

Introduction générale aux bases de données relationnelles

*Attribution - Partage dans les Mêmes Conditions :
<http://creativecommons.org/licenses/by-sa/3.0/fr/>*

Table des matières

I - Contexte	3
II - Base de données	4
III - Exercice	6
IV - Système de gestion de bases de données (SGBD)	7
V - Exercice	10
VI - Introduction à DB Fiddle	11
VII - Exercice	15
VIII - Application de base de données	16
IX - Exercice	19
X - Donnée (en relationnel) : table, objet, propriété, domaine, atomicité	20
XI - Exercice	23
XII - Langage de données : l'exemple du langage SQL	24
XIII - Exercice	26
XIV - Essentiel	27
XV - Quiz	28
Solutions des exercices	33
Abréviations	41
Index	42
Crédits des ressources	43
Contenus annexes	44

I Contexte

Durée : 2h

Environnement de travail : DB Fiddle

Pré-requis : Aucun

Les *BD* ^{p.41} sont nées à la fin des années 1960 pour combler les lacunes des systèmes de fichiers et faciliter la gestion qualitative et quantitative des données informatiques. Les *SGBD* ^{p.41} sont des applications informatiques permettant de créer et de gérer des BD (comme Oracle ou PostgreSQL par exemple).

Les BD relationnelles, issues de la recherche de Codd chez IBM, sont celles qui ont connu le plus grand essor depuis les années 1970, et qui reste encore aujourd'hui les plus utilisées. On utilise des *SGBDR* ^{p.41} pour les implémenter. Le langage *SQL* ^{p.41} est le langage commun à tous les SGBDR, ce qui permet de concevoir des BD relativement indépendamment des systèmes utilisés.

Les usages de BD se sont aujourd'hui généralisés pour entrer dans tous les secteurs de l'entreprise, depuis les petites bases utilisées par quelques personnes dans un service pour des besoins de gestion de données locales, jusqu'aux bases qui gèrent de façon centralisée des données partagées par tous les acteurs de l'entreprise.

L'accroissement de l'utilisation du numérique comme outil de manipulation de toutes données (bureautique, informatique applicative, etc.) et comme outil d'extension des moyens de communication (réseaux), ainsi que les évolutions technologiques (puissance des PC, Internet, etc.) ont rendu indispensable, mais aussi complexifié la problématique des BD.

Les conséquences de cette généralisation et de cette diversification des usages se retrouvent dans l'émergence de solutions conceptuelles et technologiques nouvelles, notamment les bases de données du mouvement NoSQL, particulièrement utilisées par les grands acteurs du web.

Plan du cours

Ce module introduit les concepts fondamentaux des bases de données relationnelles. Qu'est-ce qu'une base de données ? un système de gestion de bases de données ? une donnée relationnelle ? une application de base de données ?

Il permet également de découvrir l'*IDE* ^{p.41} web DB Fiddle¹ permettant d'écrire et exécuter du code SQL afin de créer des bases de données relationnelles.

1. <https://www.db-fiddle.com/>

II Base de données

Objectifs

- Connaître le concept de données ;
- Connaître le concept de bases de données.

Mise en situation

La base de l'informatique, c'est de travailler sur des données. Tout est donnée : des caractères que vous entrez au clavier aux photos que vous prenez avec votre téléphone, en passant par les couleurs des pixels de votre écran.

Les données sont donc aussi la base des applications web : un message, un contact, un profil, une commande, etc. Chacun de ces exemples a un point commun : une structure pré-définie. Un contact aura un nom et un prénom, une commande aura un prix et une adresse d'expédition, un message aura une date d'envoi et un auteur, etc. Comment les stocker efficacement ?

C'est précisément dans ce but que l'on utilise des bases de données : elles facilitent grandement le stockage d'un ensemble de données qui répondent à une structure logique.

Logiciel et données

Un logiciel informatique est composé de programmes, c'est à dire :

- d'instructions,
- et de données auxquelles s'appliquent ces instructions.

👁 Exemple

Par exemple un logiciel de traitement de texte est composé de fonctions :

- ouvrir, copier, coller, insérer une image, changer la police, enregistrer, etc.,
- et de fichiers sur lesquels elles s'appliquent. Dans ce cas les fichiers de traitement de texte sont les données.

Définition lâche de base de données : un ensemble de données

Az Définition

On appelle parfois base de données tout ensemble de données stocké numériquement et pouvant servir à un ou plusieurs programmes. De ce point de vue des fichiers sur un disque dur, un fichier de tableur, voire un fichier de traitement de texte peuvent constituer des bases de données.

Définition restreinte de base de données : un ensemble de données structuré

Az Définition

On appellera base de données un ensemble de données numériques qui possède une structure ; c'est à dire dont l'organisation répond à une **logique** systématique.

On parlera de modèle logique de données pour décrire cette structure.

Base de données relationnelle

👁 Exemple

Une base de données relationnelle permet d'organiser les données en tableaux (appelés relations).

espèce	eucaryote	multicellulaire	propriété
bactéries	false	false	
archées	false	false	
protistes	true	false	
champignons	true	true	décompose
végétaux	true	true	photosynthétise
animaux	true	true	ingère

Base de données de classification classique des espèces animales

Fonctions d'une base de données

💡 Fondamental

Une base de données est structurée afin de pouvoir mieux répondre à des fonctions fondamentales en informatique, telles que :

- Stocker l'information de façon fiable (c'est à dire être capable de restituer l'information entrée dans le système).
- Traiter de grands volumes de données (massification).
- Traiter rapidement les données (optimisation).
- Sécuriser les accès aux données (gérer les autorisations selon les utilisateurs).
- Contrôler la qualité des données (par exemple la cohérence par rapport à un modèle pré-établi).
- Partager les données (entre plusieurs applications dédiées à plusieurs métiers).
- Rendre accessible les données en réseau (gérer la concurrence des accès parallèles).
- Etc.

❓ Exercice

[solution n°1 p. 33]

Parmi les raisons suivantes, lesquelles justifient l'utilisation d'une base de données ?

A S'abstraire des modalités de stockage physique de l'information.

B Minimiser la redondance d'information.

C Faciliter la mise à jour de données.

D Contrôler la cohérence des données.

E Sécuriser l'accès aux données.

F Mieux gérer l'accès concurrent aux données.

IV Système de gestion de bases de données (SGBD)

Objectifs

- Connaître le concept de *SGBD* ^{p.41};
- Connaître les SGBD les plus utilisés ;
- Connaître la différence entre SGBD relationnel et SGBD non-relationnel.

Mise en situation

Lorsque l'on développe une application, on a souvent besoin de stocker des données, qui ont, la plupart du temps, des liens entre elles. Sur un réseau social, il y a des personnes qui peuvent interagir entre elles, s'envoyer des messages, se suivre, etc. Lorsque ces données deviennent très grandes, de l'ordre de plusieurs millions, il est inimaginable de les stocker dans un simple fichier, ou un tableur, qui ferait office de base de données.

Les systèmes de gestion de base de données, ou SGBD, sont des outils qui aident à manipuler des bases de données. Leur fonction est double : permettre de décrire la structure des données a priori, et gérer la cohérence des données et des liens entre elles.

Système de Gestion de Bases de Données

Az Définition

Un *SGBD* ^{p.41} est un logiciel qui prend en charge la structuration, le stockage, la mise à jour et la maintenance d'une base de données. Il est l'unique interface entre les informaticiens et les données (définition des schémas, programmation des applications), ainsi qu'entre les utilisateurs et les données (consultation et mise à jour).

Exemples de SGBD

Exemple

- Oracle est un SGBD commercial **relationnel** et relationnel-objet situé parmi les leaders du marché dans le domaine des applications métiers.
- PostgreSQL est un SGBD libre **relationnel** puissant qui offre une alternative libre (licence BSD) aux solutions commerciales d'Oracle ou IBM. PostgreSQL dispose d'une documentation d'excellente qualité et d'un support très proche du standard SQL, ce qui en fait également un bon outil d'apprentissage.
- Access est un SGBD **relationnel** commercial, qui offre une interface graphique permettant de concevoir rapidement des applications de petite envergure ou de réaliser des prototypes.
- MongoDB est un SGBD **non-relationnel** libre (licence Apache) orienté document. Il permet de gérer facilement de très grandes quantités de données réparties sur de nombreux ordinateurs. Il se base sur le format JSON.
- MySQL est un SGBD **relationnel** open source (licence GPL et commerciale) très utilisé pour la réalisation de sites web dynamiques. Depuis la version 4 MySQL implémente la plupart des fonctions attendues d'un SGBD relationnel. MySQL a son pendant libre, MariaDB.

- SQLite est un SGBD **non-relationnel** libre. C'est un système très léger, utilisé pour des applications embarquées. Il est entièrement écrit en C et comporte la majorité des fonctionnalités attendues de SQL.

Objectifs des SGBD

💡 Fondamental

- **Indépendance physique des données**

Le changement des modalités de stockage de l'information (optimisation, réorganisation, segmentation, etc.) n'implique pas de changements des programmes.

- **Indépendance logique des données**

L'évolution de la structure d'une partie des données n'influe pas sur l'ensemble des données.

- **Manipulation des données par des non-informaticiens**

L'utilisateur n'a pas à savoir comment l'information est stockée et calculée par la machine, mais juste à pouvoir la rechercher et la mettre à jour à travers des IHM ou des langages *assertionnels* simples.

- **Administration facilitée des données**

Le SGBD fournit un ensemble d'outils (dictionnaire de données, audit, tuning, statistiques, etc.) pour améliorer les performances et optimiser les stockages.

- **Optimisation de l'accès aux données**

Les temps de réponse et de débits globaux sont optimisés en fonction des questions posées à la BD.

- **Contrôle de cohérence (intégrité sémantique) des données**

Le SGBD doit assurer à tout instant que les données respectent les règles d'intégrité qui leur sont imposées.

- **Partageabilité des données**

Les données sont simultanément consultables et modifiables.

- **Sécurité des données**

La confidentialité des données est assurée par des systèmes d'authentification, de droits d'accès, de chiffrement des mots de passe, etc.

- **Sûreté des données**

La persistance des données, même en cas de panne, est assurée, grâce typiquement à des sauvegardes et des journaux qui gardent une trace persistante des opérations effectuées.

⊕ Complément

Pourquoi des SGBD ? (cf. p.44)

SGBD relationnel et non-relationnel

⊕ Complément

Jusqu'au début des années 2000, la plupart des bases de données étaient relationnelles ; mais avec l'arrivée des géants du web (Google, Amazon ou Facebook), entreprises qui gèrent des quantités énormes de données, s'est développé un mouvement important de développement de bases de données non-relationnelles, également appelées NoSQL pour *Not-Only SQL*.

🔍 Exercice

[solution n°2 p. 33]

Enquête Wikipédia : à partir de la page Wikipédia sur les SGBD, classer les SGBDR en fonction de leur licence : fr.wikipedia.org/wiki/Système_de_gestion_de_base_de_données².

A SQLite

B DB2

C MySQL

D Oracle Database

E MariaDB

F Microsoft Access

G PostgreSQL

H Microsoft SQL Server

Logiciel libre	Logiciel Freemium	Logiciel propriétaire

² https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_de_base_de_donn%C3%A9es

VI Introduction à DB Fiddle

Objectifs

- Savoir utiliser l'interpréteur SQL en ligne DB Fiddle.

Mise en situation

Il existe une grande diversité de systèmes de gestion de base de données : chacun est bien adapté à une situation donnée : de gros volumes, comme pour un moteur de recherche ? Besoin d'une cohérence parfaite, comme pour des transactions bancaires ?

Avant de faire un choix, il est utile de tester les différents SGBD, prendre en main leur syntaxe et choisir celui qui répond à vos besoins.

Mais installer tous les SGBD sur votre machine peut-être long et pénible. Dans ce module, vous apprendrez à utiliser DBFiddle, une application web qui permet de tester plusieurs SGBD sans aucune installation préalable.

Interpréteur SQL

Az Définition

Un interpréteur SQL est un programme qui traite des commandes fournies par l'utilisateur pour interagir avec une base de données. On parle parfois de *REPL* pour *Return Eval Print Loop*.

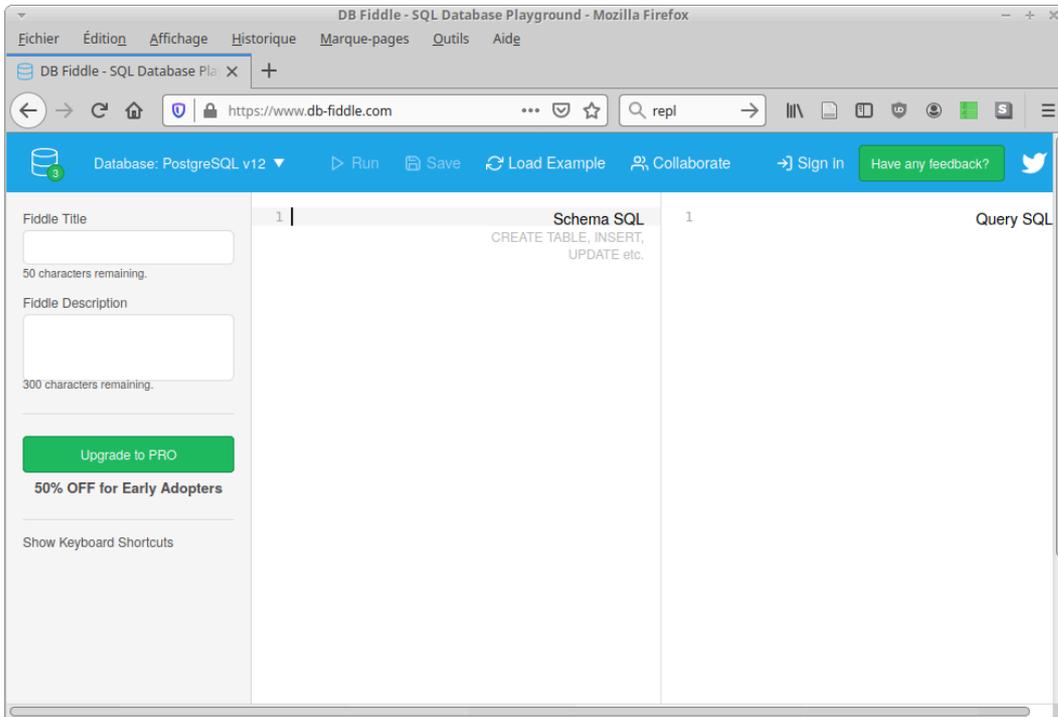
DB Fiddle

Az Définition

DB Fiddle est un interpréteur SQL en ligne, il est disponible sur le Web et il permet de créer et exécuter du code SQL sans installer de SGBD.

[db-fiddle.com](https://www.db-fiddle.com/)³

³ <https://www.db-fiddle.com/>

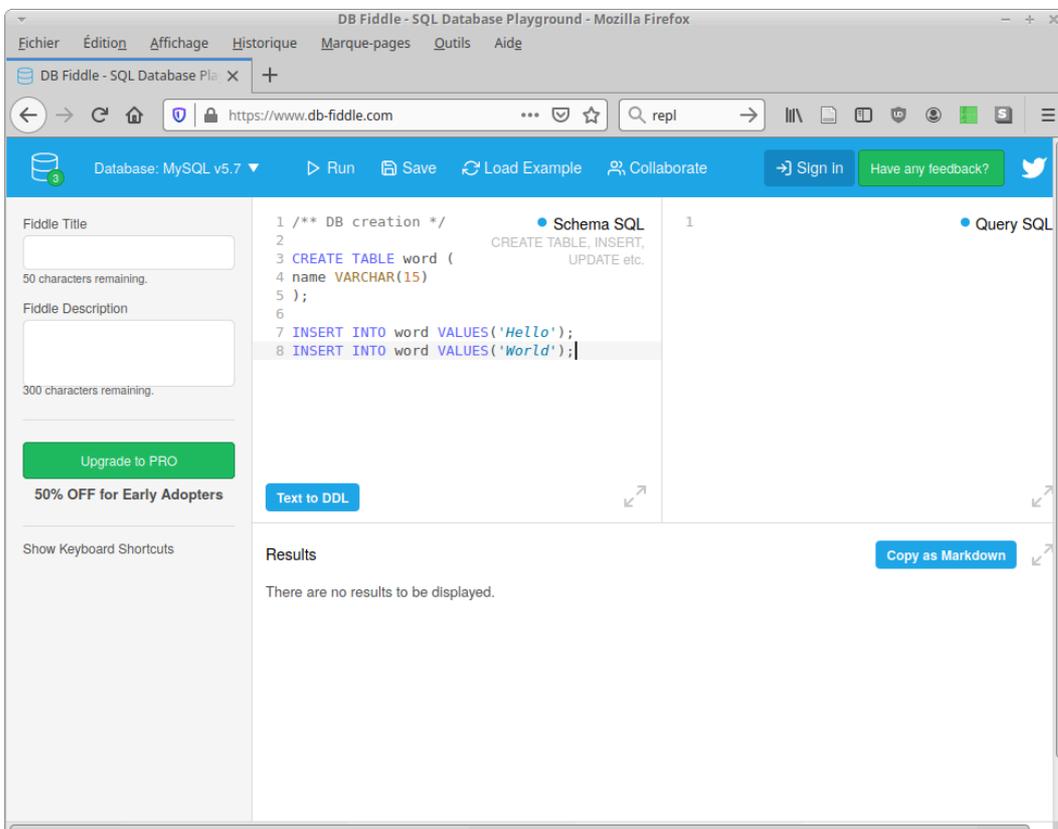


Interface d'accueil DB Fiddle

Créer une BD

Méthode

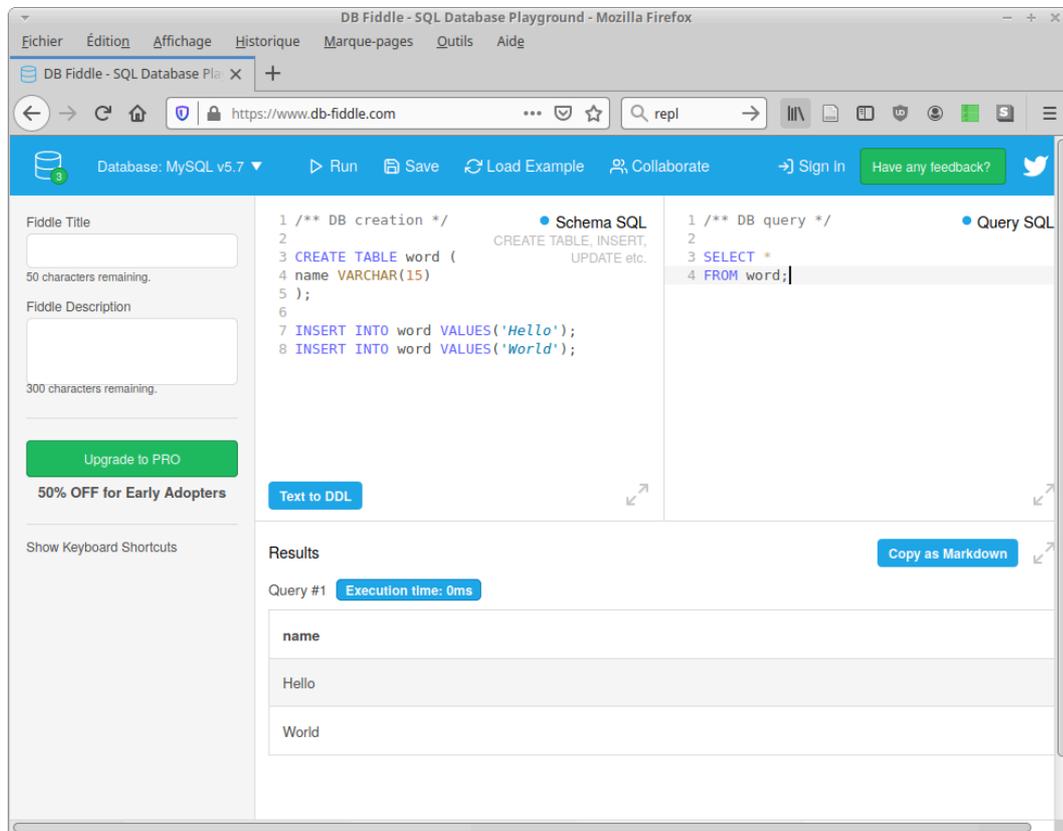
Pour créer une base de données (commande SQL CREATE TABLE) ou gérer les données (commande SQL INSERT, UPDATE, DELETE) il faut insérer le code SQL correspondant dans la partie gauche puis cliquer sur [Run](#) (ou CTRL+RETURN au clavier).



Interroger une BD

[Méthode](#)

Pour poser des question à la base de données (commande SQL SELECT) il faut ajouter le code SQL correspondant dans la partie droite puis cliquer sur [Run](#) (ou CTRL+RETURN au clavier).



[Exemple](#)

Il est possible de tester DB Fiddle avec ce code de type *Hello World*.

```

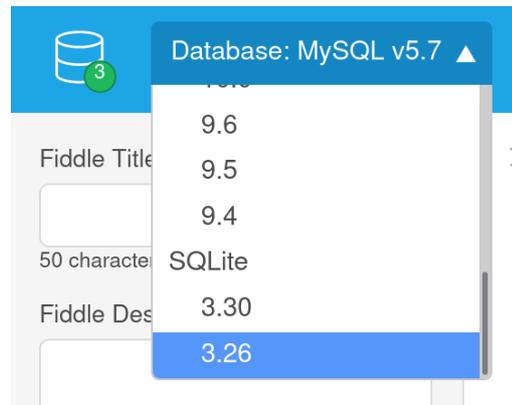
1 /** DB creation */
2
3 CREATE TABLE word (
4 name VARCHAR(15)
5 );
6
7 INSERT INTO word VALUES('Hello');
8 INSERT INTO word VALUES('World');

1 /** DB query */
2
3 SELECT *
4 FROM word;
```

Choisir un SGBD

[Méthode](#)

En haut à gauche un menu déroulant permet de choisir le type de SGBD à utiliser. On travaillera par défaut avec la dernière version de PostgreSQL ou de MySQL.



Menu déroulant du clic droit sur interpréteur

DB Disco

[+ Complément](#)

DB Disco est une alternative à DB Fiddle limitée à PostgreSQL.

pic.crzt.fr/dbdisco

② Exercice

[solution n°3 p. 34]

Copier ces commandes dans un interpréteur SQL.

```
1 CREATE TABLE liste(  
2 nombre INT  
3 );  
4  
5 INSERT INTO liste VALUES (2);  
6 INSERT INTO liste VALUES (7);  
7 INSERT INTO liste VALUES (3);  
1 SELECT MAX(nombre) FROM liste;
```

Quel est le résultat de l'exécution de ce code ?

VIII Application de base de données

Objectifs

- Connaître la différence entre une application et une base de données ;
- Savoir citer quelques exemples d'application de bases de données.

Mise en situation

Avez-vous déjà utilisé une base de données ? Lorsque vous effectuez des achats en ligne, avez-vous l'impression de manipuler directement une base de données ?

Probablement pas, car une base de données seule n'est pas directement utilisable par un utilisateur humain : elle n'est utilisable que par les informaticiens qui connaissent son langage de programmation et par les applications qui ont été programmées pour s'en servir.

Une base de donnée va donc souvent de pair avec une application, qui est l'interface entre les utilisateurs et les données elles-mêmes.

Application de base de données

Az Définition

On appelle application de base de données un logiciel informatique permettant à un utilisateur final de manipuler (lire ou écrire) les données d'une base de données.

Application web

👁 Exemple

Une application web est composée d'interfaces en HTML qui permettent d'écrire et de lire des données dans une base de données, via un langage applicatif, comme par exemple PHP.

Application web Mastodon

👁 Exemple



Mastodon, une plateforme de micro-blogging



Un toot de
@dolfsquare@tooting.intensifi.es

L'application *Mastodon* est composée d'interfaces web permettant d'entrer des données (saisir son profil, *tooter* un post, *booster* un post, etc. et de sortir des données (consulter son fil *Mastodon*, faire une recherche sur un *hashtag*, sur un utilisateur) d'une base de données.

Ces informations (profil des utilisateurs, *toots*, *hashtags*, etc.) sont stockées dans des bases de données des serveurs fédérés de *Mastodon*.

Application de bureau Access

👁 Exemple

Avec un logiciel comme Access on peut réaliser à la fois une base de données et une application permettant de manipuler cette base de données pour des besoins bureautiques simples.

bookid	bookname	year	status	borrower	returned	fname	lname
1	Information technology	1985	checked out	1234	N	Jessica	Benner
2	Information technology	1985	checked out	1236	Y	Daqing	He
2	Information technology	1985	checked out	1237	Y	Ronald	Larsen

Aperçu de Microsoft Access

Une application en programmation

👁 Exemple

Dans le cas de programmes informatiques, on peut stocker des données dans une base pour effectuer des traitements plus efficacement. Par exemple, si on dispose d'un tableau de nombres, on peut faire la somme des éléments de colonnes bien plus facilement, rapidement et sûrement qu'en écrivant soit même un programme.

Compagnie aérienne

 Exemple

Une base de données de gestion de l'activité d'une compagnie aérienne concerne les voyageurs, les vols, les avions, le personnel, les réservations, etc.

Une application à partir d'une telle base de données pourra permettre la gestion des réservations, des disponibilités des avions en fonction des vols à effectuer, des affectations des personnels volants, etc.

🔍 Exercice

[solution n°4 p. 34]

Parmi les options suivantes, lesquelles correspondent à des applications de bases de données ?

A Twitter

B SQLite

C Le logiciel d'une caisse enregistreuse de supermarché

D Une application de SMS sur téléphone

X Donnée (en relationnel) : table, objet, propriété, domaine, atomicité

Objectifs

- Connaître le concept de relation (ou table) ;
- Savoir représenter des données simples en relationnel ;
- Connaître le concept de donnée atomique.

Mise en situation

Prenez une application de vente de livres en ligne : quelles sont les données qu'il faut gérer ? En vrac, on peut penser aux livres, aux auteurs, aux clients, aux éditeurs, etc.

Si vous deviez réaliser une base de données simplifiée pour stocker ces différentes entités, il est probable que vous dessiniez des tableaux : un tableau pour les auteurs, un tableau pour les clients, etc.

Et ça tombe bien, car ce sont exactement ces tableaux que l'on appelle relations, et qui sont à la base du modèle relationnel.

Base de données relationnelle

 Rappel

Une base de données relationnelle permet d'organiser les données en tables (appelées relations).

Chaque case de la table contient une information atomique.

Ligne (objet)

 Az Définition

Chaque ligne de la table correspond à un objet que l'on veut gérer dans la base de données : une voiture, une personne, une espèce, etc.

On parle aussi d'**enregistrement**.

 Fondamental

Toutes les lignes d'une même table correspondent à des objets du même type, donc dans une table, on met soit des voitures, soit des personnes, mais on ne mélange pas les deux.

Colonne (propriété)

 Az Définition

Chaque colonne de la table correspond à une propriété des objets qui se trouvent dans la table ; tous les objets de la table partagent donc les mêmes propriétés.

On parle aussi d'**attribut**.

Domaine

Az Définition

Chaque colonne de la table est associée à un domaine de valeur fixé **a priori**, par exemple : entier, texte, booléen, etc.

On parle aussi de **type** de données.

Cellule (donnée en relationnel)

Az Définition

Une donnée en relationnel, c'est le contenu d'une cellule d'une table, qui correspond à la propriété d'un objet.

propriété 1 domaine : d1	propriété 2 domaine : d2	...
objet1, donnée 1	objet1, donnée 2	...
objet2, donnée 1	objet2, donnée 2	...
...

Une table ou relation (en relationnel)

Exemple

espèce domaine : texte	eucaryote domaine : booléen	...
bactéries	false	...
archées	false	...
...

Exemple de relation instanciée

Atomicité (contre-exemple)

⚠ Attention

Pour que la base de données fonctionne correctement on veille à ne mettre qu'une seule donnée par cellule, c'est le **principe d'atomicité** en relationnel.

espèce, domaine : texte
bactéries : procaryotes unicellulaires
archées : procaryotes unicellulaires
protistes : eucaryotes unicellulaires

espèce, domaine : texte
champignons : eucaryotes multicellulaires qui décomposent
végétaux : eucaryotes multicellulaires qui photosynthétisent
animaux : eucaryotes multicellulaires qui ingèrent

Un mauvais exemple de relation : les données ne sont pas atomiques (il y a plusieurs données par case de la table)

② Exercice

[solution n°5 p. 35]

Ordonner les mots pour compléter la table suivante :

Horodatage	Numéro Train	Prix payé (en €)	Classe
31-12-2019 14:35	67394	142.90	1
12-12-2019 14:35	68173	(3)	1
01-04-2019 9:30	(2)	53.90	2
(1)	67399	34.90	2
04-12-2019 13:50	68137	3.30	(4)

Billets de train

A 2

B 62940

C 128.90

D 03-08-2019 10:20

Réponse : ____ _

XII Langage de données : l'exemple du langage SQL

Objectifs

- Connaître la notion de langage orienté donnée ;
- Savoir tester une instruction en SQL.

Mise en situation

Imaginez : vous développez une bibliothèque musicale, et vous avez identifié la structure des données que vous souhaitez gérer : les artistes d'un côté, les albums de l'autre, les playlists des utilisateurs, etc.

Au moment de passer à la pratique, vous vous demandez alors : comment expliquer cette structure au système de gestion de base de données ? Comment insérer de nouvelles données ? Comment récupérer les données existantes ?

La réponse tient en trois mots : grâce à SQL. Ce langage est le couteau suisse des bases de données, et permet aux développeurs de communiquer avec les SGBD.

Langage de données

Az Définition

Un langage de données est un langage informatique permettant de décrire et de manipuler les schémas et les données d'une *BD* ^{p.41}.

Synonyme

Remarque

On parle aussi de langage orienté données.

SQL

Fondamental

SQL ^{p.41} est le langage consacré aux SGBD relationnels et relationnels-objet.

Il permet de :

- créer des tables, en définissant le domaine de chaque colonne ;
- insérer des lignes dans les tables ;
- lire les données entrées dans la base de données.

Création de table en SQL (définition du schéma de données)

[Exemple](#)

```
1 CREATE TABLE student (  
2 number INTEGER PRIMARY KEY,  
3 name TEXT,  
4 city TEXT  
5 );
```

Cette instruction permet de créer une relation `student` comportant les propriétés `number`, `name` et `city` de domaines, respectivement, entier, texte et texte (`number` est la clé primaire de la table, il servira à identifier les enregistrements).

Insertion de ligne en SQL (création de données)

[Exemple](#)

```
1 INSERT INTO student (number, name, city)  
2 VALUES (1, 'Holmes', 'Londres');
```

Cette instruction permet de créer l'étudiant numéro 1, de nom Holmes qui habite la ville de Londres.

Manipulation de données en SQL (exploitation des données)

[Exemple](#)

```
1 SELECT name  
2 FROM student  
3 WHERE city = 'Londres';
```

Cette instruction permet de rechercher les noms de tous les étudiants habitant la ville de Londres.

Autres langages de données

[Complément](#)

- XQuery est un langage de données mobilisé dans les bases de données arborescentes XML.
- Les bases NoSQL proposent des langages de données spécifiques, souvent inspirés du SQL. Par exemple le langage de MongoDB permet de manipuler une base de contenus JSON.

🔍 Exercice

[solution n°6 p. 35]

Exécuter l'instruction textuelle SQL ci-dessous qui, permet de créer une table dans une base de données.

```
1 CREATE TABLE composition (  
2 aliment TEXT,  
3 calories FLOAT,  
4 joules FLOAT,  
5 glucides FLOAT,  
6 protides FLOAT,  
7 lipides FLOAT  
8 );  
9  
10 INSERT INTO composition(aliment, calories, joules, glucides, protides, lipides)  
11 VALUES ('Abricot', 277, 1158, 63.4, 4.6, 0.4);  
12  
1 SELECT aliment  
2 FROM composition;
```

Quelle valeur de retour est obtenue ?

XIV Essentiel

Toute application a besoin de stocker des données. Mais une donnée, c'est assez abstrait, et si chaque application invente son propre système de stockage, la perte de temps et le risque d'erreurs sont immenses.

On utilise donc des bases de données. Lorsque les données sont structurées *a priori*, comme c'est le cas pour des personnes ou des musiques, on parle de base de données relationnelles.

Les systèmes de gestion de bases de données sont des outils qui permettent aux informaticiens de manipuler des bases de données. Comment ? Grâce au langage SQL, qui permet de communiquer avec quasiment tous les SGBD, en ayant besoin d'apprendre une seule syntaxe.

Pour tester les différents SGBD sans avoir à les installer sur votre ordinateur, vous pouvez utiliser l'application web DBFiddle.

XV Quiz

Exercice 1 : Synonymes

[solution n°7 p. 36]

Exercice

Quels sont les synonymes du mot **relation** au sens d'une base de données relationnelle ?

A table

B tableau

C lien

D association

E ami

Exercice

Quels sont les synonymes du mot **ligne** au sens d'une base de données relationnelle ?

A propriété

B objet

C enregistrement

D table

Exercice

Quels sont les synonymes du mot **colonne** au sens d'une base de données relationnelle ?

A valeur

B type

C propriété

D attribut

Exercice

Quels sont les synonymes du mot **donnée** au sens d'une base de données relationnelle ?

A colonne

B cellule

C case

D ligne

Exercice

Quels sont les synonymes du mot **domaine** au sens d'une base de données relationnelle ?

A propriété

B type

C données

D ensemble de valeurs

Exercice 7

[solution n°8 p. 37]

Classer les SGBD suivants selon qu'ils sont essentiellement relationnel ou non-relationnel.

S'aider de Wikipédia le cas échéant.

A Redis

B Neo4j

C Microsoft Access

D Elasticsearch

E 4D

F Cassandra

G Microsoft SQL Server

H MariaDB

I MySQL

J MongoDB

K OrientDB

L PostgreSQL

M DB2

SGBD relationnel	SGBD non-relationnel

Exercice 8

Utiliser un interpréteur SQL en ligne comme DB Fiddle pour donner le résultat de l'exécution des codes SQL ci-après.

Exercice

```

1 CREATE TABLE adresse (
2 pk_id INTEGER PRIMARY KEY,
3 nom VARCHAR(30),
4 prenom VARCHAR(30),
5 code_postal INTEGER,
6 ville VARCHAR(30)
7 );
8
9
10 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (1,
    'Boulgakov', 'Mikhaïl', 60200, 'Compiègne');
11 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (2, 'Tolstoï',
    'Alexis', 60420, 'Dompierre');
12 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (4, 'Gogol',
    'Nikolaï', 60420, 'Mery-la-Bataille');
13 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (5,
    'Pouchkine', 'Alexandre', 60680, 'Canly');

1 SELECT ville
2 FROM adresse
3 WHERE nom='Tolstoï';

```

Exercice

```

1 CREATE TABLE adresse (
2 pk_id INTEGER PRIMARY KEY,
3 nom VARCHAR(30),
4 prenom VARCHAR(30),
5 code_postal INTEGER,
6 ville VARCHAR(30)
7 );
8
9
10 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (1,
    'Boulgakov', 'Mikhaïl', 60200, 'Compiègne');
11 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (2, 'Tolstoï',
    'Alexis', 60420, 'Dompierre');
12 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (4, 'Gogol',
    'Nikolaï', 60420, 'Mery-la-Bataille');
13 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (5,
    'Pouchkine', 'Alexandre', 60680, 'Canly');

1 SELECT MAX(code_postal)
2 FROM adresse;

```

Exercice

```

1 CREATE TABLE voiture(
2 pk_immatriculation CHAR(7) PRIMARY KEY,
3 modele VARCHAR(30),
4 marque VARCHAR(30),
5 couleur VARCHAR(30)
6 );
7
8

```

```

9 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('AA123AA', 'Clio', 'Renault', 'Noir');
10 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('AB123NB', '807', 'Peugeot', 'Bleu');
11 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('DE001TR', 'Clio', 'Renault', 'Rouge');
12 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('AM007JB', '205', 'Peugeot', 'Rose');
13 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('BK2000B', 'Cayenne', 'Porsche', 'Noir');
14 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('ZX987FR', 'Twingo', 'Renault', 'Jaune');
15
1 SELECT couleur
2 FROM voiture
3 WHERE modele = '205';

```

Exercice

```

1 CREATE TABLE voiture(
2 pk_immatriculation CHAR(7) PRIMARY KEY,
3 modele VARCHAR(30),
4 marque VARCHAR(30),
5 couleur VARCHAR(30)
6 );
7
8
9 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('AA123AA', 'Clio', 'Renault', 'Noir');
10 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('AB123NB', '807', 'Peugeot', 'Bleu');
11 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('DE001TR', 'Clio', 'Renault', 'Rouge');
12 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('AM007JB', '205', 'Peugeot', 'Rose');
13 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('BK2000B', 'Cayenne', 'Porsche', 'Noir');
14 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES
  ('ZX987FR', 'Twingo', 'Renault', 'Jaune');
15
1 SELECT LEFT(pk_immatriculation, 2)
2 FROM voiture
3 WHERE couleur='Noir' AND marque='Renault'

```

Exercice 13

[solution n°10 p. 39]

Indiquer quels attributs sont atomiques dans la table question ci-dessous.

```

1 CREATE TABLE question (
2 label VARCHAR(255) PRIMARY KEY,
3 answer VARCHAR(255) NOT NULL,
4 questionnaire VARCHAR(255) NOT NULL
5 );
6
7 INSERT INTO question VALUES (
8 '1. Qui était roi de France de 742 à 814 ?',
9 'Charlemagne (Carolingiens)',
10 'Histoire'
11 );
12
13 INSERT INTO question VALUES (
14 '2. Qui était roi de France de 940 à 996 ?',

```

Quiz

```
15 'Hugues Capet (Capétiens)',  
16 'Histoire'  
17 );  
18  
19 INSERT INTO question VALUES (  
20 '3. Qui était roi de France de 1462 à 1515 ?',  
21 'Louis XII (Valois)',  
22 'Histoire'  
23 );
```

A label

B answer

C questionnaire

Attribut atomique	Attribut non atomique

Solutions des exercices

Solution n°1

[exercice p. 6]

Parmi les raisons suivantes, lesquelles justifient l'utilisation d'une base de données ?

A S'abstraire des modalités de stockage physique de l'information.

B Minimiser la redondance d'information.

C Faciliter la mise à jour de données.

D Contrôler la cohérence des données.

E Sécuriser l'accès aux données.

F Mieux gérer l'accès concurrent aux données.

Solution n°2

[exercice p. 10]

Enquête Wikipédia : à partir de la page Wikipédia sur les SGBD, classer les SGBDR en fonction de leur licence : fr.wikipedia.org/wiki/Système_de_gestion_de_base_de_données⁴.

Logiciel libre	Logiciel Freemium	Logiciel propriétaire
PostgreSQL	MySQL	Oracle Database
MariaDB		Microsoft Access
SQLite		Microsoft SQL Server
		DB2



- PostgreSQL, SQLite et MariaDB sont des logiciels libres : leur code est disponible ouvertement à l'inspection à l'exécution, et est modifiable.
- Oracle est une solution propriétaire gérée par l'entreprise éponyme. DB2 est la solution d'IBM. Microsoft propose un SGBDR classique (SQL Server) et une solution plus modeste pour des usages bureautique (Access).
- MySQL est une solution *freemium* d'Oracle : c'est un logiciel ouvert et libre mais il existe des licences propriétaires pour des usages commerciaux.

4. https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_gestion_de_base_de_donn%C3%A9es

Solution n°3

[exercice p. 15]

Copier ces commandes dans un interpréteur SQL.

```

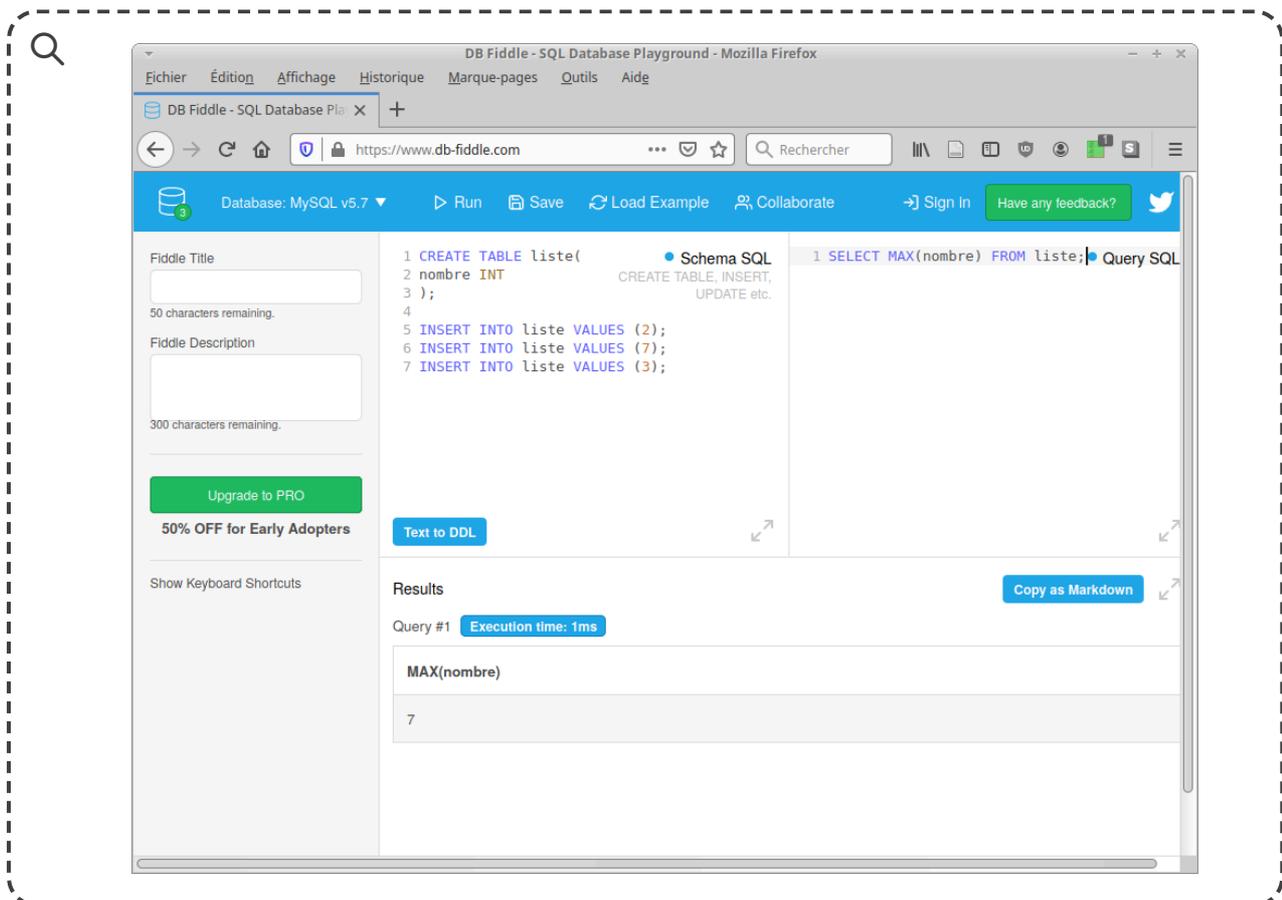
1 CREATE TABLE liste(
2 nombre INT
3 );
4
5 INSERT INTO liste VALUES (2);
6 INSERT INTO liste VALUES (7);
7 INSERT INTO liste VALUES (3);

1 SELECT MAX(nombre) FROM liste;

```

Quel est le résultat de l'exécution de ce code ?

7



Solution n°4

[exercice p. 19]

Parmi les options suivantes, lesquelles correspondent à des applications de bases de données ?

 A

Twitter Twitter, une alternative à Mastodon, est une application web de bases de données.

 B

SQLite SQLite n'est pas une application de bases de données, c'est un système de gestion de bases de données (SGBD).

C

Le logiciel d'une caisse enregistreuse de supermarché peut être vu comme une application de bases de données : les informations de tous les produits sont stockées à l'achat pour comptabilité.

D

Une application de SMS sur téléphone peut être vue comme utilisant une base de données de SMS. Le plus souvent, cette base de données est stockée directement sur le téléphone dans certains cas elle peut être stockée sur des serveurs distants.

Solution n°5

[exercice p. 23]

Ordonner les mots pour compléter la table suivante :

Horodatage	Numéro Train	Prix payé (en €)	Classe
31-12-2019 14:35	67394	142.90	1
12-12-2019 14:35	68173	(3)	1
01-04-2019 9:30	(2)	53.90	2
(1)	67399	34.90	2
04-12-2019 13:50	68137	3.30	(4)

Billets de train

A 03-08-2019 10:20**B** 62940**C** 128.90**D** 2

Solution n°6

[exercice p. 26]

Exécuter l'instruction textuelle SQL ci-dessous qui, permet de créer une table dans une base de données.

```

1 CREATE TABLE composition (
2 aliment TEXT,
3 calories FLOAT,
4 joules FLOAT,
5 glucides FLOAT,
6 protides FLOAT,
7 lipides FLOAT
8 );
9
10 INSERT INTO composition(aliment, calories, joules, glucides, protides, lipides)
11 VALUES ('Abricot', 277, 1158, 63.4, 4.6, 0.4);
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Quelle valeur de retour est obtenue ?

Solution n°7

[exercice p. 28]

Exercice

Quels sont les synonymes du mot **relation** au sens d'une base de données relationnelle ?

A table

B tableau

C lien

D association

E ami

Exercice

Quels sont les synonymes du mot **ligne** au sens d'une base de données relationnelle ?

A propriété

B objet

C enregistrement

D table

Exercice

Quels sont les synonymes du mot **colonne** au sens d'une base de données relationnelle ?

A valeur

B type

C propriété

D attribut

Exercice

Quels sont les synonymes du mot **donnée** au sens d'une base de données relationnelle ?

A colonne

B cellule

C case

D ligne

Exercice

Quels sont les synonymes du mot **domaine** au sens d'une base de données relationnelle ?

A propriété

B type

C données

D ensemble de valeurs

Solution n°8

[exercice p. 29]

Classer les SGBD suivants selon qu'ils sont essentiellement relationnel ou non-relationnel.

S'aider de Wikipédia le cas échéant.

SGBD relationnel	SGBD non-relationnel
PostgreSQL	Neo4j
Microsoft Access	MongoDb
MySQL	Cassandra
4D	OrientDB
MariaDB	Redis
Microsoft SQL Server	Elasticsearch
DB2	

Solution n°9

Exercice

```

1 CREATE TABLE adresse (
2 pk_id INTEGER PRIMARY KEY,
3 nom VARCHAR(30),
4 prenom VARCHAR(30),
5 code_postal INTEGER,
6 ville VARCHAR(30)
7 );
8
9
10 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (1, 'Boulgakov',
    'Mikhaïl', 60200, 'Compiègne');
11 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (2, 'Tolstoï',
    'Alexis', 60420, 'Dompierre');
12 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (4, 'Gogol',
    'Nikolaï', 60420, 'Mery-la-Bataille');
13 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (5, 'Pouchkine',
    'Alexandre', 60680, 'Canly');
1 SELECT ville
2 FROM adresse
3 WHERE nom='Tolstoï';

```

Dompierre

Exercice

```

1 CREATE TABLE adresse (
2 pk_id INTEGER PRIMARY KEY,
3 nom VARCHAR(30),
4 prenom VARCHAR(30),
5 code_postal INTEGER,
6 ville VARCHAR(30)
7 );
8
9
10 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (1, 'Boulgakov',
    'Mikhaïl', 60200, 'Compiègne');
11 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (2, 'Tolstoï',
    'Alexis', 60420, 'Dompierre');
12 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (4, 'Gogol',
    'Nikolaï', 60420, 'Mery-la-Bataille');
13 INSERT INTO adresse (pk_id, nom, prenom, code_postal, ville) VALUES (5, 'Pouchkine',
    'Alexandre', 60680, 'Canly');
1 SELECT MAX(code_postal)
2 FROM adresse;

```

60680

Exercice

```

1 CREATE TABLE voiture(
2 pk_immatriculation CHAR(7) PRIMARY KEY,
3 modele VARCHAR(30),
4 marque VARCHAR(30),
5 couleur VARCHAR(30)
6 );
7
8
9 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('AA123AA',
    'Clio', 'Renault', 'Noir');
10 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('AB123NB',
    '807', 'Peugeot', 'Bleu');

```

```

11 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('DE001TR',
    'Clio', 'Renault', 'Rouge');
12 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('AM007JB',
    '205', 'Peugeot', 'Rose');
13 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('BK2000B',
    'Cayenne', 'Porsche', 'Noir');
14 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('ZX987FR',
    'Twingo', 'Renault', 'Jaune');
15
1 SELECT couleur
2 FROM voiture
3 WHERE modele = '205';

```

Rose

Exercice

```

1 CREATE TABLE voiture(
2 pk_immatriculation CHAR(7) PRIMARY KEY,
3 modele VARCHAR(30),
4 marque VARCHAR(30),
5 couleur VARCHAR(30)
6 );
7
8
9 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('AA123AA',
    'Clio', 'Renault', 'Noir');
10 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('AB123NB',
    '807', 'Peugeot', 'Bleu');
11 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('DE001TR',
    'Clio', 'Renault', 'Rouge');
12 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('AM007JB',
    '205', 'Peugeot', 'Rose');
13 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('BK2000B',
    'Cayenne', 'Porsche', 'Noir');
14 INSERT INTO voiture (pk_immatriculation, modele, marque, couleur) VALUES ('ZX987FR',
    'Twingo', 'Renault', 'Jaune');
15
1 SELECT LEFT(pk_immatriculation, 2)
2 FROM voiture
3 WHERE couleur='Noir' AND marque='Renault'

```

AA

Solution n°10

[exercice p. 31]

Indiquer quels attributs sont atomiques dans la table question ci-dessous.

```

1 CREATE TABLE question (
2 label VARCHAR(255) PRIMARY KEY,
3 answer VARCHAR(255) NOT NULL,
4 questionnaire VARCHAR(255) NOT NULL
5 );
6
7 INSERT INTO question VALUES (
8 '1. Qui était roi de France de 742 à 814 ?',
9 'Charlemagne (Carolingiens)',
10 'Histoire'
11 );
12
13 INSERT INTO question VALUES (
14 '2. Qui était roi de France de 940 à 996 ?',
15 'Hugues Capet (Capétiens)',
16 'Histoire'
17 );

```

```
18
19 INSERT INTO question VALUES (
20 '3. Qui était roi de France de 1462 à 1515 ?',
21 'Louis XII (Valois)',
22 'Histoire'
23 );
```

Attribut atomique	Attribut non atomique
questionnaire	label answer



- L'attribut `label` possède deux informations : le numéro de question et la question en elle même.
- L'attribut `answer` possède deux informations : le nom du roi et sa dynastie.

Abréviations

BD : Base de Données

IDE : Integrated Development Environment (Environnement de Développement Intégré)

SGBD : Système de Gestion de Bases de Données

SGBDR : Système de Gestion de Bases de Données Relationnelles

SQL : Structured Query Language

Index

Application	16
BD.....	4, 11
Langage.....	24
LDD	24
LMD.....	24
Redondance	44
Relationnel.....	7
SGBD.....	7, 44
SQL	24

Crédits des ressources

Menu déroulant du clic droit sur interpréteur p. 14
Universel - Transfert dans le Domaine Public

Mastodon, une plateforme de micro-blogging p. 16
Licence : Domaine Public

Un toot de @dolfsquare@tooting.intensifi.es p. 17
Attribution - Partage dans les Mêmes Conditions

Aperçu de Microsoft Access p. 17
Attribution - Partage dans les Mêmes Conditions

Contenus annexes

1. Pourquoi des SGBD ?

Jadis...

Avant l'avènement des SGBD, chaque application informatique dans l'entreprise impliquait sa propre équipe de développement, ses propres supports physiques, ses propres fichiers, ses propres normes, ses propres langages, etc.

Conséquences...

L'existence conjointe et croissante de ces applications indépendantes a des effets négatifs, tels que :

- La multiplication des tâches de saisie, de développement et de support informatique ;
- La redondance anarchique des informations dans les fichiers ;
- L'incohérence des versions simultanées de fichiers ;
- La non-portabilité des traitements en raison des différences dans les formats et langages ;
- La multiplication des coûts de développement et de maintenance des applications.

Problèmes...

Les conséquences précédemment citées se répercutent sur l'entreprise en générant des problèmes humains et matériels ;

- Coûts en personnels qualifiés et en formations ;
- Remise des pouvoirs de décision entre les mains de spécialistes informatiques ;
- Tout changement matériel ou logiciel a un impact sur les applications ;
- Tout changement de la structure des données nécessite de modifier les programmes.

Or...

En réalité les applications ne sont jamais totalement disjointes, des données similaires (le cœur de l'information d'entreprise) sont toujours à la base des traitements.

On peut citer typiquement :

- Les données comptables
- Les données clients et fournisseurs
- Les données relatives à la gestion des stocks
- Les données relatives aux livraisons
- Les données marketing et commerciales
- Les données relatives au personnel
- ...

