

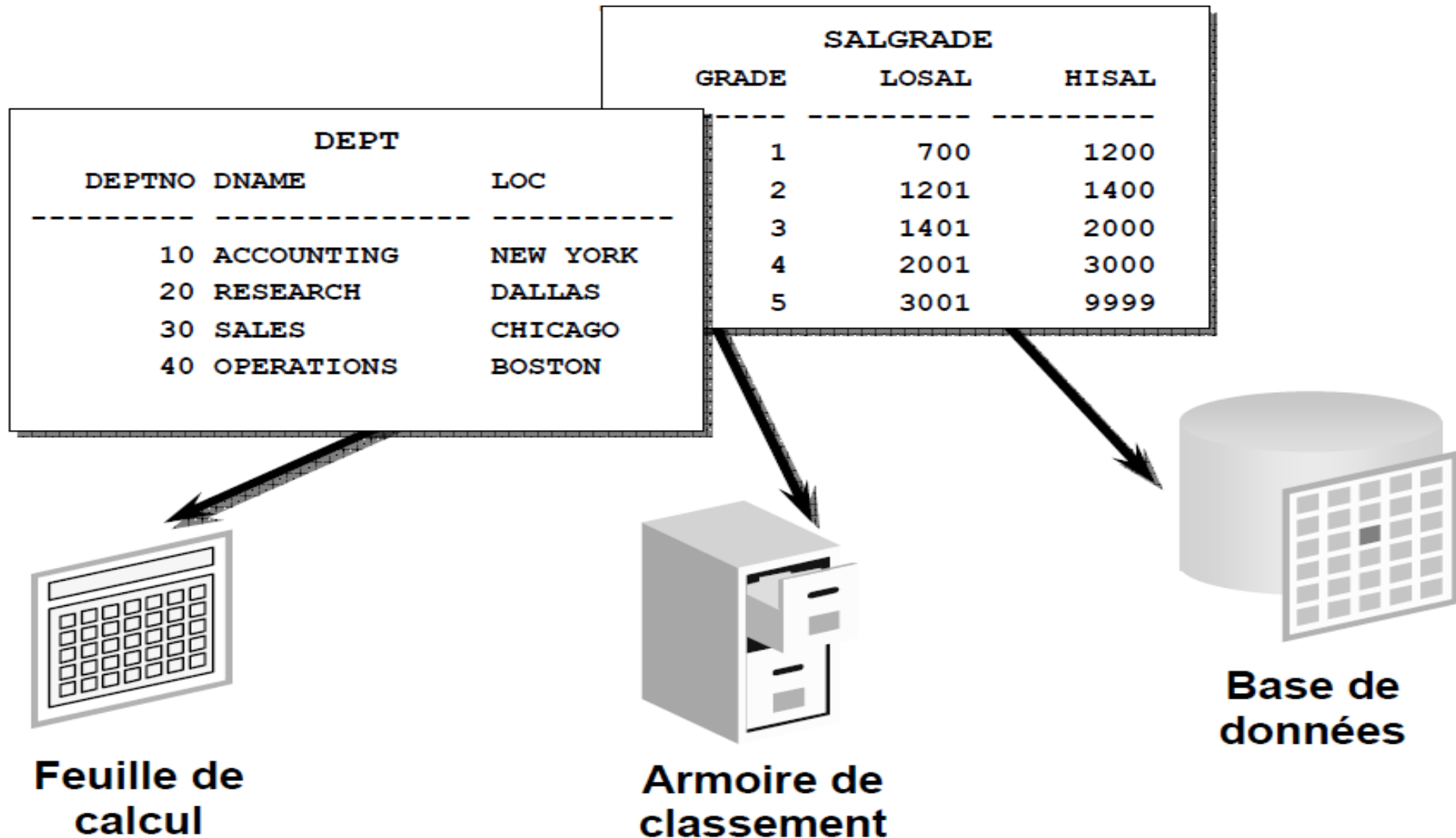
Base de données II

QAFFOU Issam

Plan du cours

- ▶ Introduction
- ▶ Partie I: SQL
 - Langage de définition des données
 - Langage de manipulation des données
 - Langage de contrôle de l'accès aux données
- ▶ Partie II: SQL
 - Introduction
 - Structures de contrôle
 - Exception
 - Curseurs
 - Procédures et Fonction
 - Déclencheurs
- ▶ Partie III: Architecture et Administration

Introduction : Stockage et Organisation des données



Base de données?

- ▶ Définition
- ▶ Types de représentation: Relationnel, Objet, multimédia, réparties, ...
- ▶ Outils: Oracle, MySQL, DB2, ...
- ▶ Adoption: Oracle (version $\geq 10g$)

Installation et prise en main

- ▶ Télécharger la version gratuite Oracle XE l'installer sur un ordinateur personnel. Cette version fonctionne en mode client-serveur.
- ▶ Lien de téléchargement:
<https://www.oracle.com/database/technologies/appdev/xe.html>
- ▶ Deux possibilités pour manipuler Oracle : interface graphique ou interface de commande (SQL*Plus).

Connexion Oracle

- ▶ **Mode graphique :**

Lancer la page d'accueil de la base de données 127.0.0.1:8080/apex

- ▶ **Mode ligne de commande SQL :**

Dans une fenêtre de commandes, lancez sqlplusSQL*PLUS. Un nom d'utilisateur et un mot de passe sont demandés.

Connect nom_utilisateur/mot_de_passe

Vous vous connectez en tant que SYSTEM avec le mot de passe que vous avez fourni lors de l'installation.

SQL*Plus

- ▶ SQL*Plus est l'outil de ligne de commande d'Oracle qui permet d'exécuter la suite de commandes SQL (Structured Query Language) standard.
- ▶ SQL est un langage fonctionnel qui permet aux utilisateurs de communiquer avec Oracle pour ajouter, mettre à jour ou modifier des données dans la base, ou en extraire des données.
- ▶ A l'aide de SQLPlus, exécutez l'interrogation suivante pour vérifier que la connexion à la base de données a été établie.

```
SQL> SELECT * FROM DUAL;
```

Pourquoi étudier SQL?

- ▶ Si vous allez travailler avec une base de données relationnelle, si vous écrivez des applications, des tâches administratives, ou la génération de rapports, vous aurez besoin de savoir comment interagir avec les données dans votre base de données.
- ▶ Même si vous utilisez un outil qui vous génère SQL, il peut y avoir des moments où vous avez besoin de contourner la fonctionnalité de génération automatique et écrire vos propres requêtes SQL.

Apprendre le SQL vous force à confronter et à comprendre les structures des données utilisées pour stocker des informations sur votre organisation, comme vous pouvez vous trouvez proposer des modification ou des ajouts à votre base de données.

Objectifs de la partie 1

- ▶ Comprendre SQL: définitions, utilités, atouts, concepts et performances;
- ▶ Comprendre la structure générale de SQL;
- ▶ Bien assimiler le modèle relationnel à partir de la manipulation de SQL;
- ▶ Se familiariser avec les clauses d'interrogation;
- ▶ Savoir quand et pourquoi utiliser chaque clause.

Historique rapide

- ▶ CODD a défini le modèle relationnel en 1970. Pour manipuler les données stockées dans des tables relationnelles, il a proposé un langage appelé DSL/Alpha;
- ▶ Un groupe de IBM a construit un prototype basé sur les idées de CODD. Ils ont créé une version simplifiée de DSL/Alpha qu'ils ont nommé SQUARE;
- ▶ Un raffinement de SQUARE a mené à un langage appelé SQUEL qui est enfin nommé SQL.

Définition et Présentation

- ▶ SQL (**S**tructured **Q**uery **L**anguage): langage d'interrogation structuré.
- ▶ C'est un langage complet de gestion de base de données relationnelles.
- ▶ **Intérêt**: interaction avec la base de données; interroger/ordonner la base de données avec des commandes pour en recevoir des résultats précis.

► SQL est à la fois:

- un langage d'interrogation de la base
ordre: **SELECT**
- un langage de définition des données
ordres: **CREATE, ALTER, DROP**
- un langage de manipulation des données
ordres: **UPDATE, INSERT, DELETE**
- un langage de contrôle de l'accès aux données
ordres: **GRANT, REVOKE**

Objets manipulés par SQL

- ✓ Identificateurs
- ✓ Tables
- ✓ Colonnes

Identificateurs

- ▶ SQL utilise des identificateurs pour désigner les objets qu'il manipule: utilisateurs, tables, colonnes, index, fonctions, etc.
- ▶ Un identificateur est un mot formé d'au plus 30 caractères (une lettre, un chiffre, ou l'un des symboles: # \$ et _);
- ▶ Il commence obligatoirement par une lettre de l'alphabet; Il ne doit pas figurer dans la liste des mot clés réservés (voir manuel de référence Oracle).

Voici quelques mots clés que l'on risque d'utiliser comme identificateurs:

ASSERT	FILE	PRIVILEGES	SET
ASSIGN	FORMAT	PUBLIC	TABLE
AUDIT	INDEX	REF	TYPE
COMMENT	LIST	REFERENCES	
DATE	MODE	SELECT	
DECIMAL	OPTION	SEQUENCE	
DEFINITION	PARTITION	SESSION	

Tables

Les relations (d'un schéma relationnel ; voir le cours sur le modèle relationnel) sont stockées sous forme de tables composées de lignes et de colonnes.

Colonnes

- ▶ Les données contenues dans une colonne doivent être toutes d'un même type de données. Ce type est indiqué au moment de la création de la table qui contient la colonne.
- ▶ Chaque colonne est repérée par un identificateur unique à l'intérieur de chaque table.
- ▶ Deux colonnes de deux tables différentes peuvent porter le même nom.
- ▶ Une colonne peut porter le même nom que sa table.
- ▶ Le nom complet d'une colonne est en fait celui de sa table, suivi d'un point et du nom de la colonne.

Contraintes d'intégrités (CI)

- * CI de domaine;
- * CI de relation;
- * CI de référence.

Soit le schéma relationnel suivant :

Acteur (**Num-Act**, Nom, Prénom)

Jouer (*Num-Act*, *Num-Film*)

Film (**Num-Film**, Titre, Année)

CI de domaine

- ▶ Toute comparaison d'attributs n'est acceptée que si ces attributs sont définis sur le même domaine. Le SGBD doit donc constamment s'assurer de la validité des valeurs d'un attribut. C'est pourquoi la commande de création de table doit préciser, en plus du nom, le type de chaque colonne.
- ▶ Lors de l'insertion des n-uplets (lignes) dans cette table, le système s'assurera que les différents champs du n-uplet satisfont les contraintes d'intégrité de domaine des attributs précisées lors de la création de la base. Si les contraintes ne sont pas satisfaites, le n-uplet n'est, tout simplement, pas inséré dans la table.

CI de relation

- ▶ Une clé primaire ne peut pas avoir une valeur nulle. Une clé primaire doit toujours être unique dans une table. Cette contrainte forte qui porte sur la clé primaire est appelée contrainte d'intégrité de relation.
- ▶ Tout SGBD relationnel doit vérifier l'unicité et le caractère défini (NOT NULL) des valeurs de la clé primaire.

CI de référence

- ▶ Lors de l'insertion d'un n-uplet dans la relation Jouer, le SGBD doit vérifier que les valeurs Num-Act et Num-Film correspondent bien, respectivement, à une valeur de Num-Act existant dans la relation Acteur et une valeur Num-Film existant dans la relation Film.
- ▶ Lors de la suppression d'un n-uplet dans la relation Acteur, le SGBD doit vérifier qu'aucun n-uplet de la relation Jouer ne fait référence, par l'intermédiaire de l'attribut Num-Act, au n-uplet que l'on cherche à supprimer.