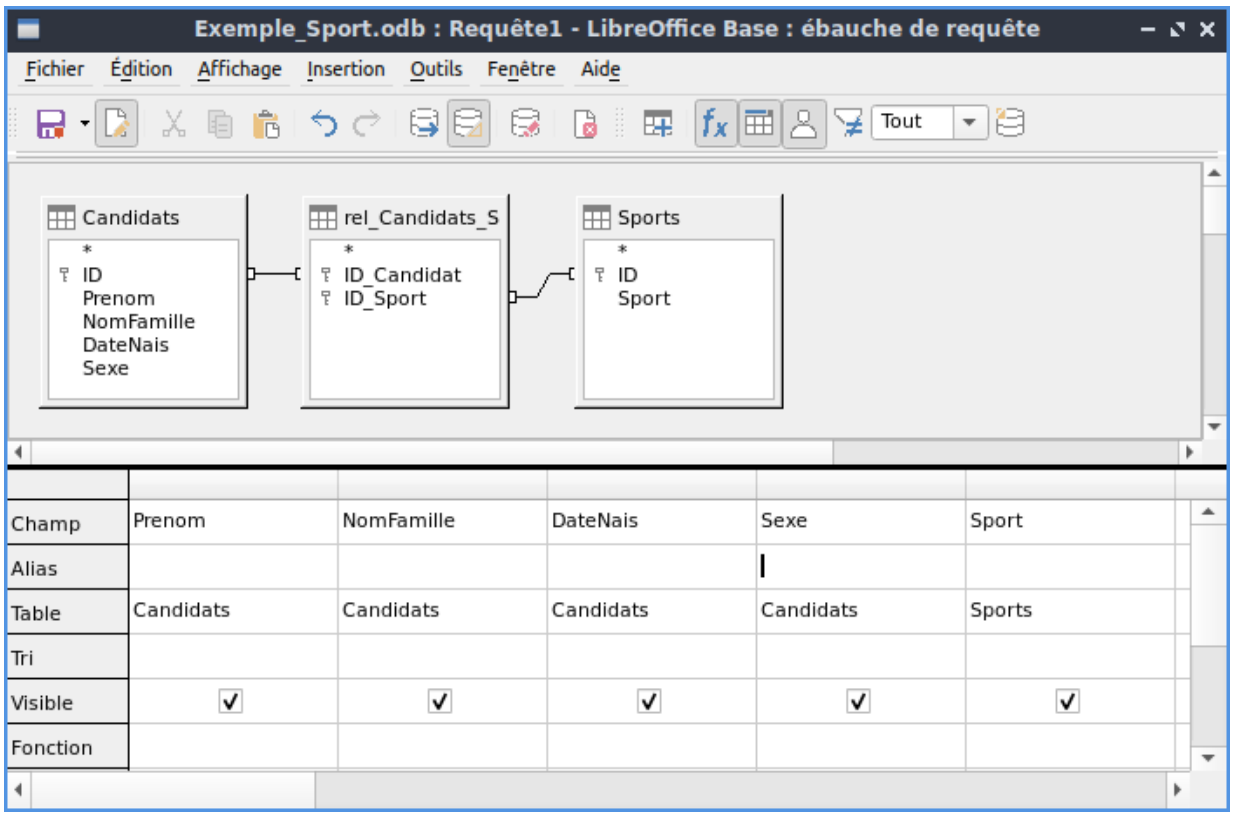
Les inconvénients les plus frappants ont jusqu'à présent été éliminés. Maintenant, vous devez à nouveau enregistrer le formulaire et le fichier de base de données. La saisie pour les candidats masculins et féminins peut commencer ainsi que leur affectation aux sports.

Ainsi, l'étape suivante est utile : les enregistrements ne doivent être saisis qu'une seule fois. Veillez à ce que les candidats puissent également s'affronter par âge et par sport. Sinon, les requêtes et rapports suivants n'ont pas de sens.

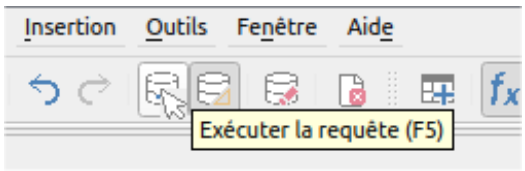
Créer une requête

Dans une requête, les contenus de différentes tables peuvent être regroupés. Chacun des candidats doit être affiché avec les sports auxquels il a participé.

Cliquez sur **Requêtes > Créer une requête en mode Ébauche**. Une boîte de dialogue, listant toutes les tables de la base données, apparaît et dans laquelle nous sélectionnerons, une à une, toutes les tables utiles pour la requête. Cela sera très clair lorsque la table rel_Candidats_Sports est sélectionnée comme deuxième table. Ensuite, on voit que les connexions des relations sont également bien reconnues.

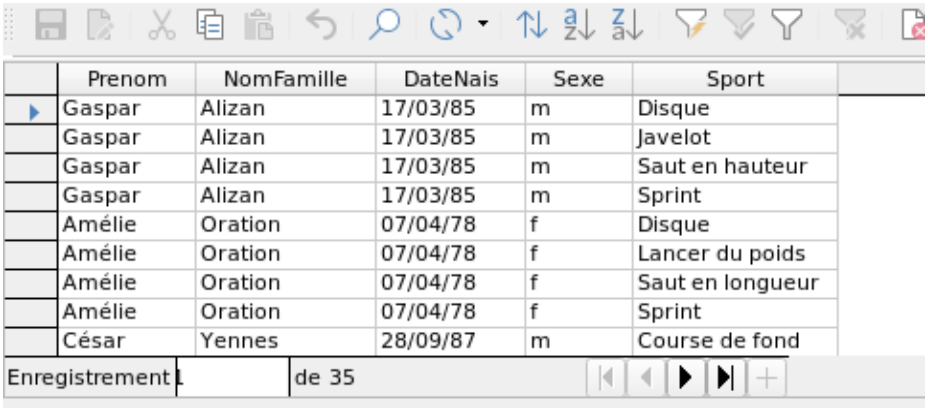


Les champs qui doivent être affichés dans la requête peuvent être ajoutés soit en double-cliquant sur les noms de champs dans la table, soit en sélectionnant une case dans la ligne Champ. Une liste déroulante apparaît, composée des noms de champ et de leurs noms de table correspondants indexés par nom de table. Afin de faire correspondre correctement les champs des tables à leurs tables, ils sont étiquetés “Nom de table”.“Nom de champ” dans les requêtes. Si “*” est utilisé à la place du nom de champ, cela signifie que tous les champs de la table correspondante sont affichés.



En principe, une requête doit être exécutée une fois avant l’enregistrement pour voir si elle produit réellement le résultat souhaité. À cet effet, trois possibilités sont offertes :

- cliquez sur le bouton **Exécuter la requête** illustrée ci-dessus ;
- appuyez sur *F5*
- ou choisissez **Édition > Exécuter la requête**.



	Prenom	NomFamille	DateNais	Sexe	Sport
▶	Gaspar	Alizan	17/03/85	m	Disque
	Gaspar	Alizan	17/03/85	m	Javelot
	Gaspar	Alizan	17/03/85	m	Saut en hauteur
	Gaspar	Alizan	17/03/85	m	Sprint
	Amélie	Oration	07/04/78	f	Disque
	Amélie	Oration	07/04/78	f	Lancer du poids
	Amélie	Oration	07/04/78	f	Saut en longueur
	Amélie	Oration	07/04/78	f	Sprint
	César	Yennes	28/09/87	m	Course de fond

Enregistrement 1 de 35

La requête affiche désormais toutes les combinaisons de candidats et de sports valides. Si les candidats ont plusieurs sports, ils ont autant d'enregistrements. Les candidats sans sport n'apparaissent pas.

Afin de rassembler différentes tranches d'âge dans un concours, l'année de naissance est déterminante. Cela dépend de l'âge d'une personne cette année-là.

L'année de naissance peut être utilisée avec différentes fonctions. Par exemple, considérez les données suivantes.

Champ	Prenom	NomFamille	DateNais	Sexe	Sport	YEAR(NOW( )) - YEAR("DateNais")
Alias						
Table	Candidats	Candidats	Candidats	Candidats	Sports	

```
YEAR(NOW())-YEAR("DateNais")
```

NOW () signifie "maintenant", cette fonction retourne la date et l'heure système. L'année en cours est lue avec YEAR (NOW ()).
YEAR ("DateNais") extrait l'année de naissance. Une différence est calculée entre les deux, indiquant l'âge ou plus exactement l'âge de la personne dans l'année en cours.
Ces fonctions, et bien d'autres, qui fonctionnent avec le moteur HSQLDB intégré, sont décrites dans l'annexe A de ce manuel.

	Prenom	NomFamille	DateNais	Sexe	Sport	YEAR(NOW( )) - YEAR("DateNais")
▶	Gaspar	Alizan	17/03/85	m	Disque	35
	Gaspar	Alizan	17/03/85	m	Javelot	35
	Gaspar	Alizan	17/03/85	m	Saut en hi	35
	Gaspar	Alizan	17/03/85	m	Sprint	35
	Amélie	Oration	07/04/78	f	Disque	42
	Amélie	Oration	07/04/78	f	Lancer du	42
	Amélie	Oration	07/04/78	f	Saut en lo	42
	Amélie	Oration	07/04/78	f	Sprint	42
	César	Yennes	28/09/87	m	Course de	33

Enregistrement 1 de 35

Lorsque la requête est exécutée au cours d'une année donnée (par exemple 2020), les résultats appropriés apparaissent pour cette même année.

L'ensemble du code que nous avons saisi dans le champ apparaît également dans l'en-tête de colonne. Cela est résolu en entrant un alias pour le code sous lequel le résultat apparaîtra.

Champ	YEAR(NOW( )) - YEAR("DateNais")
Alias	AgeSport
Table	

Dans la ligne Alias, le terme “AgeSport” est entré. Pour une personne qui n’a pas encore fêté son anniversaire cette année, l’âge calculé indique 1 an en trop. Voir plus bas pour un remède à ce défaut

Sport	AgeSport
Disque	35
Javelot	35
Saut en hauteur	35
Sprint	35
Disque	42
Lancer du poids	42

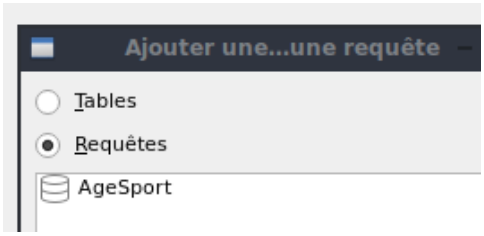
Si la requête est exécutée à nouveau, l’en-tête de colonne ne contient plus le code, mais le terme AgeSport.

Ainsi, la requête sera enregistrée sous le nom AgeSport pour éviter la confusion liée à l’utilisation du code comme nom.

Pour remédier à l’erreur citée précédemment (Age erroné pour une personne n’ayant pas encore fêté son anniversaire), on pourra remplacer le code du champ par le suivant :

```
CASEWHEN
   ( MONTH( "DateNais" ) > MONTH( CURDATE( ) )
   OR ( MONTH( "DateNais" ) = MONTH( CURDATE( ) )
   AND DAY( "DateNais" ) > DAY( CURDATE( ) ) ) )
THEN DATEDIFF( 'yy', "DateNais", CURDATE( ) ) - 1
ELSE DATEDIFF( 'yy', "DateNais", CURDATE( ) )
```

Cette requête est ensuite utilisée comme base pour la requête suivante, que le AgeSport attribue désormais à un Groupe_Age.

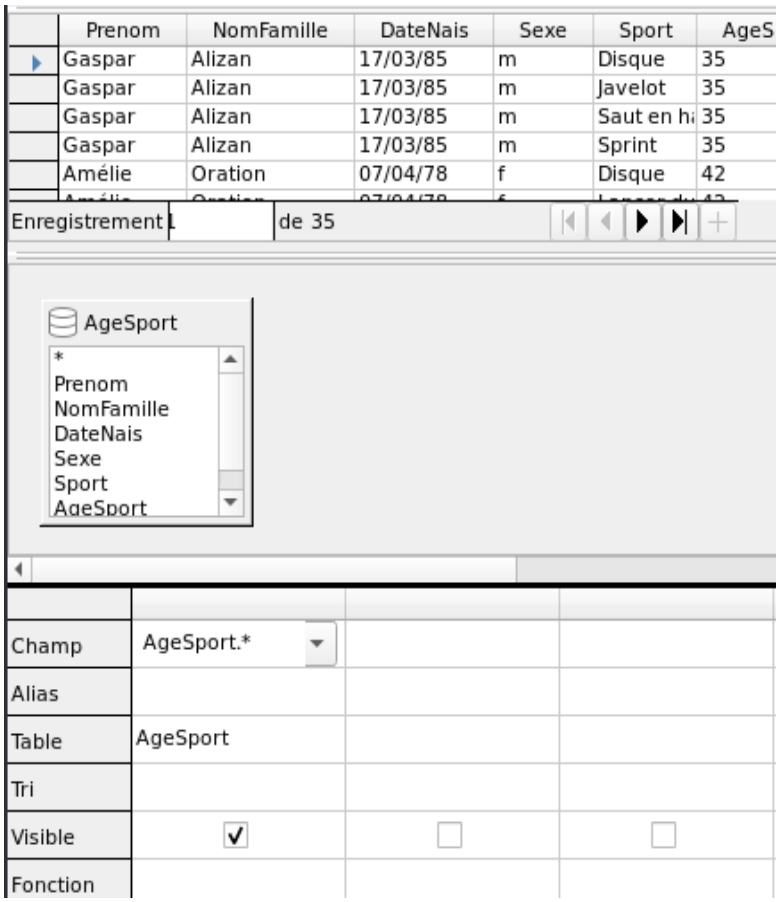


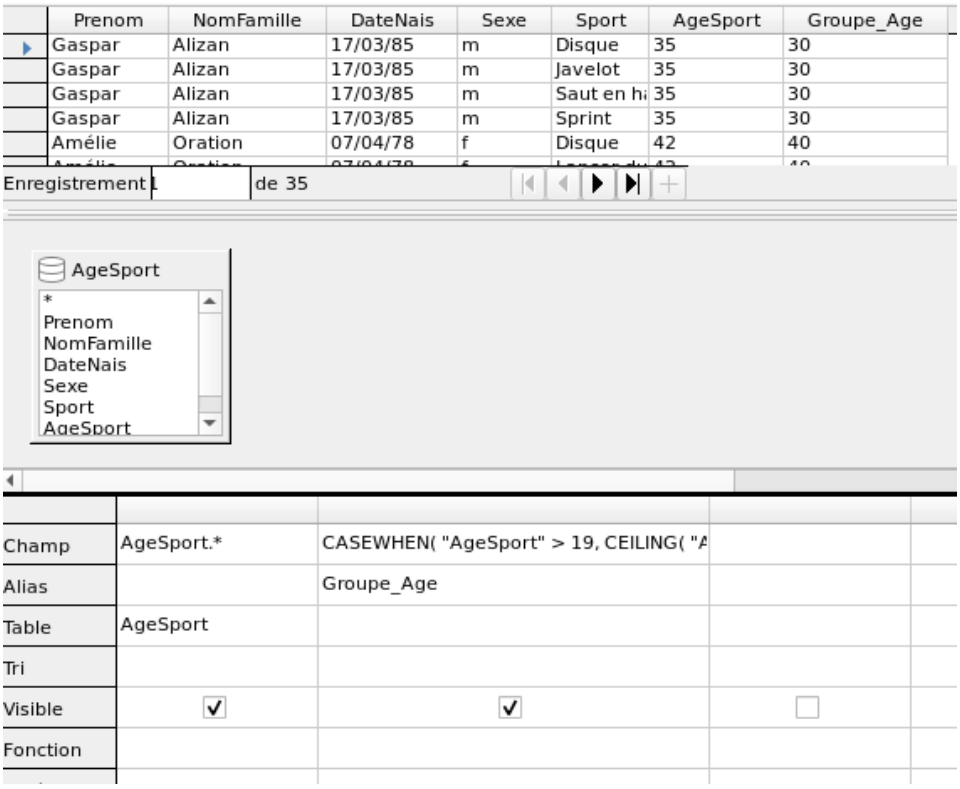
La requête AgeSport a été choisie comme base de la 2e requête. Avant le terme AgeSport dans le conteneur de table, il y a une icône différente de celle des tables de la première requête. Ce symbole indique que la base de la nouvelle requête est une requête et non une table.

Double-cliquez sur * ou sélectionnez *AgeSport.** pour sélectionner tous les champs. La requête suivante renvoie donc le même résultat, mais la formule n’est plus visible.

Accédez maintenant au champ AgeSport. Il va permettre de déterminer dans quelle tranche d’âge la personne concernée participe.

Afin que le calcul ne soit pas trop complexe, il convient de créer les classifications suivantes : pour les moins de 20 ans, les candidats sont divisés en groupes d’âge de 2 ans. À partir de 20 ans et plus, les groupes sont formés par tranche de 10 ans, par exemple, 20–29 ans.





Ces formules n'appartiennent plus vraiment au groupe de compétences utilisateur débutant. Pour des affectations plus sophistiquées, veuillez vous référer à l'annexe A de ce manuel.

La formule dans le second champ de la requête ;

```
CASEWHEN( [AgeSport] > 19,
   CEILING( [AgeSport] / 10 ) * 10,
   [AgeSport] - MOD( [AgeSport], 2 ) )
```

(Les guillemets sont remplacés par des crochets, que Base traduit ensuite dans la cellule de champ)

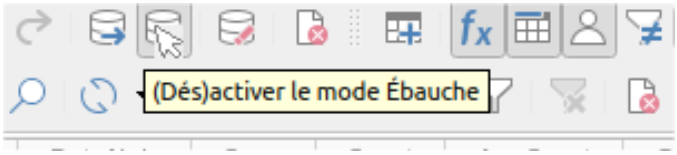
CASEWHEN(condition, valeur1, valeur2). Si condition est vraie, valeur 1 est retournée, autrement valeur2 est retournée. Valeur peut être un nombre ; une chaîne de caractère ; le résultat d'un calcul.. En français, cela signifie :

- Si AgeSport est supérieur à 19 ans, il est utilisé pour calculer l'âge inférieur suivant qui se termine par un zéro pour obtenir le Groupe_Age. CEILING("AgeSport" / 10) signifie : arrondi de la division de "AgeSport" par 10 (ex : 25/10=2,5 → Ceiling (2,5) = 2). Ce résultat est ensuite multiplié par 10, et on obtient 20.
- Si AgeSport est inférieur à 19 ans, calculer MOD ("AgeSport", 2) et le retrancher à la valeur de AgeSport. MOD("AgeSport", 2) signifie : Reste de la division (modulo) de AgeSport par 2 (ex : pour 17 / 2 le reste est 1).

Tous les candidats de plus de 19 ans sont ainsi affectés à une tranche d'âge tous les dix ans. Tous les candidats jusqu'à 19 ans sont attribués à un groupe d'âge d'amplitude 2 ans.

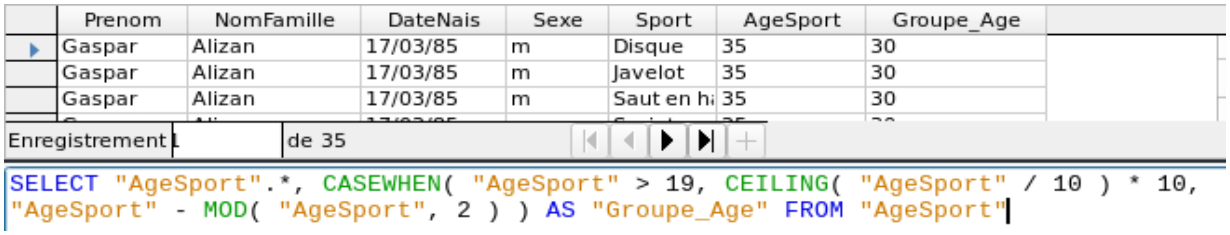
Cette formule a aussi reçu un alias, dans ce cas l'alias est Groupe_Age.

Enregistrez la requête sous le nom *Inscription*.



La désactivation du mode ébauche n'est pas vraiment nécessaire, car toutes les entrées sont possibles dans la vue de conception sans problème majeur. Cependant, regarder de plus près le code dans une requête ne peut jamais faire de mal.

Pour terminer, il existe des expressions SQL qui s'intègrent mal dans le mode Ébauche ou qui n'y sont pas réalisables. Dans ce cas on utilise l'entrée directe du code SQL.



Prenom	NomFamille	DateNais	Sexe	Sport	AgeSport	Groupe_Age
Gaspar	Alizan	17/03/85	m	Disque	35	30
Gaspar	Alizan	17/03/85	m	Javelot	35	30
Gaspar	Alizan	17/03/85	m	Saut en hi	35	30

Enregistrement 1 de 35

```
SELECT "AgeSport".*, CASEWHEN( "AgeSport" > 19, CEILING( "AgeSport" / 10 ) * 10,
"AgeSport" - MOD( "AgeSport", 2 ) ) AS "Groupe_Age" FROM "AgeSport"
```

Note

Les noms des champs et des tables sont entourés de guillemets doubles et affichés en ocre. Les mots-clés du code SQL sont en bleu, les fonctions en vert.

```
SELECT "AgeSport".*, CASEWHEN("AgeSport" > 19, CEILING("AgeSport" / 10) * 10,
"AgeSport" – MOD("AgeSport", 2)) AS "Groupe_Age" FROM "AgeSport"
```

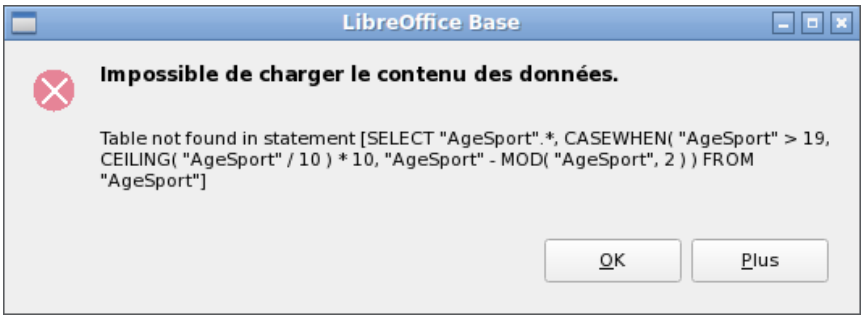
Les parties de la formule ont déjà été mentionnées. Voici maintenant la structure :

```
SELECT "AgeSport".*,... AS "Groupe_Age" FROM "AgeSport"
```

Sélectionnez (SELECT), dans `(FROM)` la table `AgeSport,` tous les champs de la table ("AgeSport".*), plus ce qui est retourné par la formule, dont l'alias (AS) est "`Groupe_Age`".

Le code ne fait pas la distinction entre les tables et les requêtes comme base des données. Il ne fonctionne donc que dans l'interface utilisateur graphique de Base. Une requête ne peut pas avoir le même nom qu'une table et réciproquement.

Lorsque l'utilisateur clique sur le bouton SQL (Exécuter directement l'instruction SQL) en mode Vue SQL , la base de données répond avec un message d'erreur. Le HSQLDB embarqué ne connaît pas la requête "AgeSport" à laquelle se réfère la requête en cours.



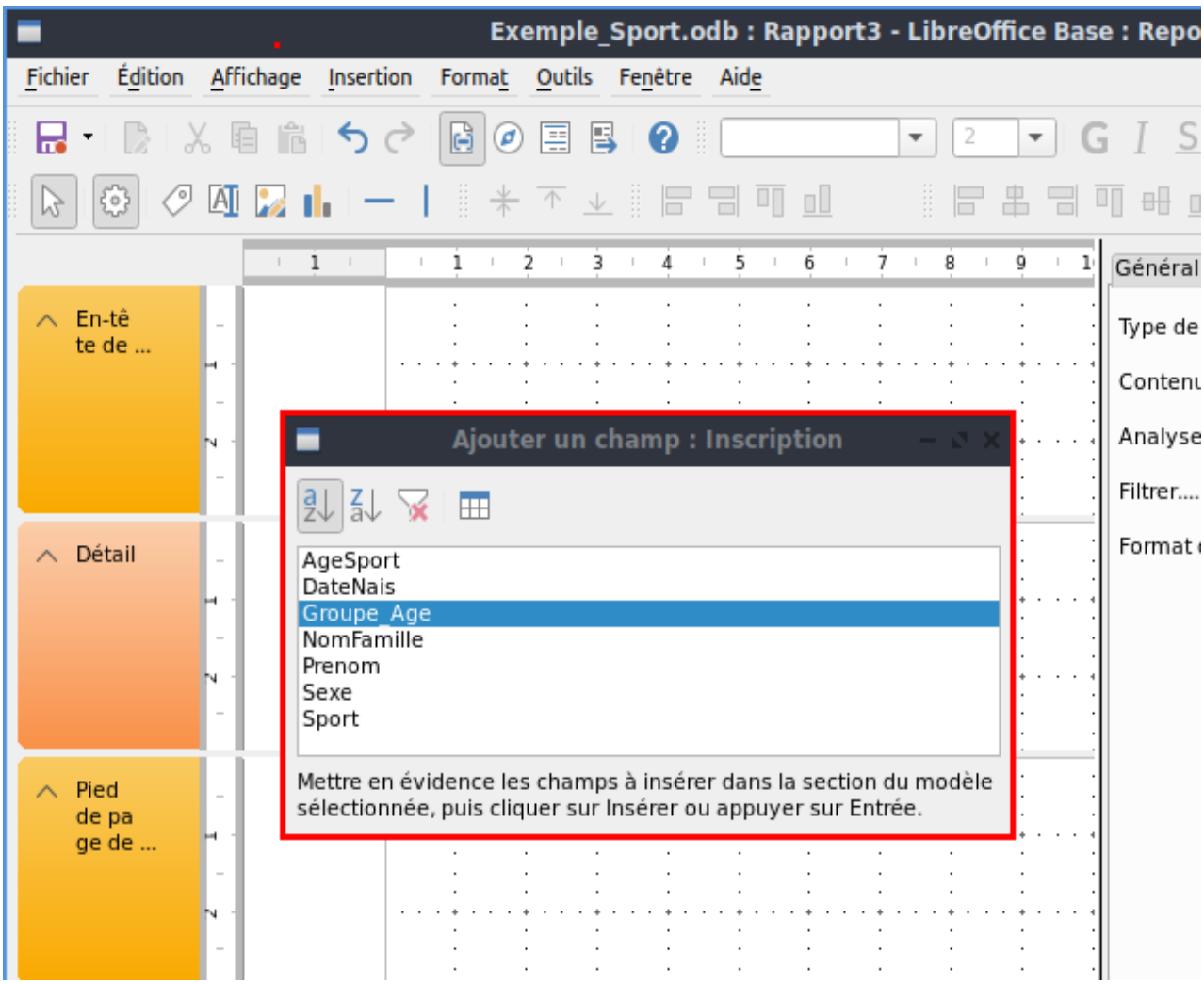
Créer un rapport

Utilisez **Rapports > Créer un rapport en mode Ébauche** pour créer un rapport avec une liste claire des entrées, triées par sport, sexe et tranche d'âge.

Au démarrage de l'éditeur, la boîte de dialogue **Ajouter un champ** apparaît d'abord au premier plan. Utilisez cette boîte de dialogue pour prendre comme base les champs de la table triée par ordre alphabétique. La requête doit être choisie comme source de données.

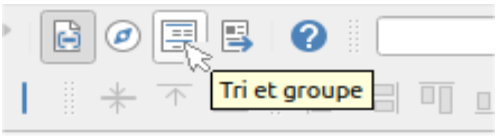
Dans la fenêtre de rapport, le côté droit montre un aperçu des propriétés de l'objet actuellement actif. S'il n'est pas visible, sélectionnez **Affichage > Propriétés**.

Dans Propriétés, sélectionnez l'onglet **Données > Type de contenu > Requête**. Pour la propriété Contenu sélectionnez la requête *Inscription*, qui était la requête récapitulative finale.



Le Générateur de rapports affiche désormais tous les champs de la requête sélectionnée.

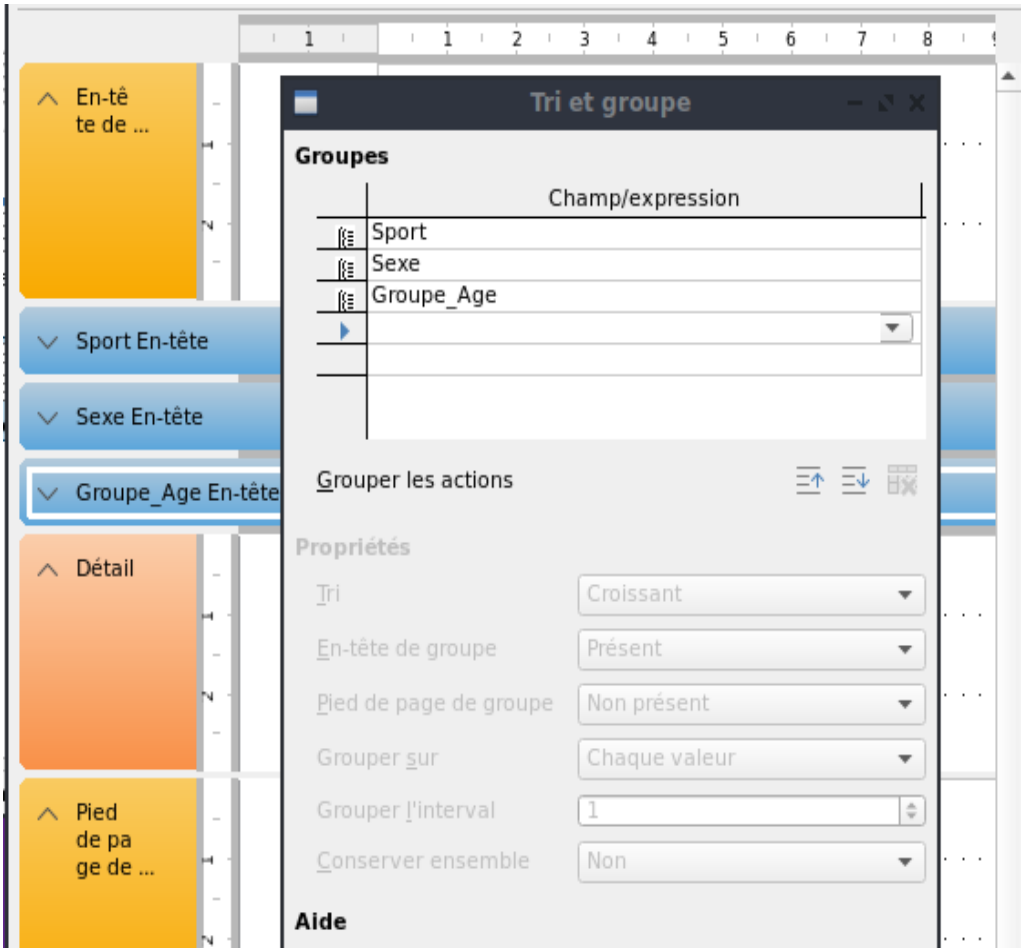
À l'étape suivante, ouvrez la boîte de dialogue **Tri et groupe** à l'aide du bouton de la barre d'outils ou en sélectionnant **Affichage > Tri et groupe**.



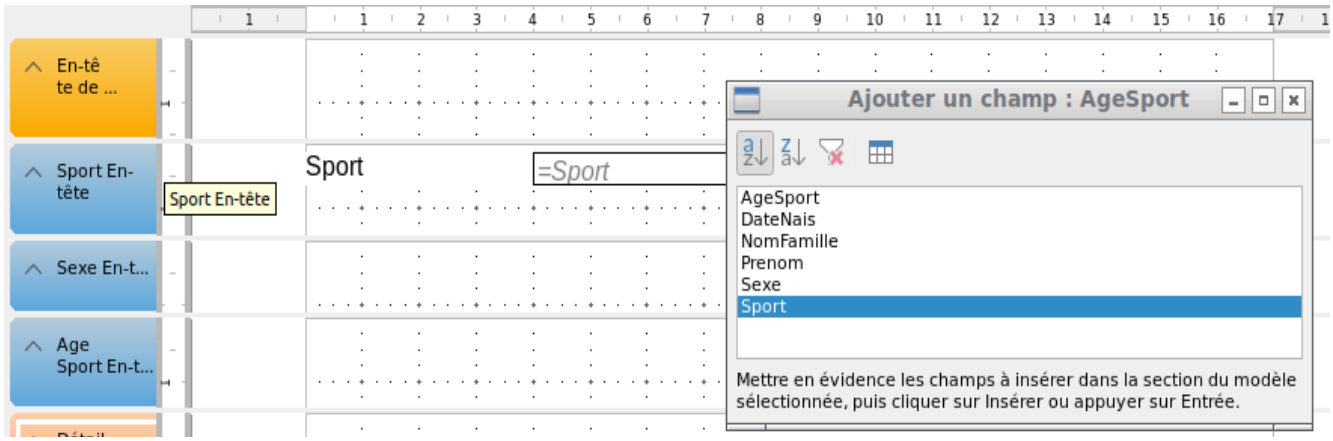
Dans la boîte de dialogue Tri et groupe, sélectionnez les champs Sport, Sexe et Groupe_Age.

Un en-tête de groupe pour le tri apparaît à chaque sélection de tri sur le côté gauche du rapport. Utilisez les paramètres par défaut pour les propriétés des sélections de tri et de regroupement.

Fermez la boîte de dialogue Tri et groupe.



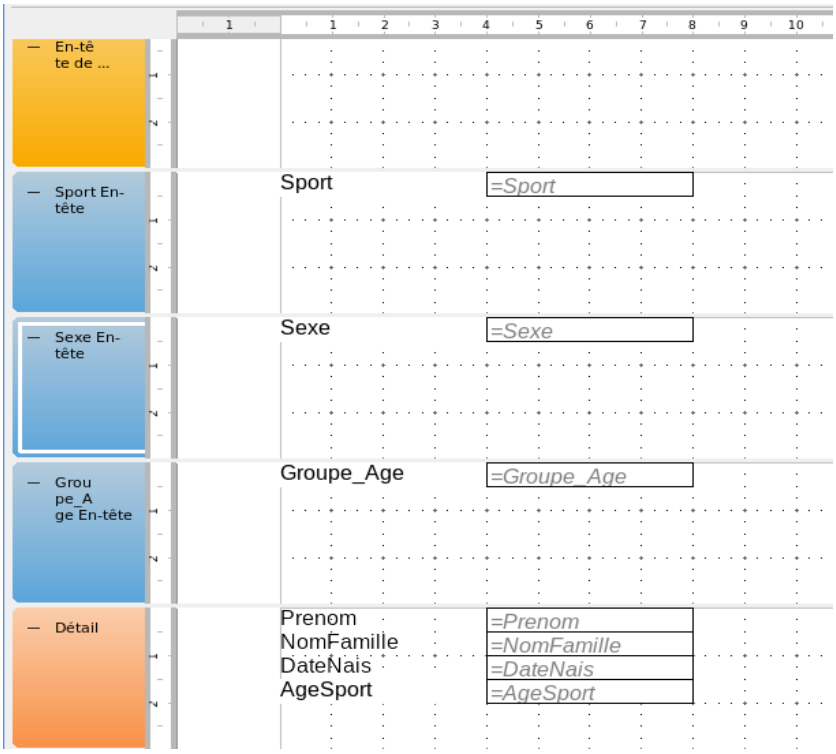
Sélectionnez l'en-tête de groupe, Sport. Ceci est visible dans la bordure blanche de l'en-tête. À l'aide de la boîte de dialogue Ajouter un champ, cliquez sur le champ de départ pour insérer un champ d'étiquette et un champ de texte qui afficheront le contenu du champ de candidat.



De la même manière, affectez les champs Sexe et Groupe_Age aux en-têtes de groupe appropriés.

Insérez tous les champs restants dans la zone Détails.

Le projet de rapport devrait maintenant ressembler à ceci :



Enregistrez le rapport. Le nom pourrait être quelque chose comme *Liste des participants*.

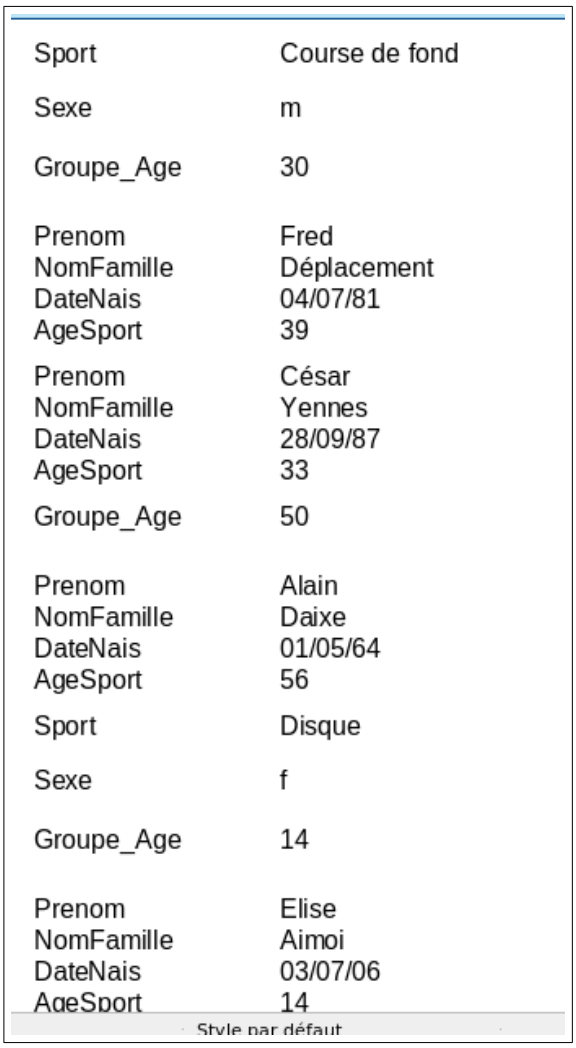
Ensuite, enregistrez le fichier de base de données lui-même ; sinon, le rapport n'est stocké que temporairement.

Note

Surtout lors de la conception d'un rapport à l'aide du Générateur de rapports, il échoue souvent en raison d'instabilités du programme. Il est important de sauvegarder régulièrement le rapport et le fichier de base de données.

Heureusement, l'exécution ultérieure d'un rapport n'est pas affectée par ces instabilités.

Si ce rapport est exécuté avec les données appropriées, le résultat ressemble à l'image suivante.



Le début du rapport montre deux candidats qui souhaitent se lancer dans la course de fond et appartiennent au groupe d'âge 30.

Lors de l'exécution du rapport, des défauts de conception apparaissent :

- Les distances entre les en-têtes de groupe et le contenu de la section Détail sont beaucoup trop grandes.
- Le sexe est indiqué par m et f. Mieux vaut des noms tels que Hommes et Femmes.
- Probablement en raison de l'opération de division dans la requête, le champ Groupe_Age contient des nombres avec une décimale.
- Les champs d'étiquette de la section Détail (prénom, nom, ...) seraient mieux placés ensemble horizontalement comme en-têtes de tableau sous l'étiquette et le champ Groupe_Age dans l'en-tête Groupe_Age.

Définition des distances entre les champs du rapport

Pour réduire les distances verticales entre les champs, utilisez la souris pour faire glisser la bordure inférieure de l'en-tête Groupe_Age vers le bas des en-têtes de tableau. Vous pouvez également mettre en évidence une section et régler la hauteur de l'en-tête de groupe dans **Propriétés**. Par conséquent, aucun champ mais uniquement l'en-tête de groupe ne doit être étiqueté.

Il n'est pas possible qu'une section soit plus petite que les étiquettes et les champs qu'elle contient.

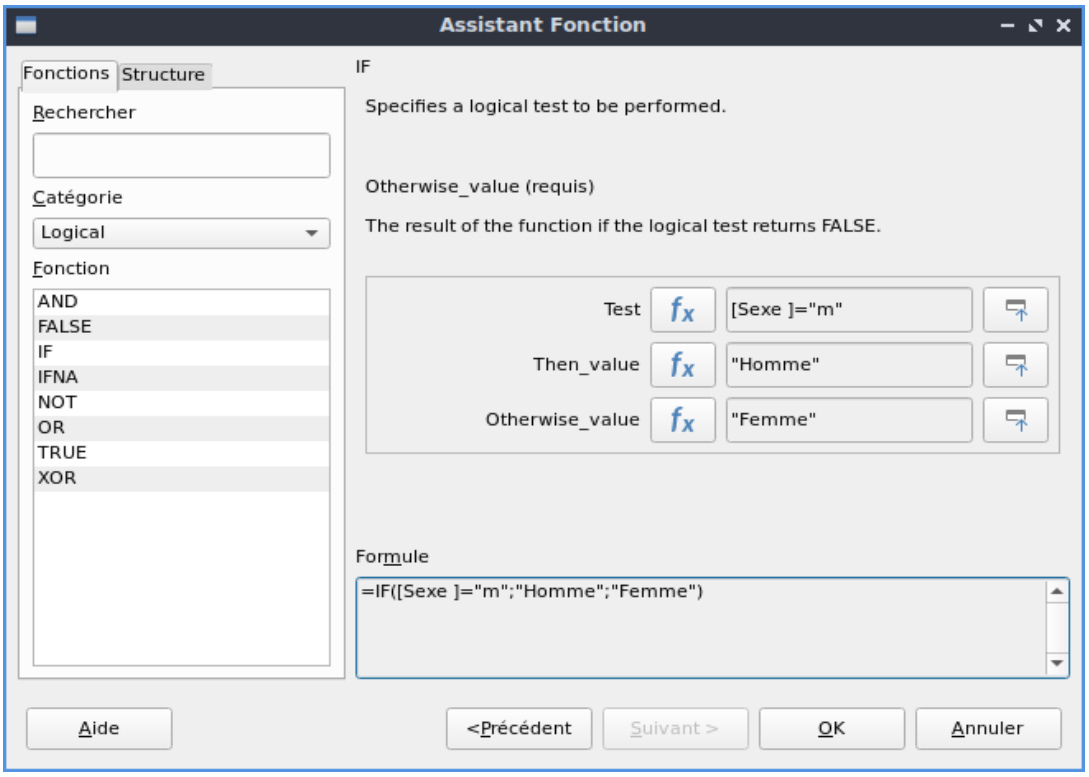
La zone d'en-tête Groupe_Age doit être suffisamment grande pour prendre les champs d'étiquette de la section Détail.

Tenez également compte, lors de la sélection des distances, qu'un groupe suivant n'apparaît pas trop près du groupe précédent. L'étiquette et les zones de texte peuvent également nécessiter une distance par rapport au haut de la section les contenant. Si cette distance au sommet n'est pas souhaitée, des pieds de page de groupe plutôt que des en-têtes peuvent être affichés pour fournir cette distance. Une telle préférence est possible dans **Affichage > Tri et groupe** pour chaque groupe.

Influencer le contenu d'un champ de texte par une formule

La désignation du sexe dans la table ne suffit pas pour les exigences d'une liste de candidats. Le changement de nom du champ peut être effectué dans la requête. Mais comme la requête a déjà été créée, vous pouvez utiliser des fonctions dans le rapport à la place.

Mettez en surbrillance le champ de texte "= Sexe". Dans les propriétés sur le côté droit de l'onglet Générateur de rapports, sélectionnez **Données > Champ de données**. Cliquez sur le bouton avec l'ellipse (…).L'assistant de fonction s'affiche.



Dans **Catégorie**, sélectionnez **Logique**, puis double-cliquez sur **IF**.

Saisissez la valeur de test. Il s'agit du champ de la requête à partir duquel les données sont lues et mises entre crochets. Les textes doivent être placés entre guillemets. Lorsque le champ Sexe a la valeur **m**, alors **Homme** est affiché dans le champ du rapport. S'il n'y a pas de m, alors **Femme** apparaît.

Cliquez sur **OK** pour confirmer la saisie.

Ainsi, le libellé du champ est modifié en conséquence.

Modifier la mise en forme d'un champ de texte

Le champ qui reçoit le contenu de la base de données est identifié dans le rapport comme une zone de texte, mais il peut être formaté comme les champs des tables dans Writer ou Calc.

Mettez en surbrillance le champ “= Groupe_Age”. Dans les propriétés sur la droite, sélectionnez **Général > Formatage (**dernière ligne). La propriété Mise en forme est définie sur “texte”. Cliquez sur le bouton avec l’ellipse (…). Cela ouvre la boîte de dialogue de mise en forme, également utilisée dans Calc, Writer ou lors de la création de formulaires. Sélectionnez **Catégorie > Nombre** et confirmez avec **OK.** Choisissez également le nombre de décimales.

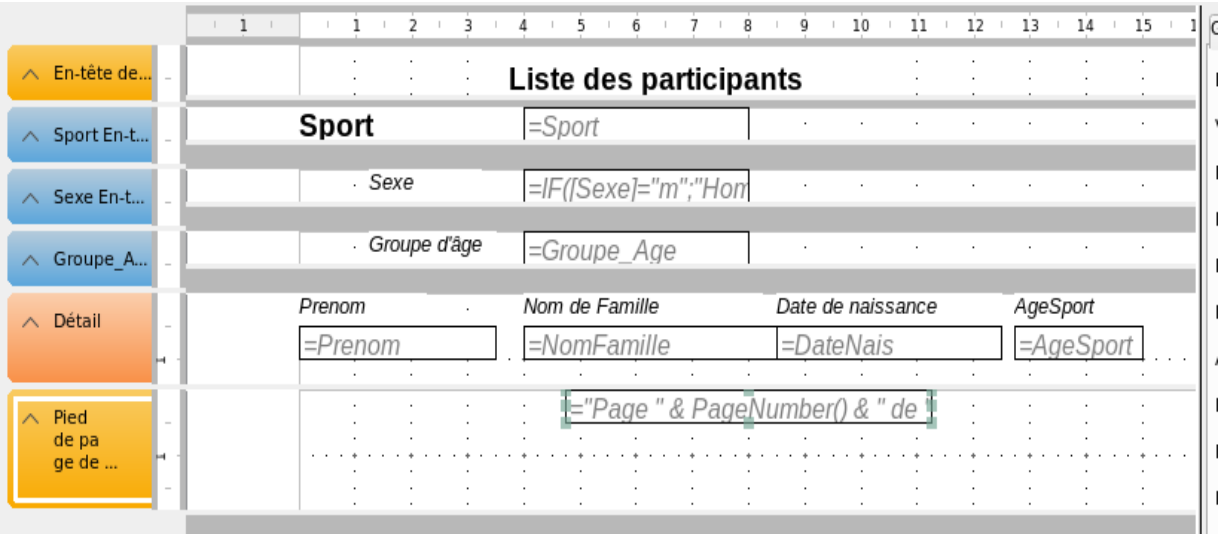
Le format de l’affichage passe désormais du texte aux nombres.

Déplacement des boîtes dans le générateur de rapports

Les champs peuvent également être déplacés au-delà des limites d’une section vers une autre section dans le Générateur de rapports. Cependant, un espace suffisant pour le champ doit être présent dans la section de destination. Aucune partie d’un champ ne peut exister au même emplacement qu’une partie d’un autre champ.

Le positionnement des champs à l’aide d’une souris n’est pas précis, vous devez donc prévoir suffisamment d’espace dans chaque section pour les champs. Dans une section comme l’en-tête de groupe Groupe_Age, les champs peuvent ensuite être positionnés avec précision à l’aide des touches fléchées.

Pour positionner les champs à l’aide du clavier, regardez les propriétés **Général > Position X** et **Général > Position Y**, déplacer le contrôle avec Maj+Flèches (grands déplacements) ou plus finement avec Alt+Flèches (petits déplacements).



Ici, un champ d’étiquette pour l’en-tête a été ajouté. L’entrée du texte pour le champ d’étiquette apparaît dans les propriétés des champs.

Le pied de page est défini dans les propriétés : **Visible > Oui**. La marge inférieure du document contient trop d’espace. N’oubliez pas que le montant disponible est déjà en plus réduit par la taille des marges de la page. Par défaut, toutes les marges de page sont définies pour le format de page lettre à 0,79 pouces (2 centimètres).

Pour plus d’options de mise en forme des rapports, reportez-vous au Chapitre 6, Rapports.

Liste des participants

Sport	Course de fond		
Sexe	Homme		
Groupe d'âge	30		
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
Fred	Déplacement	04/07/81	39
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
César	Yennes	28/09/87	33
Groupe d'âge	50		
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
Alain	Daixe	01/05/64	56
Sport	Disque		
Sexe	Femme		
Groupe d'âge	14		
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
Elise	Aimoi	03/07/06	14
Groupe d'âge	20		
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
Dolly	Prane	01/02/99	21
Groupe d'âge	30		
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
Clara	Hocquet	25/12/84	36
Groupe d'âge	40		
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
Amélie	Oration	07/04/78	42
Sexe	Homme		
Groupe d'âge	30		
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
Agathe	Zeublouse	06/07/89	31
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
César	Yennes	28/09/87	33
Prenom	*Nom de Famille*	*Date de naissance*	*AgeSport*
Gaspar	Alizan	17/03/85	35

Extensions de la base de données exemple

L'exemple présenté ici n'est que la première étape d'une base de données dans le secteur du sport. Maintenant, ajoutez un champ pour chaque élément possible d'informations nécessaires. Un bon endroit pour ajouter un tel champ est dans la table rel_Candidats_Sports.

Cependant, si les entrées s'appliquent à plusieurs compétitions pour le même sport, incluez un champ de date ou un autre champ qui peut être attribué à la compétition respective dans la table rel_Candidats_Sports. Le champ fait alors également partie de la clé primaire de la table.

Peut-être que l'adhésion au club pourrait également être ajoutée. L'ajout d'un champ dans la table Candidats serait suffisant. Lorsque de nombreux clubs ont le même nom, cela suggère d'ajouter une table Club séparée et également la clé étrangère correspondante dans la table Candidats.

Ensuite, bien sûr, il faut déterminer, comme pour toutes les compétitions, qui doit être placé dans quel groupe d'âge et quel sport pertinents. Le tri est nécessaire ici, ce qui peut à nouveau se transformer en un rapport avec une liste de résultats.

Ce serait bien si chaque candidat (encore une fois via un rapport) obtenait un certificat magnifiquement conçu avec la performance personnelle et le classement.

De telles améliorations sont facilement possibles, comme décrit dans d'autres chapitres de ce guide.