



Créer une base de données

Tout le monde a entendu parler de **bases de données**, mais sans trop savoir ce dont il s'agit réellement.

Si on parle de "Base de données des assurés sociaux", on se doute que tout un chacun y a sa place, et que cela permet de savoir qui a droit à quoi, suivre ses remboursements médicaux, et autres renseignements nécessaires.

Mais... comment cela fonctionne-t-il ?

Et moi même, puis-je avoir "ma" base de données ?

Table des matières [+]

- [Créer une base de données](#)
 - [Qu'est-ce qu'une base de données ?](#)
 - [Bien comprendre l'organisation d'une base de données](#)
 - [Ajouter une nouvelle table](#)
 - [Créer des relations entre les données des tables](#)
 - [Pourquoi faire une base de données ?](#)
 - [Quelle base de données ?](#)
 - [Le langage SQL](#)
 - [Comment créer une base de données ?](#)

- Access de Microsoft
 - OpenOffice ou LibreOffice
 - Apache OpenOffice
 - LibreOffice
 - Conclusion
-

Qu'est-ce qu'une base de données ?

Une **base de données (BD ou BDD)** est un ensemble structuré de données homogènes, stocké sur un support informatique, et permettant un accès simplifié et sécurisé à ces données *via* un logiciel.

? ensemble structuré :

Les données sont réparties dans des **tables**, comparables à des fichiers de données.

? données homogènes :

On ne "mélange" pas par exemple des données provenant de domaines différents.

? stocké sur un support informatique :

Une base de données se présente comme un simple et unique fichier, même si sa taille peut être gigantesque.

? accès simplifié :

Les logiciels envoient des **requêtes** à la base données, qui elle, renvoie les données demandées.

De plus, on peut créer entre les données, des **relations** : cela permet par exemple, à partir d'un numéro de Sécurité Sociale, d'extraire de la base de données tous les remboursements d'un patient, car ces données seront en relation, *via* le numéro de Sécurité Sociale.

De plus, ces relations permettent d'assurer l'**intégrité** de l'ensemble des données, en interdisant la création de données incohérentes entre elles.

Ce système remplace avantageusement ce qui existait avant, à savoir des fichiers de données, indépendants les uns des autres.

Cela demandait beaucoup de travail et de soin aux programmeurs, pour gérer, mettre à jour ou exploiter les données ainsi réparties, et la moindre erreur pouvait générer beaucoup de problèmes.

Par exemple, rajouter un champ à un fichier nécessitait d'écrire un programme permettant de créer le nouveau fichier (recopier les données, et rajouter le nouveau champ), puis supprimer l'ancien fichier, donner l'ancien nom du fichier au nouveau fichier, puis, ensuite, modifier TOUS les programmes utilisant ce fichier, sans parler des modifications induites sur les documents imprimés, les écrans de saisie ou formulaires...

? Si vous voulez approfondir vos connaissances sur ce sujet : [Page Wikipedia](#) .

Bien comprendre l'organisation d'une base de données

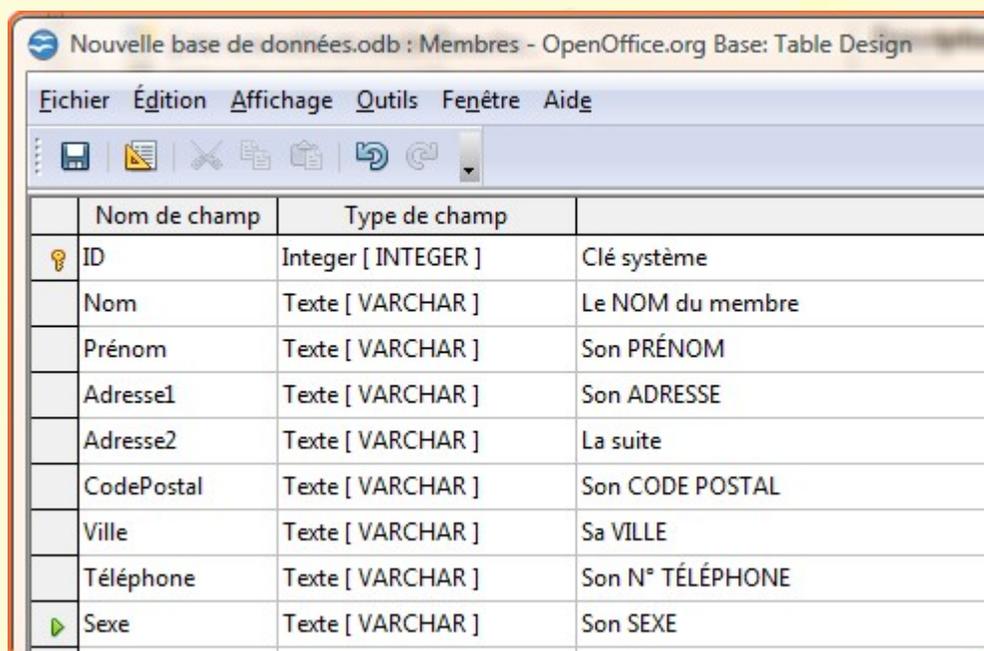
Prenons comme exemple, une base de données destinée à faciliter la gestion d'une association loi 1901.

Une association a des **membres**, et elle a besoin de certains renseignements les concernant : nom, prénom, adresse, téléphone...

Nous créerons donc une première **table** que nous appellerons... *Membres*.

Cette table "Membres" comportera donc les **champs** suivants :

1. *Nom* : zone de texte suffisamment grande (disons 40 caractères)
2. *Prénom* : idem
3. *Adresse1* et *Adresse2* : deux zones de texte (disons 40 caractères)
4. *Code postal* : zone de texte de 5 caractères
5. *Ville* : zone de texte de 40 caractères
6. *Téléphone* : zone de texte de 10 caractères



The screenshot shows the 'Table Design' window for a new database named 'Membres'. The window title is 'Nouvelle base de données.odb : Membres - OpenOffice.org Base: Table Design'. The menu bar includes 'Fichier', 'Édition', 'Affichage', 'Outils', 'Fenêtre', and 'Aide'. Below the menu is a toolbar with various icons. The main area displays a table with the following fields:

| | Nom de champ | Type de champ | |
|---|--------------|---------------------|------------------|
| 🔑 | ID | Integer [INTEGER] | Clé système |
| | Nom | Texte [VARCHAR] | Le NOM du membre |
| | Prénom | Texte [VARCHAR] | Son PRÉNOM |
| | Adresse1 | Texte [VARCHAR] | Son ADRESSE |
| | Adresse2 | Texte [VARCHAR] | La suite |
| | CodePostal | Texte [VARCHAR] | Son CODE POSTAL |
| | Ville | Texte [VARCHAR] | Sa VILLE |
| | Téléphone | Texte [VARCHAR] | Son N° TÉLÉPHONE |
| ▶ | Sexe | Texte [VARCHAR] | Son SEXE |

Voici notre table Membres !

Les tailles des champs sont définies dans une autre partie de l'écran.

Avec cette première table "Membres", nous pourrons déjà enregistrer les données 'papier' de nos adhérents !

Et le logiciel dont nous parlerons plus loin, va déjà nous permettre de saisir ces informations, très simplement.

Ah ! Mais ! Nous avons oublié une donnée importante, à savoir le genre ou le sexe de la personne !

Ce n'est pas un problème avec une base de données, il suffit de rajouter un champ supplémentaire, Sexe, zone texte de 1 caractère, pour M ou F, (Masculin ou Féminin).

La structure d'une table est modifiable à tout moment, ce qui est bien pratique !

Attention !



Le système nous oblige à ajouter un champ index, qui identifie chaque enregistrement en les numérotant en séquence. C'est ce qu'on appelle une clé unique.

Ajouter une nouvelle table

Maintenant que nous avons ces données à disposition, nous souhaitons mettre en place une deuxième table appelée **Sorties** : il s'agit des sorties organisées par l'association.

Cette table comprendra trois informations : la Date (de la sortie), un Titre (Par exemple : Musée - Rouen), une Description (détails).

Sa création ne pose pas de problème, et on peut saisir les sorties prévues.

Nouvelle base de données.odb : Sorties - OpenOffice.org Base: Table Design

Fichier Édition Affichage Outils Fenêtre Aide

| | Nom de champ | Type de champ | |
|--|--------------|---------------------|--|
| | ID | Integer [INTEGER] | |
| | Date | Date [DATE] | |
| | Titre | Texte [VARCHAR] | |
| | Description | Texte [VARCHAR] | |

Voici notre table Sorties !

Les tailles des champs sont définies dans une autre partie de l'écran.

Créer des relations entre les données des tables

Bien maintenant, nous devons savoir qui participe ou pas à telle ou telle sortie, bien sûr !

Pour cela, nous allons créer une troisième table, appelée **Listes**, qui va permettre de faire le lien entre Membres et Sorties. Elle comprendra deux champs : la Sortie (Numéro de la sortie), le Membre (N° du membre).

Nouvelle base de données.odb : Listes - OpenOffice.org Base: Table Design

Fichier Édition Affichage Outils Fenêtre Aide

| | Nom de champ | Type de champ | |
|--|--------------|---------------------|--|
| | ID | Integer [INTEGER] | |
| | Sortie | Integer [INTEGER] | |
| | Membre | Integer [INTEGER] | |

Voici notre table Listes !

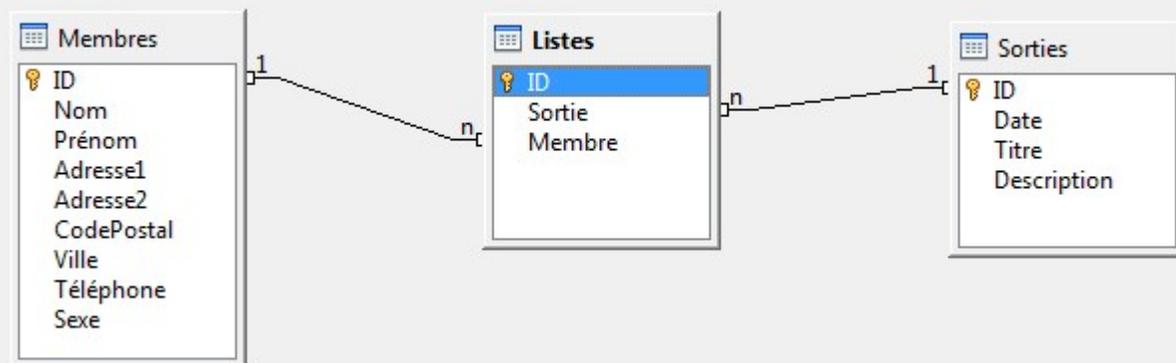
Les tailles des champs sont définies dans une autre partie de l'écran.

Et nous allons mettre en place une **relation** entre ces trois tables.

Les tables Membres et Listes sont liées par la clé du membre.

Les tables Sorties et Liste sont liées par la clé de la sortie.

Cette relation logique permet ensuite d'obtenir une liste des membres prévus pour une sortie.



Voici les relations entre nos tables !

Les mentions 1,n indiquent le type de correspondance.

Un membre (1) peut participer à plusieurs sorties (n) par exemple.

Ceci permet d'obtenir par exemple, cette **vue** des membres pour une sortie prévue.

| Date | Sortie | Nom | Prénom |
|----------|------------------|--------|----------|
| 27/01/13 | Aérodrome - Lens | DUPONT | Marcel |
| 27/01/13 | Aérodrome - Lens | MARTIN | Paulette |
| 27/01/13 | Aérodrome - Lens | HENRI | Jacques |

Voici la liste des membres pour une sortie prévue !

Cette vue peut accéder à l'ensemble des champs des trois tables.

Mais ce n'est pas tout !

Une base de données va nous permettre de faire très facilement des **formulaires de saisie**, mais aussi des **rapports** consultables à l'écran ou imprimables, de qualité professionnelle.

Pourquoi faire une base de données ?

Vous participez peut-être activement à la vie d'une association, d'un club, d'une amicale...
L'exemple ci-dessus vous a peut-être convaincu(e) que cet outil peut vous être utile, à plus d'un titre.

En tant que particulier, vous avez peut-être une collection (timbres-poste, disques, porte-clefs ou autres...).

Une base de données vous permettrait d'enregistrer tous vos objets collectionnés, y compris pour chacun, une photographie de l'objet.

C'est en effet possible !

Quelle base de données ?

Selon la nature de vos besoins, il y a le choix entre deux grandes familles de bases de données :

- Base de données locale
- Base de données distante

Une base de données locale se situe sur sa propre machine.

Via un **réseau local**, d'autres utilisateurs peuvent s'y connecter si cette base est **partagée**.

Les formats les plus accessibles aux néophytes sont [Access](#) (payant) et [OpenOffice](#) (gratuit) ou [LibreOffice](#) (gratuit également).

Une base de données distante se situe sur un serveur accessible via Internet.

Les formats sont nombreux et variés, mais ne sont guère accessibles aux néophytes.

En effet, pour pouvoir les utiliser, il faut aller beaucoup plus loin dans la programmation...

Le langage SQL

Le langage **SQL** (*Structured Query Language*) est un langage quasi-universel d'interrogation de bases de données.

C'est un langage qui simplifie grandement l'accès aux bases de données.

Les outils évoqués plus haut traduisent nos demandes en langage SQL.

Par exemple, notre liste des membres pour une sortie s'écrit, en SQL :

```
SELECT      "Sorties"."Date",      "Sorties"."Titre"      AS      "Sortie",  
"Membres"."Nom",      "Membres"."Prénom" FROM      "Listes",      "Sorties",  
"Membres"      WHERE      "Listes"."Sortie"      =      "Sorties"."ID"      AND  
"Listes"."Membre" = "Membres"."ID"
```

Où les mots signifient :

- SELECT : Sélectionner
- AS : en tant que, comme, nommés
- FROM : à partir des tables suivantes
- WHERE : où, répondant à la condition qui suit
- AND : et (logique)
- Sorties, Membres, Listes les noms de nos tables
- Date, Tire, Nom, Prénom... : les noms des champs dans les tables concernées.

Comment créer une base de données ?

Nous adressant par définition à des néophytes, nous ne parlerons ici que des solutions qui sont à la portée des particuliers ou d'associations au budget toujours limité.

Nous resterons volontairement sur des solutions simples et économiques, qui permettent de créer et de gérer une base de données locale, pour un usage personnel ou dans le cadre d'une association.

D'autres logiciels commerciaux existent comme FileMaker Pro (+400€), 4D (+300€)...

D'autres logiciels libres aussi mais moins complets (Annuaire, Base 6, SmartDb...).

Access de Microsoft

Access fait partie de la suite Microsoft Office, mais peut être acquis séparément.

[À cette adresse](#), vous pouvez consulter une vidéo de présentation et de prise en main de ce logiciel très complet.

C'est un logiciel payant (189€), mais il est proposé à un prix plus abordable ici : [Access 2010](#)

OpenOffice ou LibreOffice

Ces deux suites bureautiques, issues d'un même projet, comportent toutes deux un gestionnaire de bases de données.

La scission récente du projet initial ne permet pas de dire quel sera l'avenir de ces logiciels.

À ce jour, LibreOffice semble plus dynamique quant aux évolutions ; OpenOffice de son côté s'appuie sur la logistique et la puissance du projet Apache (gestion de serveurs web).

Il vous appartient de faire votre propre choix entre ces deux solutions, qui toutes deux, nécessitent la plateforme [Java](#) sur votre ordinateur.

Apache OpenOffice

[Télécharger Apache OpenOffice.](#)

LibreOffice

[Télécharger LibreOffice.](#)

Conclusion

Nous espérons que vous avez compris l'intérêt d'une base de données.

Il ne vous reste plus qu'à vous lancer sur le sujet, et vous apprécierez rapidement cet outil !

© <http://Ordi-Senior.fr> - 2010:2012

Toute reproduction est interdite sauf autorisation expresse du propriétaire du site.