

# Bases de Données : Introduction

Stéphane Devismes

Université Grenoble Alpes

10 septembre 2020

# Plan

1 Organisation

2 Introduction

# Plan

1 Organisation

2 Introduction

# Intervenants

- **Cours** (Stephane.Devismes@univ-grenoble-alpes.fr)  
<http://www-verimag.imag.fr/~devismes/WWW/enseignements.html>
  - **Le mercredi, sauf exceptions** (12 séances, 1h30)
  - *Support : Slides (à trous) disponibles sur ma page web*
  - **COVID** : petites videos
- **TDs et TPs** : deux groupes (11 séances, 1h30)
  - Jean-Noël Bouvier  
(Jean-Noel.Bouvier@univ-grenoble-alpes.fr)
  - **le vendredi matin, sauf les trois premières semaines (mardi, 9h45-11h15) 9h45-11h15, Gr 2**
  - **le vendredi matin, sauf les trois premières semaines (mardi, 8h-9h30) 11h30-13h, Gr 1**
  - *Support : Poly de TD/TP disponible sur ma page web*

# Amménagement COVID

**COURS : A partir de la semaine prochaine : distanciel !**

- 1 cours = une à plusieurs petites vidéos
- Cloud UGA
  - <https://cloud.univ-grenoble-alpes.fr/index.php/s/wipgK9sfHZW5HEs>

## **Où envoyer le lien ?**

- <https://trombi.univ-grenoble-alpes.fr/>
- [etu-2019-polytech-kai3ri160@univ-grenoble-alpes.fr](mailto:etu-2019-polytech-kai3ri160@univ-grenoble-alpes.fr)

# Amménagement COVID

## COURS : **A partir de la semaine prochaine : distanciel !**

- 1 cours = une à plusieurs petites vidéos
- Cloud UGA
  - <https://cloud.univ-grenoble-alpes.fr/index.php/s/wipgK9sfHZW5HEs>

### **Où envoyer le lien ?**

- <https://trombi.univ-grenoble-alpes.fr/>
- [etu-2019-polytech-kai3ri160@univ-grenoble-alpes.fr](mailto:etu-2019-polytech-kai3ri160@univ-grenoble-alpes.fr)
- Au jour et à l'horaire du cours : discussion, point sur les TD/TPs, questions de cours
  - **Préparer vos questions à l'avance !**
  - Discord : <https://discord.gg/rSZpTnY>

# Amménagement COVID

## COURS : **A partir de la semaine prochaine : distanciel !**

- 1 cours = une à plusieurs petites vidéos
- Cloud UGA
  - <https://cloud.univ-grenoble-alpes.fr/index.php/s/wipgK9sfHZW5HEs>

### **Où envoyer le lien ?**

- <https://trombi.univ-grenoble-alpes.fr/>
- [etu-2019-polytech-kai3ri160@univ-grenoble-alpes.fr](mailto:etu-2019-polytech-kai3ri160@univ-grenoble-alpes.fr)
- Au jour et à l'horaire du cours : discussion, point sur les TD/TPs, questions de cours
  - **Préparer vos questions à l'avance !**
  - Discord : <https://discord.gg/rSZpTnY>

## TD/TP :

- Etudiants en binôme
- Présence (être masqué) une séance sur deux, en alternance
- **Collaboratif** : faire le point chaque semaine avec son binôme

# Evaluation

- Une note d'examen (1h30, en fin de semestre).
- Une note de contrôle continu : 2 DMs (en binôme) organisés par le chargé de TD/TP. Note : moyenne des deux DMs.

# Evaluation

- Une note d'examen (1h30, en fin de semestre).
- Une note de contrôle continu : 2 DMs (en binôme) organisés par le chargé de TD/TP. Note : moyenne des deux DMs.
  - 1 Interrogation SQL
  - 2 Conception

# Evaluation

- Une note d'examen (1h30, en fin de semestre).
- Une note de contrôle continu : 2 DMs (en binôme) organisés par le chargé de TD/TP. Note : moyenne des deux DMs.
  - 1 Interrogation SQL
  - 2 Conception

$$\text{Note finale} = 2/3 \text{ EXAM} + 1/3 \text{ CC}$$

# Plan

1 Organisation

2 Introduction

# But du cours

Introduction aux bases de données relationnelles :

**Interrogation et Conception** de Bases de Données.

Le cours et les TDs sont illustrés par des TPs en SQL (Oracle, Linux).

# Le problème

Comment **stocker**, **utiliser**, **mettre à jour**, ... un grand nombre de données ?

Par exemple :

- 1 les emprunts dans une bibliothèque,
- 2 les salaires des employés dans une entreprise,
- 3 les comptes dans une banque,
- 4 les réservations de places dans les trains, ...

# Un peu de vocabulaire

- **BD**, base de données (DB, database).

# Un peu de vocabulaire

- **BD**, base de données (DB, database).
- **SGBD**, système de gestion de bases de données (DBMS, database management system).
  - L'ensemble des outils (logiciels) permettant **l'organisation, le contrôle, la consultation et la modification** d'une base de données.

# Un peu de vocabulaire

- **BD**, base de données (DB, database).
- **SGBD**, système de gestion de bases de données (DBMS, database management system).
  - L'ensemble des outils (logiciels) permettant **l'organisation, le contrôle, la consultation et la modification** d'une base de données.
- **SGBDR**, système de gestion de base de données relationnelle (RDBMS, relational database management system)
  - un SGBD, qui en plus gère les relations, c'est-à-dire qu'on peut définir des contraintes qui garantissent l'intégrité référentielle et fonctionnelle des données.

# Modèle relationnel de données

Le **modèle relationnel** a été proposé par **Edgar Frank Codd** en 1970.

Il est, de très loin, le modèle le plus utilisé.

# Modèle relationnel de données

Le **modèle relationnel** a été proposé par **Edgar Frank Codd** en 1970.

Il est, de très loin, le modèle le plus utilisé.

Le SGBDR le plus utilisé pour les modèles relationnels est Oracle Database (46 % des bases de données commerciales en 2016), qui est basé sur SQL, *Structured Query Language*.

# SQL

SQL a été adopté comme **norme internationale** par l'ISO en 1987.

Cette norme a connu des révisions, en particulier en 1992 (SQL2), 1999 (SQL3), et ensuite en 2003, 2008 et 2011.

# SQL

SQL a été adopté comme **norme internationale** par l'ISO en 1987.

Cette norme a connu des révisions, en particulier en 1992 (SQL2), 1999 (SQL3), et ensuite en 2003, 2008 et 2011.

**Remarque** : SQL n'est pas un langage de programmation : il ne comporte pas de condition ni de boucle, mais on peut l'englober dans un langage de programmation comme PL/SQL ou y accéder à partir de Java en utilisant JDBC, par exemple (on parlera d'**intégration** SQL).

# Oracle

En 1979, **Relational Software, Inc.** (actuellement **Oracle Corporation**) présente la **première version commercialement disponible de SQL**. Oracle Corporation est maintenant un des principaux éditeurs mondiaux de logiciels.

# Oracle

En 1979, **Relational Software, Inc.** (actuellement **Oracle Corporation**) présente la **première version commercialement disponible de SQL**. Oracle Corporation est maintenant un des principaux éditeurs mondiaux de logiciels.

Un de ses produits phares reste le SGBDR **Oracle Database**.

# Oracle

En 1979, **Relational Software, Inc.** (actuellement **Oracle Corporation**) présente la **première version commercialement disponible de SQL**. Oracle Corporation est maintenant un des principaux éditeurs mondiaux de logiciels.

Un de ses produits phares reste le SGBDR **Oracle Database**.

**ATTENTION** : selon l'implantation de SQL utilisée il peut y avoir des différences concernant la gestion des dates, des chaînes, des valeurs absentes,... Par exemple **Oracle n'est pas toujours conforme à la norme SQL3**.

# Oracle

En 1979, **Relational Software, Inc.** (actuellement **Oracle Corporation**) présente la **première version commercialement disponible de SQL**. Oracle Corporation est maintenant un des principaux éditeurs mondiaux de logiciels.

Un de ses produits phares reste le SGBDR **Oracle Database**.

**ATTENTION** : selon l'implantation de SQL utilisée il peut y avoir des différences concernant la gestion des dates, des chaînes, des valeurs absentes,... Par exemple **Oracle n'est pas toujours conforme à la norme SQL3**.

De même, il y a de légères différences entre les différentes versions de SQL selon les SGBDR (*e.g.*, MySQL, PostgreSQL, SQL Server, ...). **Dans ce cours, nous suivrons la syntaxe Oracle uniquement !**

# Relations

En mathématiques, l'**algèbre élémentaire** définit et étudie les opérations algébriques sur les **nombre**s réels ou complexes : addition, soustraction, multiplication, division, extraction de racine.

De façon similaire, l'**algèbre relationnelle** définit et étudie les opérations sur les **relations**.

# Relations

En mathématiques, l'**algèbre élémentaire** définit et étudie les opérations algébriques sur les **nombre**s réels ou complexes : addition, soustraction, multiplication, division, extraction de racine.

De façon similaire, l'**algèbre relationnelle** définit et étudie les opérations sur les **relations**.

Une **base de données relationnelle** est une base de données structurée suivant les principes de l'algèbre relationnelle.

# Relations

En mathématiques, l'**algèbre élémentaire** définit et étudie les opérations algébriques sur les **nombre**s réels ou complexes : addition, soustraction, multiplication, division, extraction de racine.

De façon similaire, l'**algèbre relationnelle** définit et étudie les opérations sur les **relations**.

Une **base de données relationnelle** est une base de données structurée suivant les principes de l'algèbre relationnelle.

Une **requête** (query) est une demande : c'est l'interrogation d'une base de données pour obtenir une certaine information.

## But du cours BIS

L'**efficacité** du SGBD est fondamentale pour les grosses bases de données, mais dans ce cours d'introduction aux bases de données cette question n'est pas abordée.

# But du cours BIS

L'**efficacité** du SGBD est fondamentale pour les grosses bases de données, mais dans ce cours d'introduction aux bases de données cette question n'est pas abordée.

Par contre la **clarté** est fondamentale (éviter les erreurs, travail en groupe) et cela nécessite :

- un bon modèle formel (le modèle relationnel),
- une bonne conception de la structure des bases,
- une bonne structure pour les requêtes,
- des commentaires et des tests,
- et surtout, ne jamais oublier de spécifier précisément les relations et le résultat des requêtes (qui est aussi une relation).

# Structure d'un SGBD

# Structure d'un SGBD

- ① LMD, langage de manipulation des données (DML, Data Manipulation Language) : pour consulter ou modifier le **contenu** de la base.

# Structure d'un SGBD

- 1 LMD, langage de manipulation des données (DML, Data Manipulation Language) : pour consulter ou modifier le **contenu** de la base.
- 2 LCT, langage de contrôle des **transactions** (TCL, Transaction Control Language) : pour gérer les transactions.

Une transaction regroupe une série de modifications de la base dans une seule opération logique.

# Structure d'un SGBD

- 1 LMD, langage de manipulation des données (DML, Data Manipulation Language) : pour consulter ou modifier le **contenu** de la base.
- 2 LCT, langage de contrôle des **transactions** (TCL, Transaction Control Language) : pour gérer les transactions.

Une transaction regroupe une série de modifications de la base dans une seule opération logique.

- 3 LDD, langage de définition des données (DDL, Data Definition Language) : pour créer ou modifier la **structure** de la base.

# Structure d'un SGBD

- ① LMD, langage de manipulation des données (DML, Data Manipulation Language) : pour consulter ou modifier le **contenu** de la base.
- ② LCT, langage de contrôle des **transactions** (TCL, Transaction Control Language) : pour gérer les transactions.

Une transaction regroupe une série de modifications de la base dans une seule opération logique.

- ③ LDD, langage de définition des données (DDL, Data Definition Language) : pour créer ou modifier la **structure** de la base.
- ④ SQL procedural (PL/SQL pour Oracle Database) : ensemble d'outils pour développer des procédures, déclencheurs (**triggers**) et fonctions utilisateurs (UDF : User Define Function) et pour que SQL s'**interface** avec des langages hôtes.

# Structure d'un SGBD

- 1 LMD, langage de manipulation des données (DML, Data Manipulation Language) : pour consulter ou modifier le **contenu** de la base.
- 2 LCT, langage de contrôle des **transactions** (TCL, Transaction Control Language) : pour gérer les transactions.

Une transaction regroupe une série de modifications de la base dans une seule opération logique.

- 3 LDD, langage de définition des données (DDL, Data Definition Language) : pour créer ou modifier la **structure** de la base.
- 4 SQL procedural (PL/SQL pour Oracle Database) : ensemble d'outils pour développer des procédures, déclencheurs (**triggers**) et fonctions utilisateurs (UDF : User Define Function) et pour que SQL s'**interface** avec des langages hôtes.
- 5 LCD, langage de contrôle des données (DCL, Data Control Language) : pour gérer les **privileges**, *i.e.*, les utilisateurs et les actions qu'ils peuvent entreprendre.

## Plan du cours (1/2)

Le plan du cours suit la structure précédente sauf pour le dernier point : LCD n'est pas traité, à la place une partie du cours porte sur la **normalisation**.

## Plan du cours (1/2)

Le plan du cours suit la structure précédente sauf pour le dernier point : LCD n'est pas traité, à la place une partie du cours porte sur la **normalisation**.

IDEE GENERALE : On peut aussi classer les personnes concernées par une base de données en trois catégories, **du moins « puissant » au plus « puissant »**, et **du plus fréquent au moins fréquent**.

Par exemple, pour gérer les emprunts dans une bibliothèque :

- l'adhérent (**l'utilisateur**) consulte la base,
- le bibliothécaire (**le gestionnaire**) modifie les données, et
- l'informaticien (**le concepteur**) qui a créé la structure de la base peut au besoin modifier cette structure.

## Plan du cours (2/2)

- 1 (DML) Requêtes. Le simple **utilisateur** ne peut, en général, que consulter la base de données.
- 2 (TCL) Transactions. Le **gestionnaire** peut modifier (ajouter, mettre à jour, supprimer) des données (*i.e.*, des valeurs).
- 3 (DDL) Conception. Le **concepteur** crée la structure de la base de données, avec ses contraintes, à partir d'un cahier des charges.
- 4 (PL/SQL) Conception, suite. Le **concepteur** peut avoir besoin d'un langage procédural pour exprimer certaines contraintes.
- 5 (JDBC) Programmation. Le **concepteur** peut avoir besoin d'un langage procédural pour créer une application utilisant une base de données.
- 6 Normalisation. Le **concepteur** doit aussi savoir analyser la qualité de la structure d'une base existante, et l'améliorer si besoin (à l'aide la normalisation).