

# Chapitre V : Les bases de données

---

## Introduction

La prise en compte de masse d'informations importantes dans des environnements riches soulève deux principales difficultés :

1. Maîtrise de la représentation de données complexes :

Comment représenter des informations très diverses, très complexes, relevant de différents domaines (à l'intérieur de l'entreprise) et malgré tout interdépendantes.

Ainsi, en tant qu'étudiant, vous appartenez au Système d'Information de l'université, comme Hervé Martin en tant qu'enseignant... or il existe bien un lien entre ces deux informations, lien qui passe par d'autres éléments : cours, TP, formation...

2. Maîtrise des accès personnalisés :

Comment mettre à disposition les informations dont un utilisateur a besoin sans le noyer sous des tonnes de données, sans l'empoisonner avec des données dont il n'a que faire, et en garantissant le bon usage des informations. Ainsi, les étudiants peuvent avoir accès à leur résultat d'examen, mais ils ne peuvent pas les modifier.

## Définition

Une **base de données** est un ensemble structuré de données enregistrées dans un ordinateur et accessibles de façon sélective par plusieurs utilisateurs.

Une base de données (son abréviation est BD, en anglais DB *database*) est une entité dans laquelle il est possible de stocker des données de façon structurée et avec le moins de redondance possible. Ces données doivent pouvoir être utilisées par des programmes, par des utilisateurs différents. Ainsi, la notion de base de données est généralement couplée à celle de réseau, afin de pouvoir mettre en commun ces informations, d'où le nom de base. On parle généralement de système d'information pour désigner toute la structure regroupant les moyens mis en place pour pouvoir partager des données.

## Les avantages d'une base de données

Pourquoi préférer alimenter une base de données plutôt qu'un tableur (sous Excel par exemple) ou qu'un document (sous Word par exemple) ? Les bases de données sont bien plus efficaces lorsqu'il s'agit de retrouver des informations. Vous pouvez poser une question à une base de données, par exemple : « Combien de personnes ont accepté de participer à l'événement que j'organise » ou « Quel client a acheté ce produit », et recevoir immédiatement la réponse.

Tout comme les tableurs, les bases de données organisent vos informations en listes. Cependant, contrairement aux tableurs, vous pouvez les afficher sous forme de formulaires. Un formulaire ressemble tout simplement à un formulaire papier et peut être mis en ligne. Les formulaires permettent d'accéder en toute liberté à tous les détails enregistrés. Grâce à eux, vous pouvez par exemple afficher tous vos projets dans une liste, puis cliquer pour consulter les détails d'un projet en particulier. Les formulaires facilitent également la saisie des données.

Qui dit bases de données, dit également précision. Dotées de listes déroulantes attractives, de rubriques renseignées automatiquement et de bien d'autres fonctions encore, les bases de données accélèrent la saisie des données et veillent à leur cohérence.

Utilisez une base de données pour relier vos informations. Imaginons que vous ayez une liste de clients et une liste d'articles vendus. En connectant (ou en « reliant ») ces listes, vous pourrez afficher tous les articles achetés par un client en particulier. Libre à vous, par la suite, de relier les enregistrements de la base de données à des fichiers, des photos, des films, mais aussi des pages Web, pour tout centraliser. Essayez donc de faire la même chose avec un tableur !

Les bases de données font bien plus que contenir des données. Elles aident à résoudre des problèmes, à répondre à des questions et à prendre des décisions. Vous pouvez créer des formules et des calculs pour vous aider dans vos analyses, puis générer rapidement des rapports pour partager votre point de vue.

La souplesse des impressions est l'un des autres points forts des bases de données. Qu'il s'agisse d'imprimer des étiquettes de publipostage, des rapports, des factures, des badges en vue de l'organisation d'une manifestation ou de formulaires papier, tout est plus simple avec les bases de données.

## **SGBD (système de gestion de bases de données)**

Un SGBD (en anglais DBMS pour *Data Base Management System*) permet d'inscrire, de retrouver, de modifier, de trier, de transformer ou d'imprimer les informations de la base de données. Il permet d'effectuer un compte-rendu des informations enregistrées et comporte des mécanismes pour assurer la cohérence des informations, éviter des pertes d'informations due à des pannes, assurer la confidentialité et permettre son utilisation par d'autres logiciels. Ses principaux rôles sont :

- Décrire les données qui seront stockées.
- Manipuler ces données (ajouter, modifier, supprimer des informations).
- Consulter les données et traiter les informations obtenues (sélectionner, trier, calculer, agréger,...).
- Définir des contraintes d'intégrité sur les données (contraintes de domaines, d'existence,...).
- Définir des protections d'accès (mots de passe, autorisations,...).
- Résoudre les problèmes d'accès multiples aux données.
- Prévoir des procédures de reprise en cas d'incident (sauvegardes, journaux,...).

La plupart des SGBD sont basés sur un modèle **Client - Serveur** (voir Figure 40). C'est-à-dire que la base de données se trouve sur un serveur qui ne sert qu'à ça, et pour interagir avec cette base de données, il faut utiliser un logiciel "client" qui va interroger le serveur et transmettre la réponse que le serveur lui aura donnée. Le serveur peut être installé sur une machine différente du client ; c'est souvent le cas lorsque les bases de données sont importantes.

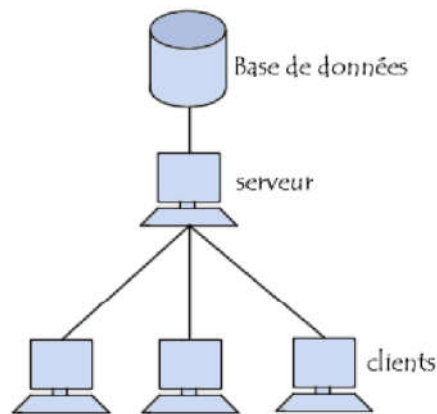


Figure 40: Exemple d'une architecture contenant une base de données

## Les usagers du SGBD

Il existe plusieurs catégories d'usagers des SGBD, parmi lesquelles il y a l'administrateur de bases de données, le concepteur de base de données, le développeur, ainsi que les utilisateurs - plus ou moins avisés - des systèmes informatiques.

- **L'administrateur de bases de données** (anglais *database administrator* abr. DBA) est un expert en SGBD, il s'occupe d'installer et de maintenir le SGBD ainsi que les outils annexes qui l'accompagnent. Il est la personne responsable de l'intégrité, de la sécurité, de la disponibilité des informations contenues dans les bases de données ainsi que de la performance du SGBD. Il protège les informations contre les accidents dus à des mauvaises manipulations, des erreurs de programmation, des utilisations malveillantes ou des pannes qui entraîneraient des détériorations du contenu des bases de données. Pour ce faire, l'administrateur de base de données autorise ou interdit l'accès aux informations et surveille l'activité du SGBD. Il effectue régulièrement des copies de sauvegarde en vue de permettre la récupération de données qui ont été perdues ou détériorées. L'administrateur utilise les outils d'administration de base de données ou le langage de commande du SGBD.
- **Le concepteur de base de données** (anglais *database designer*) est la personne qui identifie les informations qui seront enregistrées dans la base de données, les relations entre ces informations et les contraintes telles que la présence ou non de redondance. Le concepteur de base de données a une connaissance approfondie de l'usage qui est fait de ces informations et des règles qui en découlent. Il est chargé d'organiser la base de données de manière appropriée en mettant en place les structures nécessaires au stockage des informations.

- Les **développeurs** créent des logiciels applicatifs dans un langage de programmation de haut niveau. Le logiciel comporte des instructions qui font appel au SGBD pour rechercher ou modifier les informations.
- Les **utilisateurs avisés** ont les connaissances nécessaires pour accéder aux données à partir de leur ordinateur personnel. Ils ont été autorisés par l'administrateur à voir certaines informations et à les modifier. Les utilisateurs avisés peuvent rechercher, ajouter, modifier ou supprimer des données.
- Les **utilisateurs profanes** accèdent aux informations à travers un logiciel applicatif. Ils exécutent des commandes ou choisissent des menus. Les opérations effectuées par ces utilisateurs sont moins sophistiquées et limitées aux possibilités offertes par le logiciel applicatif.

## Construction et fonctionnement d'un SGBD

Un SGBD est composé de nombreux programmes, parmi lesquels le moteur, le catalogue, le processeur de requêtes, le langage de commande et des outils :

- le moteur de base de données est le cœur du SGBD, il manipule les fichiers de la base de données, transmet les données de et vers les autres programmes, et vérifie la cohérence et l'intégrité des données ;
- un programme qui manipule le catalogue : le magasin qui contient la description de l'organisation de la base de données, les listes de contrôle d'accès, le nom des personnes autorisées à manipuler la base de données et la description des règles de cohérence (contraintes). Selon les modèles de SGBD ces informations peuvent être modifiées en utilisant le langage de commande, ou alors à l'aide d'une interface graphique ;
- le processeur de requête exécute les opérations demandées. Selon le modèles de SGBD, ces opérations peuvent être formulées dans un langage de commande, ou à l'aide d'une interface graphique du type QBE (Query by Example, en français requête par l'exemple) ;
- la majorité des SGBD comportent au moins un langage de commande. Ce langage de requête permet de manipuler le contenu de la base de données. Reconnu par la majorité des SGBD du marché, SQL est devenu le langage standard.
- Les outils du SGBD servent à créer des compte-rendus (reports), des écrans pour la saisie des informations, importer et exporter les données de et vers la base de données, et manipuler le catalogue. Ces outils sont utilisés par l'administrateur de bases de données pour effectuer des sauvegardes, des restaurations de données, autoriser ou interdire l'accès à certaines informations, et effectuer des modifications du contenu de la base de données - création, lecture, modification et suppression d'informations, abrégé CRUD (anglais create, read, update, delete). Ces outils servent également à surveiller l'activité du moteur.

## Langage de requêtes SQL

Le SQL (Structured Query Language) est un langage permettant de communiquer avec une base de données (BD). Ce langage informatique est très utilisé par les développeurs Web pour communiquer avec les données d'un site Web. Il permet la lecture, l'insertion et la suppression des données dans une base de données. SQL peut également être utilisé pour gérer des BD, c'est-à-dire créer, supprimer ou mettre à jour des bases de données.

## SQL SELECT

La commande SELECT permet de lire des données d'une base et de retourner des enregistrements dans un tableau de résultat. Cette commande peut sélectionner une ou plusieurs colonnes d'une table. La forme générale de SELECT est la suivante :

```
SELECT nom_du_champ
FROM table
WHERE une ou plusieurs conditions
```

**Exemple** une base de données appelée « Client » qui contient des informations sur les clients d'une entreprise.

Identifiant	Prénom	Nom	Ville	Salaire
1	Pierre	Dupont	Paris	2000\$
2	Marine	Durant	Nantes	1000\$
3	Pierre	Martin	Marseille	1500\$

SELECT ville FROM client ;

Ville
Paris
Nante
Marseille

SELECT prénom, nom FROM client;

Prénom	Nom
Pierre	Dupont
Marine	Durant
Pierre	Martin

Pour sélectionner toutes les colonnes d'une table, on écrit :

SELECT \* FROM client ;

SELECT DISTINCT prénom FROM client;

Prénom
Pierre
Marine

Trouver la liste des clients qui habitent à Paris :

SELECT \* FROM client WHERE Ville = 'Paris';

Trouver le salaire maximal des clients :

SELECT MAX(Salaire) FROM client ;

On peut également utiliser MIN, AVG, COUNT, SUM

Il existe plusieurs **opérateurs de comparaisons** qu'on peut utiliser dans WHERE tels que =, <>, !=, >, <, >=, <=, **IN**, **BETWEEN**, **NOT BETWEEN**, **LIKE** et **IS NULL**.

Pour construire des conditions complexes dans la clause WHERE, on peut utiliser **OR** ou **AND**.

Exemples :

```
SELECT *
FROM client
WHERE Ville = 'Paris' OR Ville = 'Nantes' ;

SELECT prénom
FROM client
WHERE Prénom = 'Pierre' AND Salaire > 1500;
```

```
SELECT prénom
FROM client
WHERE      nom = 'Dupont' OR
           nom = 'Durant' OR
           nom = 'Matrin' ;
```

cette requête est équivalente à la requête suivante :

```
SELECT prénom
FROM client
WHERE nom IN ('Durant', 'Dupont', 'Martin');
```

```
SELECT id, nom
FROM client
WHERE Salaire BETWEEN 100 AND 1000 ;

SELECT id, nom
FROM client
WHERE Salaire NOT BETWEEN 100 AND 1000 ;
```

**SELECT** prénom, nom

**FROM** client

**WHERE** nom **LIKE** "%D";

c.à.d. dont le nom se termine par D.

**SELECT** prénom, nom

**FROM** client

**WHERE** nom **LIKE** "D%";

c.à.d. dont le nom commence par D.

**SELECT** prénom, nom

**FROM** client

**WHERE** nom **LIKE** "%D%";

c.à.d. dont le nom se termine par D.

**SELECT** prénom, nom

**FROM** client

**WHERE** nom **LIKE** "%D%";

c.à.d. dont le nom contient le caractère D.

**SELECT** prénom, nom

**FROM** client

**WHERE** nom **LIKE** "D%t";

qui commence par D et se termine la lettre t.

11100

```
SELECT prénom, nom  
FROM client  
WHERE nom LIKE "A_C";
```

le caractère “\_” underscore sera remplacé par n’importe quel caractère tel que ABC, AEC, etc.

```
SELECT COUNT(*)  
FROM client ;
```

c.à.d. compte le nombre d’enregistrements dans la table la réponse = 3

```
SELECT COUNT(prénom)  
FROM client ;
```

c.à.d. compte le nombre de clients ayant un prénom la réponse =3, un attribut différent de la clé primaire peut ne pas avoir une valeur (NULL) dans une table.

```
SELECT COUNT(DISTINCT prénom)  
FROM client ;
```

la réponse = 2

```
SELECT SUM(Salaire)  
FROM client;
```

la réponse = 4500\$

```
SELECT AVG(Salaire)  
FROM client  
WHERE prénom = 'Pierre';
```

la réponse = 1750\$.

## Etude de cas – Access (A faire dans la classe)

### Exercices

**Exercice 1:** On considère la base de données relationnelle (DB) stockant des informations sur les étudiants de Master et leurs superviseurs. La DB est constitué d'une seule relation (tableau) appelée R :

ID_Etud	Nom_Et	Age_Et	Ville_Et	Spécialité	Nom_supervis	Ville_supervisors	Date_Fin
123	Avi	23	Zahlé	Economie	Samuelson	Beyrouth	02/01/2016
123	Avi	23	Beyrouth	Math	Nash	Byblos	10/10/2014
456	Darin	28	Tripoli	Math	Nash	Byblos	12/12/2015
789	Len	25	Baalbek	Physique	Einstein	Princeton	01/12/2015
999	Richard	26	Zahlé	Physique	Aline	Zahlé	31/04/2003

Ecrire des requêtes SQL afin correspondantes aux questions suivantes :

- 1) Trouver les noms des étudiants qui habitent à Zahlé ou ayant la spécialité Math.
- 2) Trouver le nom et la ville des superviseurs qui encadrent des étudiants qui habitent à Baalbek.
- 3) Donner toutes les informations concernant l'étudiant le plus âgé.
- 4) Combien d'étudiants sont inscrits en Physique ?
- 5) Trouver le nom et la ville des étudiants supervisés par Nash.
- 6) Trouver les noms des étudiants et superviseurs qui habitent dans la même ville.
- 7) Donner la spécialité des étudiants qui habitent dans une ville dont le nom commence par la lettre 'Z' et qui sont âgés de plus de 23 ou bien dont le nom de la ville de leur superviseur se terminant par la lettre 's' ou bien qui soutiennent leur Master l'année prochaine.
- 8) Donner les villes des superviseurs qui supervisent des étudiants qui habitent dans des villes dont le nom commence par la lettre 'B'.
- 9) Trouver la spécialité des étudiants et le nom de leurs encadrants qui soutiennent leur Master en 2015.
- 10) Trouver les informations des étudiants âgés de 26, 28 ou 25 et qui habitent à Tripoli.
- 11) Donner le nom des étudiants et de leur spécialité pour ceux qui ont déjà soutenu leur Master.

**Exercice 2 :** Soit une base de données relationnelle stockant les informations concernant les livres empruntés des bibliothèques par des étudiants. Elle est composée de trois relations nommées Livre, Étudiant et Bibliothèque présentées ci-dessous :

Nom_Bib	Adresse_Bib
Du Liban	Zouk Mosbeh
Antoine	Sin el Fil
LDLP	Beyrouth

ID_Livre	Titre	Auteur	Catégorie	Année	Nom_bib
1	AA	X	Fiction	2000	Du Liban
2	BB	Y	Nature	2010	Antoine
3	CC	X	Arts	2008	Antoine
4	DD	Y	Fiction	2010	Du Liban
5	EE	Z	Cuisine	2015	Antoine

ID_Etudiant	Spécialité	Genre
123	Mathématique	Masculin
124	Informatique	Féminin
125	Physique	Masculin

Exprimez en français et donnez le résultat de l'exécution de chacune des requêtes suivantes :

- 1) SELECT COUNT(\*)  
FROM Étudiant  
WHERE Genre = 'Masculin' AND spécialité = 'Physique'
- 2) SELECT Titre  
FROM Livre  
WHERE Année BETWEEN 2008 AND 2010 OR Catégorie = 'Fiction'
- 3) SELECT DISTINCT Auteur  
FROM Livre  
WHERE Nom\_bib LIKE 'A%e'
- 4) SELECT Titre, Catégorie  
FROM Livre  
WHERE Catégorie IN ('Fiction', 'Arts', 'Cuisine') AND Nom\_bib = 'Du Liban'
- 5) SELECT COUNT(DISTINCT Nom\_bib)  
FROM Livre  
SELECT COUNT(DISTINCT Nom\_Bib)  
FROM Bibliothèque
- 6) SELECT \*  
FROM Livre  
WHERE Annee = 2010 AND Nom\_bib = 'Du Liban'
- 7) SELECT \*  
FROM Livre  
WHERE Annee = 2010 OR Nom\_bib = 'Du Liban'

11100

- 8) SELECT Auteur  
FROM Livre  
WHERE Catégorie < > 'Fiction' AND Année > 2008
  
- 9) SELECT MAX(ID\_Etudiant)  
FROM Etudiant  
WHERE Genre='Féminin'
  
- 10) SELECT Nom\_Bib, Adresse\_Bib  
FROM Bibliothèque  
WHERE Nom\_Bib NOT IN (SELECT Nom\_bib FROM Livre)