

SQL

Un langage de définition
et manipulation de base de données

Nous vivons dans une ère de l'information

- ✓ La quantité de données produite chaque année ne cesse de croître
- ✓ Il n'a jamais été aussi facile et bon marché d'obtenir de l'information
- ✓ L'enjeu n'est plus la quantité mais l'organisation et la pertinence

- ✓ Stocker l'information de façon durable
- ✓ Assurer la cohérence et l'intégrité de l'information
- ✓ Extraire une partie de l'information
- ✓ Sécuriser l'accès à l'information

Fichier texte

Données peu ou pas structurées
Pas de contrôle d'intégrité
Accès séquentiel
Sécurité très faible
Traitements limités
Accès simultanés problématiques
Taille limitée par le système de fichier

Tableur

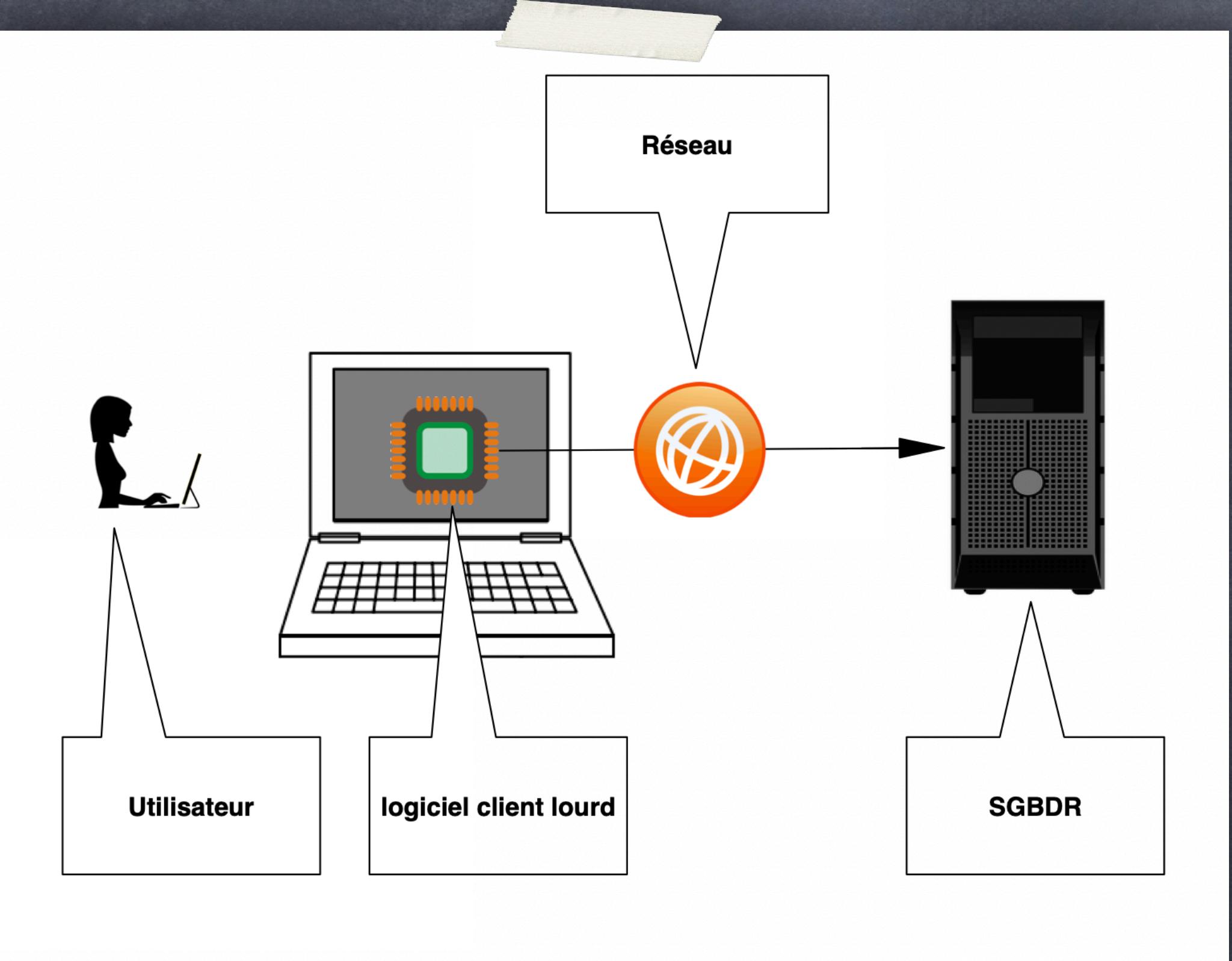
Données plus structurées
Peu de contrôle d'intégrité
Accès toujours séquentiel
Sécurité faible
Quelques traitements intégrés
Accès simultanés problématiques
Taille limités par le système de fichier et la RAM

SGBDR

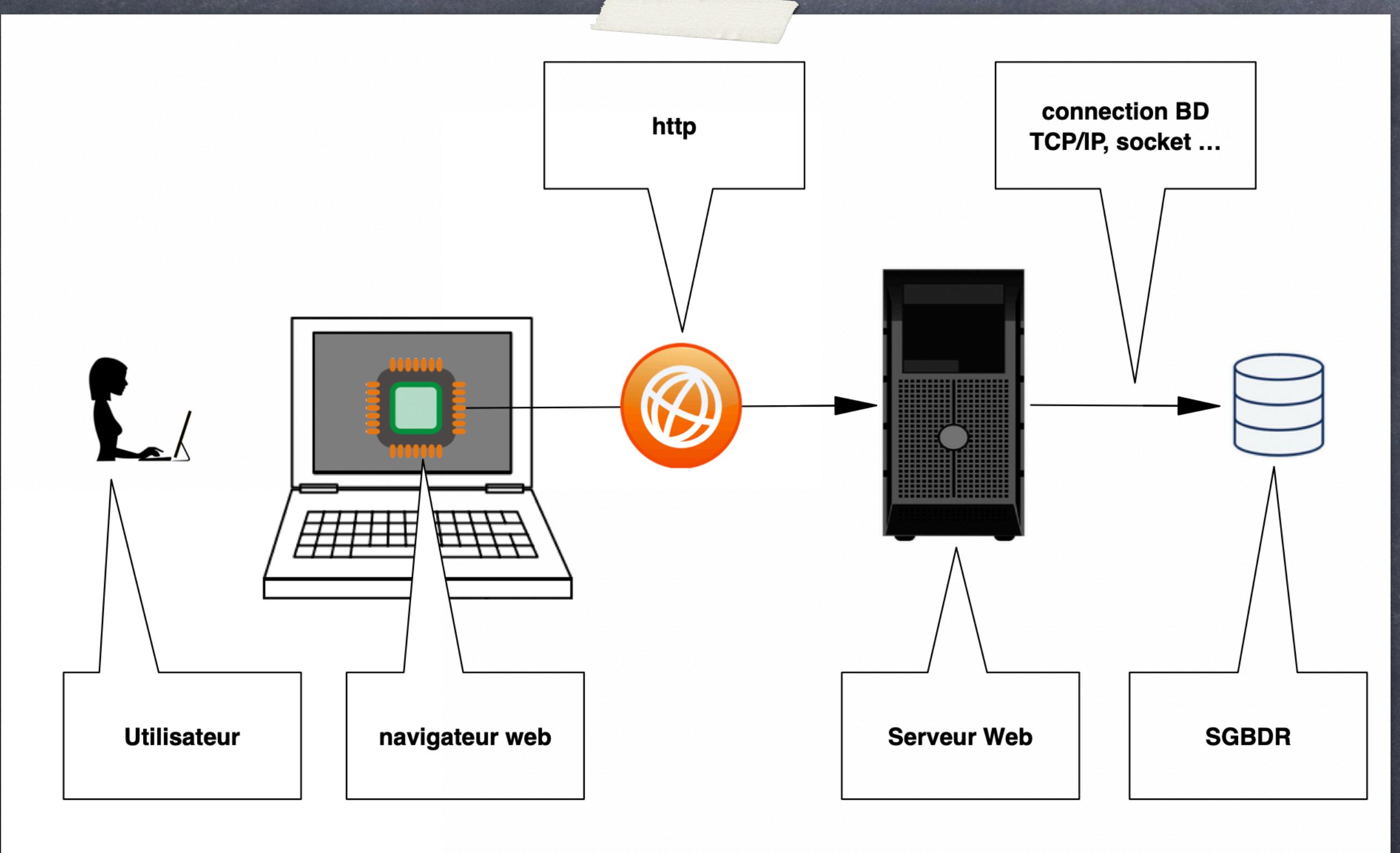
Données structurées
Contrôle de l'intégrité des données
Accès non séquentiel
Bonne sécurité
Traitements intégrés
Peut gérer un grand nombre d'accès
Peut gérer une très grande quantité de données

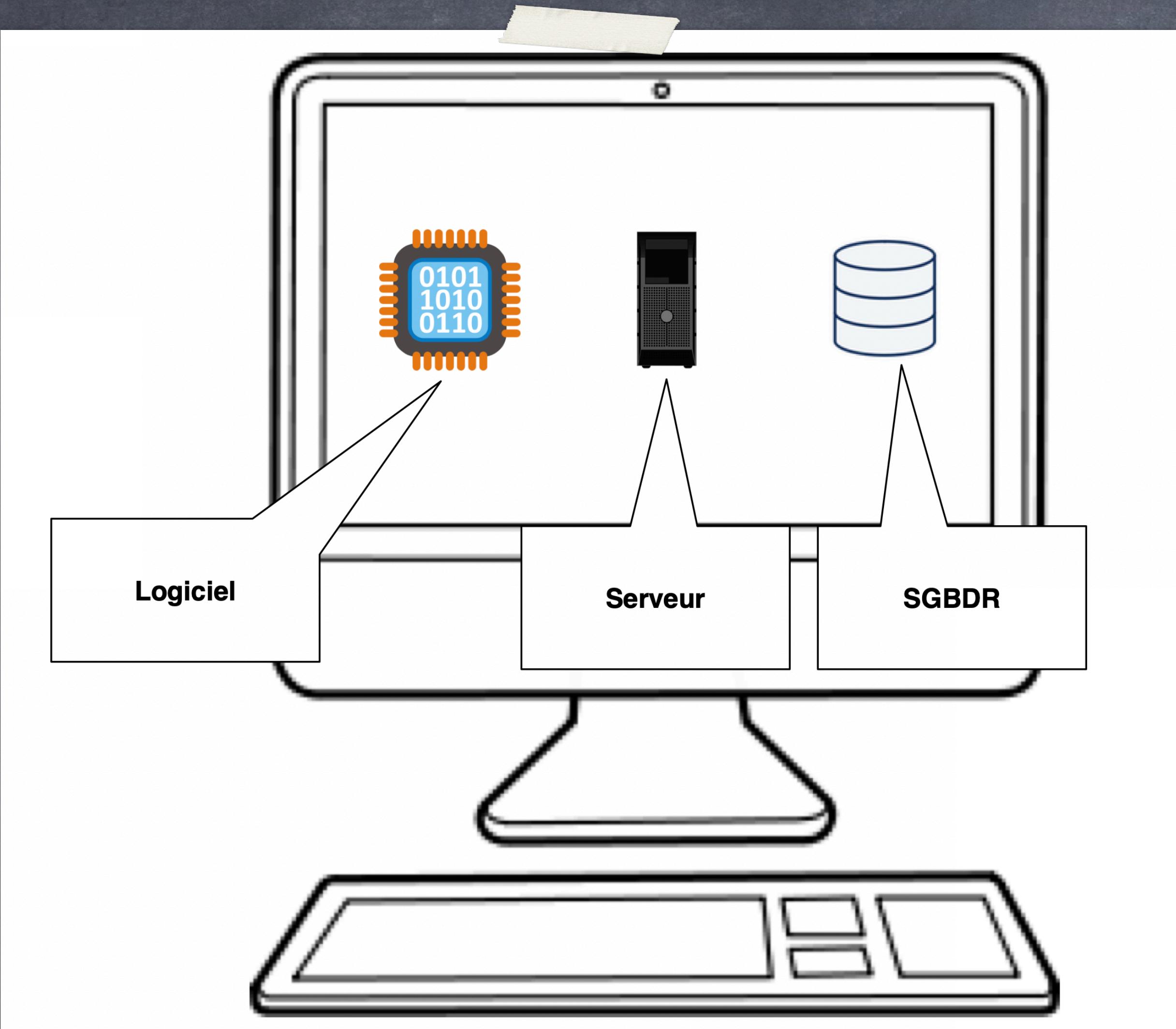
- ✓ Système de Gestion de Base de Données Relationnelles (RDBMS en anglais)
- ✓ Un logiciel qui permet le stockage et l'exploitation des données
- ✓ SQL est le langage de définition et de manipulation de données interprété par tous les SGBDR du marché
- ✓ Comme HTML, SQL est un standard, chaque éditeur de SGBDR l'adapte et ajoute ses propres fonctions. Le SGBDR le plus conforme au standard est PostGreSQL

Le modèle client lourd



Le modèle client léger





XAMPP est une solution intégrée qui installe un SGBDR (MariaDB), un serveur web (Apache) et un interpréteur PHP

Apache Friends

Télécharger Extensions Hébergement Communauté À propos Rechercher.. Rechercher FR

83 XAMPP Apache + MariaDB + PHP + Perl

Quid XAMPP ?

XAMPP est l'environnement de développement PHP le plus populaire

XAMPP est une distribution Apache entièrement gratuite et facile à installer contenant MySQL, PHP et Perl. Le paquetage open source XAMPP a été mis au point pour être incroyablement facile à installer et à utiliser.



Télécharger
Cliquez ici pour d'autres versions

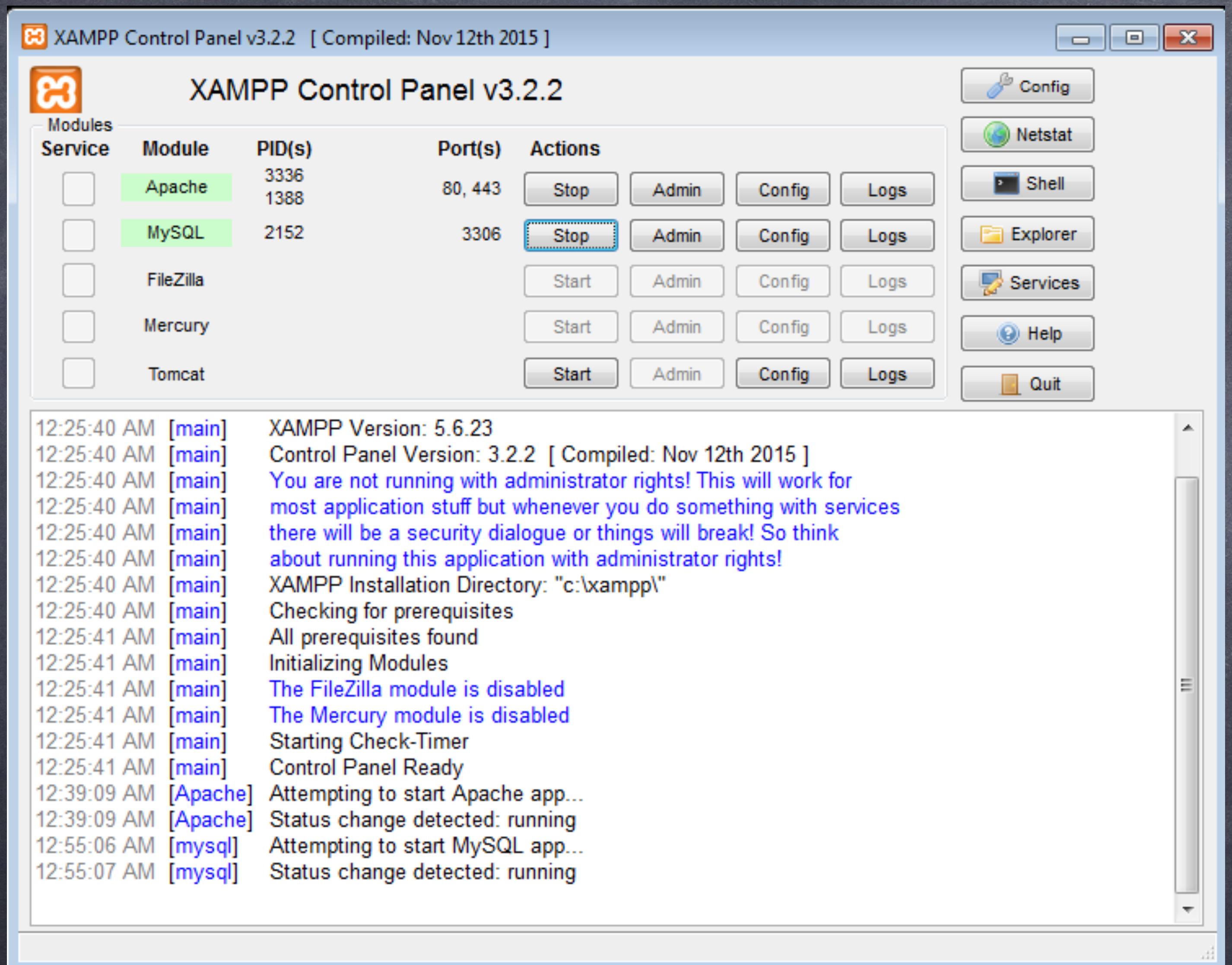
 XAMPP pour Windows
8.0.10 (PHP 8.0.10)

 XAMPP pour Linux
8.0.10 (PHP 8.0.10)

 XAMPP pour OS X
8.0.10 (PHP 8.0.10)

Aller sur le site

XAMPP Control panel



docker compose up

compose.yml

```
version: '3.8'

services:
  mysql:
    image: mysql:latest
    container_name: mysql_container
    environment:
      MYSQL_ROOT_PASSWORD: 123
      MYSQL_DATABASE: formation_sql
      MYSQL_USER: mysql_user
      MYSQL_PASSWORD: 456
    ports:
      - "3306:3306"
    volumes:
      - mysql_data:/var/lib/mysql
    networks:
      - custom-network

volumes:
  mysql_data:
    driver: local

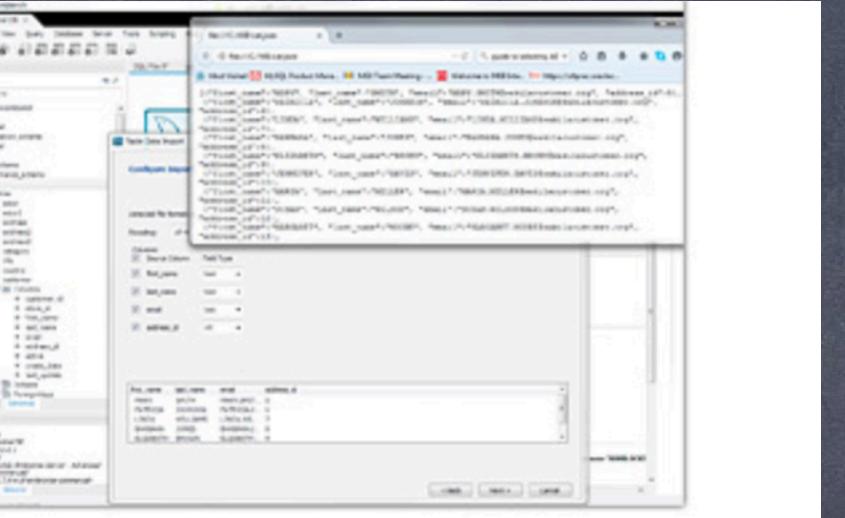
networks:
  custom-network:
    driver: bridge
```

- ✓ Console
(attention aux jeux de caractères avec certaines console Windows)
- ✓ phpMyAdmin avec XAMPP
- ✓ Un plugin pour votre IDE
- ✓ Un logiciel client (Valentina Studio, Mysql Workbench, Navicat ...)

MySQL Workbench

Enhanced Data Migration

[Download Now »](#)



The screenshot shows the MySQL Workbench interface with the 'Data Migration' feature open. The main window displays a list of tables and their columns from a source database. A configuration dialog box is overlaid, titled 'Configure migration', showing specific mapping details between source and target columns. The 'Test Type' dropdown is set to 'All'. Below the dialog, a preview table shows sample data being mapped.

The screenshot shows the VS Code interface with the Extensions view open. A search bar at the top of the Extensions view contains the text "mysql client". Below it, two extensions are listed under the "mysql client" category:

- MySQL** by Jun Han: 682K downloads, 4 stars. Status: Installer.
- MySQL** by cweijan: 165ms rating. Status: connect — sql.

A green callout bubble with the text "ajouter une connexion" (Add connection) points to the "connect — sql" button in the Connect To Server dialog. The dialog is titled "Connect To Server" and includes fields for Connection Name, Connection Scope (set to Global), Database Type (MySQL selected), Host (127.0.0.1), Username (root), Port (3306), Password, Databases (Connection database), Include Databases (Example: mysql,information_schema), RequestTimeout (10000), ConnectTimeout (5000), Timezone (+00:00), SSH Tunnel (disabled), and Use SSL (disabled). Buttons at the bottom are "Connect" and "Close".

Création de table

Jeu de caractère

Evite de récréer un base existante

Nom de la BD

```
CREATE DATABASE IF NOT EXISTS formation_sql
DEFAULT CHARACTER SET utf8;
USE formation_sql;
```

Activation de la BD toutes les nouvelles
requêtes seront exécutée sur cette base

nom de la colonne

nom de la table

```
CREATE TABLE IF NOT EXISTS personnes(  
    id INT UNSIGNED AUTO_INCREMENT  
    prenom VARCHAR(20),  
    nom VARCHAR(30) NOT NULL,  
    PRIMARY KEY (id)  
) ;
```

attributs de la
colonne

type de la colonne

- ✓ Les noms de table au pluriel
- ✓ Les noms de colonne au singulier
- ✓ Les noms (colonnes, tables ...) ne doivent comporter ni espaces, ni accents ni caractères spéciaux

La clef primaire d'une table est une colonne ou un groupe de colonnes qui identifient de façon unique une ligne (pas de doublons sur une clef primaire)

- ✓ Clef sémantique : valeurs signifiantes (numéro de sécurité sociale...)
- ✓ Clef technique : valeur numérique sans significations et souvent auto incrémentée

Type	Description
varchar(taille)	Nombre variable de caractères la taille peut aller de 1 à 255 octets
char(taille)	Nombre fixe de caractères, la taille peut aller de 1 à 255 octets
text	Nombre variable de caractères de 1 à 65 535 octets, si l'argument taille est omis alors la taille maximale sera utilisée
mediumtext	Nombre variable de caractères, 16 777 215 octets au maximum
longtext	Nombre variable de caractères, 4 294 967 295 octets au maximum

Note : Certains caractères sont codés sur plus d'un octet (caractères accentués, emoji ...)

Type	Description
tinyint	Entier codé sur 1 octet de 0 à 255 ou de -128 à +127
smallint	Entier codé sur 2 octets de 0 à 65 535 ou de -32 768 à +32 767
mediumint	Entier codé sur 3 octets de 0 à 16 777 215 ou de -8 388 608 à +8 388 607
int	Entier codé sur 4 octets de 0 à 4 294 967 295 ou de -2 147 483 648 à +2 147 483 648
bigint	Entier codé sur 8 octets de 0 à $2^{64} - 1$ ou de -2^{63} à $+2^{63} - 1$

Type	Description
float	Un nombre à virgule flottante codé sur 4 octets
double	Un nombre à virgule flottante codé sur 8 octets
decimal(taille, dec)	Un nombre à virgule fixe, l'argument taille indique le nombre total de chiffre et l'argument dec le nombre de chiffres après la virgule. decimal(4,2) permet donc de stocker des nombres de 0,00 à 99,99

Les types de données temporelles

Type	Description
date	Une date au format AAAA-MM-JJ de 1000-01-01 à 9999-12-31
datetime	Une date et une heure au format AAAA-MM-JJ hh:mm:ss ex : 2021-06-12 15:08:00
time	Une heure au format hh:mm:ss de -838:59:59 à 838:59:59
year	Une année de 1901 à 2155
timestamp	Une date sous forme de timestamp c'est à dire un entier représentant le nombre de secondes écoulées depuis le premier janvier 1970 à minuit. de 1970-01-01 00:00:01 à 2038-01-09 03:14:07

Créer une table adresses avec les colonnes suivantes :

- id (smallint unsigned)
- rue
- code_postal
- ville

Créer une table commandes avec les colonnes suivantes :

- id
- date_commande
- montant

CRUD

Create Read Update Delete

Activation de la BD

nom de la table

```
USE formation_sql;  
  
INSERT INTO personnes (prenom, nom)  
VALUES  
('Albert', 'Londres');
```

liste des colonnes

liste des valeurs
dans le même ordre que la liste des colonnes

```
USE formation_sql;
INSERT INTO personnes (prenom, nom)
VALUES
('Ansel', 'Adams'),
('Dorothea', 'Lange'),
('Vivian', 'Maier'),
('Diane', 'Arbus');
```

```
USE formation_sql;  
# Sélectionne toutes les lignes et toutes les colonnes  
SELECT * FROM personnes;
```

* pour toutes les colonnes, on peut également indiquer la liste des colonnes désirées en séparant chaque colonne d'une virgule
ex : SELECT nom, prenom FROM personnes

```
USE formation_sql;  
SELECT * FROM personnes WHERE id = 2
```

```
USE formation_sql;  
DELETE FROM personnes WHERE id=5;
```

Attention, c'est DELETE FROM et non
DELETE * FROM

Filtre, ne supprime que les lignes qui
répondent oui à la question

Filtre, ne modifie que les lignes qui répondent oui à la question

```
USE formation_sql;  
UPDATE personnes  
SET prenom='Aoifa'  
WHERE id=4;
```

Nouvelle valeur

Les critères de recherche

opérateur	description
=	égalité
<	infériorité
<=	infériorité ou égalité
>	supériorité
>=	supériorité ou égalité
!= ou <>	différence
between	interval ex : age between 5 and 8
in	alternatives ex : nom in ('jean', 'Pierre')
like	recherche avec joker ex : nom like 'a%' => tous les noms qui commencent par a le % remplace n'importe quelle suite de caractères

Dans la table personnes écrire les requêtes qui retournent les résultats suivants :

Les personnes dont l'id est entre 2 et 4
Toutes les personnes dont le prénom n'est pas Ansel
Toutes les personnes dont le nom commence par a
Toutes les personnes dont le nom contient a

```
DROP TABLE IF EXISTS `livres_simples`;
CREATE TABLE `livres_simples` (
  `id` mediumint unsigned NOT NULL AUTO_INCREMENT,
  `titre` varchar(80) NOT NULL,
  `prix` decimal(5,2) NOT NULL,
  `auteur` varchar(50) NOT NULL,
  `editeur` varchar(50) NOT NULL,
  `genre` varchar(50) NOT NULL,
  `date_publication` date DEFAULT NULL,
  `nationalite_auteur` varchar(50) DEFAULT NULL,
  `langue` varchar(50) DEFAULT NULL,
  `auteur_prenom` varchar(50) DEFAULT NULL,
  `auteur_nom` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`id`)
);
```

```
INSERT INTO `livres_simple` VALUES ('1', 'Le guide du routard intergalactique', '13.00', 'Douglas Adams', 'Grasset', 'Science fiction', '2000-02-08', 'anglais', 'français', 'Douglas', 'Adams'), ('2', 'Dune', '18.00', 'Frank Herbert', 'Hachette', 'Science fiction', '1995-03-09', 'américain', 'français', 'Frank', 'Herbert'), ('3', 'La nausée', '12.50', 'J.P. Sartre', 'PUF', 'Essai', '1988-12-11', 'français', 'français', 'J.P.', 'Sartre'), ('4', 'Désir et servitude', '9.00', 'F. Lordon', 'PUF', 'Essai', '2002-06-26', 'français', 'français', 'F.', 'Lordon'), ('5', 'Elévation', '11.00', 'David Brin', 'Grasset', 'Science fiction', '1998-12-17', 'américain', 'français', 'David', 'Brin'), ('6', 'Critique de la raison pure', '12.00', 'Emmanuel Kant', 'PUF', 'Philosophie', '2000-09-04', 'français', 'français', 'Emmanuel', 'Kant'), ('7', 'Propos sur le bonheur', '9.00', 'Alain', 'PUF', 'Philosophie', '1989-11-09', 'français', 'français', null, 'Alain'), ('8', 'Fondation', '14.00', 'Isaac Asimov', 'Robert Laffont', 'Science fiction', '1990-07-26', 'américain', 'français', 'Isaac', 'Asimov'), ('9', 'En terre étrangère', '10.00', 'Robert Heinlein', 'Robert Laffont', 'Science fiction', '1996-08-01', 'américain', 'français', 'Robert', 'Heinlein'), ('10', 'La République', '8.00', 'Platon', 'PUF', 'Philosophie', '1994-07-30', 'grecque', 'français', null, 'Platon'), ('11', 'Pensées', '11.00', 'Blaise Pascal', 'Hachette', 'Philosophie', '1996-06-16', 'français', 'français', 'Blaise', 'Pascal'), ('12', 'Discours de la méthode', '9.00', 'Blaise Pascal', 'Hachette', 'Philosophie', '2011-11-25', 'français', 'français', 'Blaise', 'Pascal'), ('13', 'Coder proprement', '18.00', 'Robert Martin', 'Dunod', 'Informatique', '1993-07-07', 'américain', 'français', 'Robert', 'Martin'), ('14', 'SQL Avancé', '45.00', 'Joe Celko', 'Dunod', 'Informatique', '2005-02-28', 'américain', 'français', 'Joe', 'Celko'), ('15', 'Programmer avec MySQL', '35.00', 'Christian Soutou', 'Eyrolles', 'Informatique', '1998-08-20', 'français', 'français', 'Christian', 'Soutou'), ('16', 'Crimes presque parfaits', '28.00', 'Patricia Highsmith', 'Hachette', 'Roman policier', '1991-01-08', 'américain', 'français', 'Patricia', 'Highsmith'), ('17', 'Carol', '22.00', 'Patricia Highsmith', 'Hachette', 'Roman policier', '2002-07-04', 'américain', 'français', 'Patricia', 'Highsmith'), ('18', 'Des chats et des hommes', '22.00', 'Patricia Highsmith', 'Hachette', 'Roman policier', '1992-11-03', 'américain', 'français', 'Patricia', 'Highsmith'), ('19', 'Sur les pas de Ripley', '22.00', 'Patricia Highsmith', 'Hachette', 'Roman policier', '2000-01-02', 'américain', 'français', 'Patricia', 'Highsmith'), ('20', 'L'inconnu du Nord Express', '22.00', 'Patricia Highsmith', 'Hachette', 'Roman policier', '2004-11-04', 'américain', 'français', 'Patricia', 'Highsmith'), ('21', 'Ripley et les ombres', '22.00', 'Patricia Highsmith', 'Hachette', 'Roman policier', '2007-07-26', 'américain', 'français', 'Patricia', 'Highsmith'), ('22', 'D'un retournement l'autre', '9.00', 'F. Lordon', 'La Fabrique', 'Théâtre', '2006-08-24', 'français', 'français', 'F.', 'Lordon'), ('23', 'Imperium', '15.00', 'F. Lordon', 'La Fabrique', 'Essai', '1993-11-19', 'français', 'français', 'F.', 'Lordon'), ('24', 'Et la vertu sauvera le monde', '6.00', 'F. Lordon', 'La Fabrique', 'Essai', '1992-10-16', 'français', 'français', 'F.', 'Lordon'), ('25', 'Economistes à gages', '7.50', 'F. Lordon', 'La Fabrique', 'Essai', '1995-08-23', 'français', 'français', 'F.', 'Lordon'), ('26', 'SQL for smarties', '55.86', 'Joe Celko', 'Morgan Kauffmann', 'Informatique', '2013-03-01', 'américain', 'anglais', 'Joe', 'Celko'), ('27', 'Data and Databases : Concepts in Practice', '67.47', 'Joe Celko', 'Morgan Kauffmann', 'Informatique', '2002-04-15', 'américain', 'anglais', 'Joe', 'Celko'), ('28', 'Thinking in Sets : Auxiliary, Temporal, and Virtual Tables in SQL', '28.43', 'Joe Celko', 'Morgan Kauffmann', 'Informatique', '2015-04-02', 'américain', 'anglais', 'Joe', 'Celko'), ('29', 'SQL Antipatterns', '33.44', 'Bill Karwin', 'O'Reilly', 'Informatique', '1992-10-12', 'américain', 'anglais', 'Bill', 'Karwin'), ('30', 'The pragmatic programmer : From journeyman to master', '33.85', 'Andrew Hunt', 'Addison Wesley', 'Informatique', '1991-06-15', 'américain', 'anglais', 'Andrew', 'Hunt'), ('31', 'Sei personaggi in cerca d'autore', '12.00', 'Luigi Pirandello', 'Grasset', 'Théâtre', '1921-06-09', 'italien', 'italien', 'Luigi', 'Pirandello'), ('32', 'La nuova colonia', '9.00', 'Luigi Pirandello', 'Grasset', 'Théâtre', '1926-03-24', 'italien', 'italien', 'Luigi', 'Pirandello'), ('33', 'Il nome della rosa', '10.00', 'Umberto Eco', 'Hachette', 'Roman historique', '1980-11-12', 'italien', 'italien', 'Umberto', 'Eco'), ('34', 'Clean code', '24.00', 'Robert Martin', 'Dunod', 'Informatique', '1989-11-04', 'Américain', 'anglais', 'Robert', 'Martin'), ('35', 'De bellum gallicum', '8.00', 'Julius Caesar', 'Hachette', 'Essai', '1990-06-12', 'romain', 'latin', 'Julius', 'Caesar'), ('36', 'El Ingenioso Hidalgo Don Quijote de la Mancha', '8.00', 'Miguel de Cervantes', 'Hachette', 'Roman', '1970-10-16', 'espagnol', 'castillan', 'Miguel', 'de Cervantes'), ('37', 'Los versos del capitán', '6.00', 'Pablo Neruda', 'Grasset', 'Poésie', '1984-11-25', 'chilien', 'castillan', 'Pablo', 'Neruda'), ('38', 'Elogio de la sombra', '8.00', 'Jorge Luis Borges', 'Grasset', 'Essai', '1978-02-05', 'argentin', 'castillan', 'Jorge Luis', 'Borges'), ('39', 'Pride and prejudice', '9.00', 'Jane Austen', 'Hachette', 'Roman', '1982-08-14', 'anglais', 'anglais', 'Jane', 'Austen');
```

```
SELECT * FROM livres_simples
WHERE
genre = 'science fiction'
AND
editeur = 'Grasset'
```

Tous les livres dont le genre est science fiction et
l'éditeur Grasset

Combinaison de critères

```
SELECT * FROM livres_simples
WHERE
genre = 'science fiction'
OR
éditeur = 'Grasset'
```

Tous les livres dont le genre est science fiction ou
l'éditeur Grasset

```
SELECT * FROM livres_simples
WHERE
genre = 'science fiction'
AND NOT
éditeur = 'Grasset'
```



Tous les livres dont le genre est science fiction **et** dont l'éditeur n'est pas Grasset

Le mot clef NOT inverse le critère

```
SELECT * FROM livres_simples  
WHERE  
auteur_prenom IS NULL
```

Tous les livres dont le prénom de l'auteur est inconnu

Null n'est pas une valeur, c'est une inconnue on ne peut donc comparer null à rien pas même à un autre null

Toute opération sur null ne peut donc que retourner null

Rechercher les livres d'informatique écrits en français

Rechercher les livres écrits en italien ou en castillan

Rechercher tous les livres qui ne sont pas en anglais
et dont le prix est inférieur à 12

Autres clauses

Tous les livres par ordre décroissant de prix

```
SELECT * FROM livres_simples  
ORDER BY prix DESC;
```

Tous les livres édités par Hachette
par ordre croissant de prix

```
SELECT * FROM livres_simples  
WHERE editeur = 'Hachette'  
ORDER BY prix;
```

Par défaut l'ordre de tri es croissant (ASC), mais il est possible de trier par ordre décroissant avec le mot clef DESC.

Pour un tri sur plusieurs colonnes on utilise un seul order by et on sépare chaque colonne par une virgule

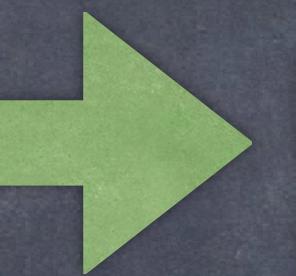
ex : ORDER BY prix DESC, éditeur

Les 5 livres les plus chers édités par Hachette

```
SELECT * FROM livres_simples
WHERE editeur = 'Hachette'
ORDER BY prix DESC
LIMIT 5;
```

Les 5 suivants

```
SELECT * FROM livres_simples
WHERE editeur = 'Hachette'
ORDER BY prix DESC
LIMIT 5 OFFSET 5;
```

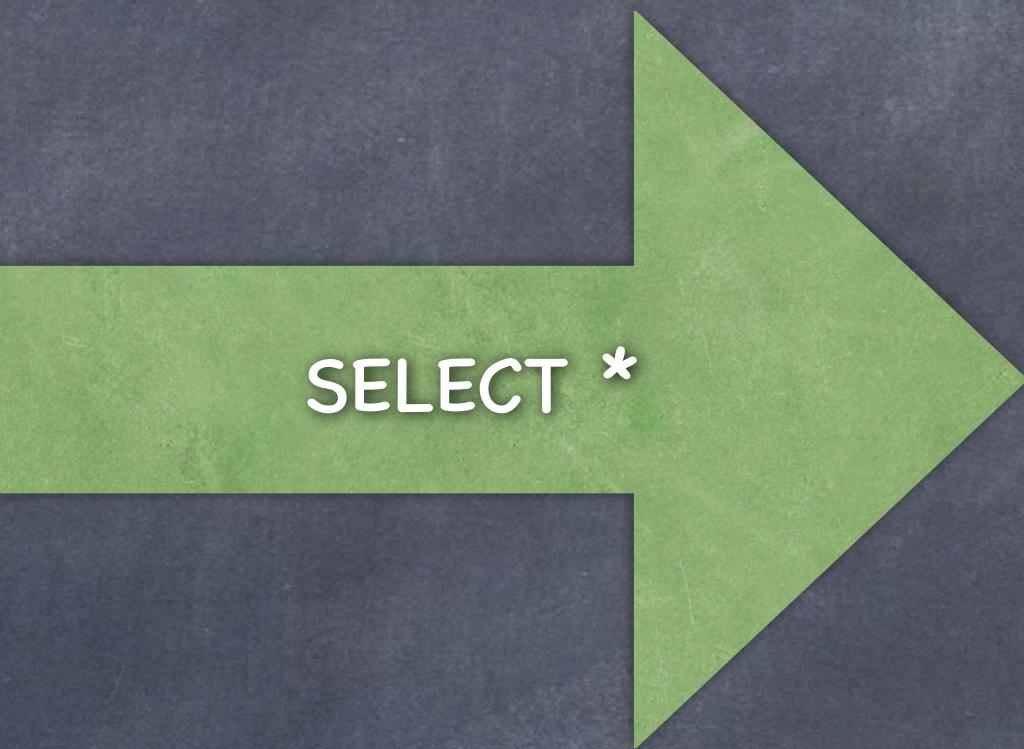


0	livre 1
1	livre 2
2	livre 3
3	livre 4
4	livre 5
5	livre 6
6	livre 7
7	livre 8
8	livre 9
9	livre 10

Les agrégats

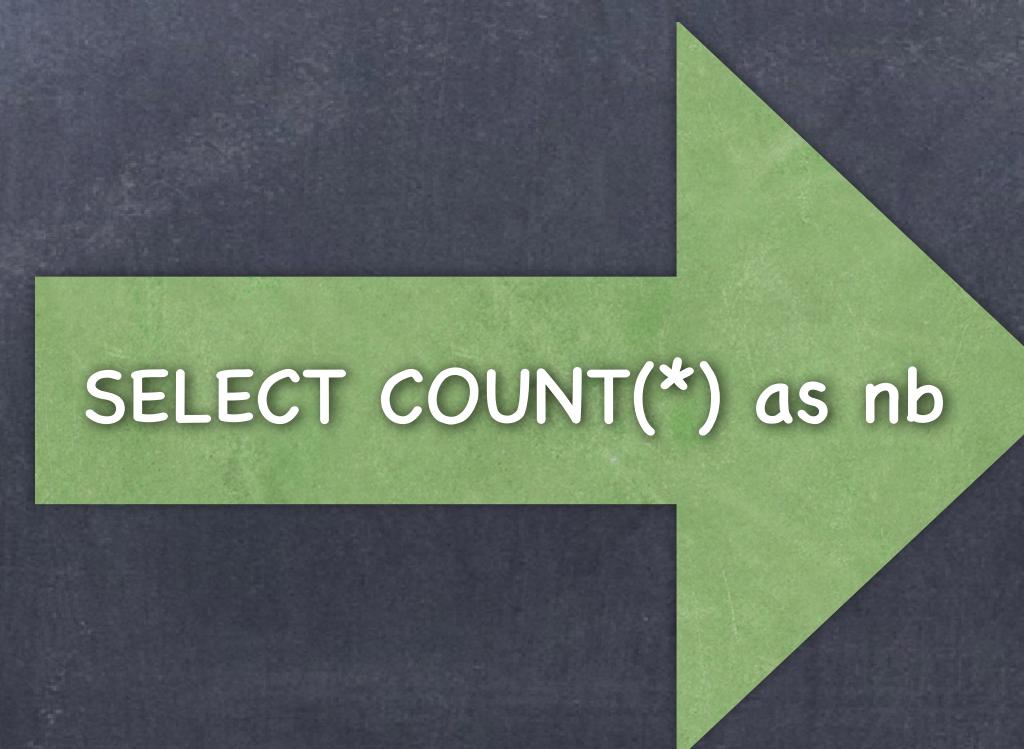
Une requête dont le résultat regroupe (agrège)
les données de plusieurs lignes

nom	age
Valérie	45
Paul	48
Céline	35
Sophie	38



nom	age
Valérie	45
Paul	48
Céline	35
Sophie	38

nom	age
Valérie	45
Paul	48
Céline	35
Sophie	38



nb
4

Fonction	Description
COUNT	compte le nombre de lignes
SUM	additionne les valeurs d'une colonne
MIN	donne la plus petite valeur d'une colonne
MAX	donne la plus grande valeur d'une colonne
AVG	moyenne des valeurs d'une colonne
STD	écart type des valeurs d'une colonne

Le nombre total de livres

```
SELECT COUNT(*) as nb FROM livres_simples
```

La somme des prix des livres de Grasset

```
SELECT SUM(prix) as total  
FROM livres_simples  
WHERE editeur= 'Grasset'
```

Afficher :

La moyenne des prix des livres d'informatique

Le livre le plus cher édite par Grasset

"Je veux le total des ventes par régions"

Les fonctions d'agrégations travaillent sur l'ensemble de la requête,
le regroupement permet de calculer des sous totaux

Le nombre de livres par éditeur

```
SELECT editeur, count(*) as nb  
FROM livres_simples  
GROUP BY editeur
```

La requête ne doit comporter que des fonctions d'agrégation et des colonnes présentent dans la clause group by

Liste des éditeurs sans doublons

```
SELECT editeur  
FROM livres_simples  
GROUP BY editeur
```

Autre possibilité

```
SELECT DISTINCT editeur  
FROM livres_simples
```

1. Le SGBDR liste les colonnes de la table
2. Le SGBDR collecte toutes les données
3. Le SGBDR applique les filtres de la clause WHERE
4. Le SGBDR calcule les agrégats sur les lignes filtrées

Comment filtrer sur un agrégat ?

Liste des éditeurs qui ont édité
plus de trois auteurs différents

```
SELECT
editeur,
count(DISTINCT auteur) as nb
FROM livres_simples
GROUP BY editeur
HAVING nb > 3
```

Afficher :

Les auteurs qui ont écrit dans plus de deux genres

Les éditeurs qui ont publié des livres
dans plus d'une langue

Travailler
avec plusieurs tables

Dans notre base de données nous avons une table personnes et une table adresses

Comment lier une adresse avec une personne ?

Il faut partager une information entre les deux tables
Une colonne dans une des deux tables reprend une information de l'autre table
Cette colonne est appelée **clef étrangère**

id	nom	prénom	id_adresse
1	Hugo	Victor	1
2	Kant	Emmanuel	2
3	Freud	Sigmund	2

id	rue	code_postal	ville
1	3 rue Orfila	75020	Paris
2	5 Grande rue	25440	Quingey

Le type de la clef étrangère doit être le même que
celui de la colonne référencée dans l'autre table

```
ALTER TABLE personnes
ADD COLUMN id_adresse SMALLINT UNSIGNED;
```

liste des tables

SELECT

personnes.id, prenom, nom,

rue, code_postal, ville

FROM personnes, addresses

WHERE id_adresse = addresses.id

les colonnes de la table personne
on préfixe id avec le nom de la
table car il existe aussi une
colonne id dans addresses

règles de correspondance

```
SELECT
```

```
p.id, p.prenom, p.nom,
```

```
a.rue, a.code_postal, a.ville
```

```
FROM personnes as p, adresses as a
```

```
WHERE id_adresse = a.id
```

Les règles de correspondances sont dans une clause ON,
cela laisse la clause WHERE pour les filtres et sépare
donc les deux opérations

```
SELECT
    p.id, p.nom,
    a.rue, a.code_postal, a.ville
FROM personnes as p
JOIN adresses as a
ON p.id_adresse = a.id
WHERE a.ville = 'Paris'
```

Les valeurs nulles sont filtrées, elles n'apparaissent pas dans le résultat de la requête

Ici on fait une correspondance entre une clef primaire et une clef étrangère, la clef primaire est unique par définition donc :

Une personne possède une seule adresse

Un adresse peut concerner plusieurs personnes

id	nom	prénom	id_adresse
1	Hugo	Victor	1
2	Kant	Emmanuel	2
3	Freud	Sigmund	2
4	Atreides	Paul	null

id	rue	code_postal	ville
1	3 rue Orfila	75020	Paris
2	5 Grande rue	25440	Quingey

LEFT JOIN retourne toutes les lignes de la table de gauche et les colonnes en correspondance sur la table de droite

```
SELECT
    p.id, p.nom,
    a.rue, a.code_postal, a.ville
FROM
    personnes as p LEFT JOIN adresses as a
ON p.id_adresse = a.id
WHERE a.ville = 'Paris'
```

table de gauche

table de droite

La colonne id_adresse sur la dernière ligne fait référence à une valeur qui n'existe pas dans la table adresses

c'est mal !!

id	nom	prénom	id_adresse
1	Hugo	Victor	1
2	Kant	Emmanuel	2
3	Freud	Sigmund	2
4	Atreides	Paul	3



id	rue	code_postal	ville
1	3 rue Orfila	75020	Paris
2	5 Grande rue	25440	Quingey

Placer une contrainte sur la table
qui contient la clef étrangère

nom de la clef étrangère
dans la table

```
ALTER TABLE personnes
ADD CONSTRAINT
personnes_to_adresses
FOREIGN KEY (id_adresse)
REFERENCES adresses(id)
```

nom de la contrainte
ce nom doit être unique
sur toute la BD

nom de la table et de la colonne
référencée par la clef étrangère

S'il y a une contrainte, le SGBDR nous empêche de violer cette contrainte donc impossible de supprimer une adresse si elle est référencée par une personne

Suppression en cascade des personnes quand on supprime l'adresse liée

```
ALTER TABLE personnes
ADD CONSTRAINT
personnes_to_adresses
FOREIGN KEY (id_adresse)
REFERENCES adresses(id)

ON DELETE CASCADE

ON UPDATE CASCADE;
```

Mise à jour en cascade de la valeur de la clef étrangère quand la colonne liée est modifiée

```
ALTER TABLE personnes
ADD CONSTRAINT
personnes_to_adresses
FOREIGN KEY (id_adresse)
REFERENCES adresses(id)

ON DELETE SET NULL
```

La clef étrangère devient null
quand l'adresse liée est supprimée

Définir une relation entre les tables
personnes et commandes

Ecrire la requête qui permet d'obtenir les commandes
et les personnes avec leur adresse

Les fonctions

- ✓ Un fonction est un programme qui retourne une valeur,
- ✓ Cette valeur peut être utilisée comme une colonne
- ✓ Une fonction peut admettre des arguments qui seront utilisés pour calculer la valeur de retour

la fonction

```
SELECT  
DATE_FORMAT(  
    date_publication,  
    '%d/%m/%Y'  
) as date_pub  
FROM livres_simples
```

la date

le format

%d	le jour du mois
%j	le jour de l'année
%m	le numéro du mois
%M	le nom du mois
%Y	l'année sur 4 chiffres
%y	l'année sur 2 chiffres
%u	le numéro de la semaine dans l'année commençant le lundi
%U	le numéro de la semaine dans l'année commençant le dimanche

%W	le nom du jour de la semaine
%w	le numéro du jour de la semaine commençant le dimanche
%H	l'heure sur 24 heures
%h	l'heure sur 12 heures
%i	les minutes
%s	les secondes
%p	mention pm ou am sur l'heure

Extraction d'une partie d'une date

day	le jour du mois
dayofyear	le jour de l'année
day name	le nom du jour de la semaine
month	le numéro du mois
monthname	le nom du mois
year	l'année sur 4 chiffres

```
SELECT  
CONCAT(prenom, ' ', nom) as nom_complet  
FROM livres_simple
```

Problèmes avec les colonnes NULL

```
SELECT  
CONCAT_WS(' ', prenom, nom) as nom_complet  
FROM livres_simple
```

Plus de problèmes avec les colonnes NULL