

Langage de Requêtes (Bases de Données)

Lynda Tamine-Lechani

`lynda.lechani@irit.fr`

Objectifs du cours

Les principaux objectifs de ce cours sont :

1. Introduire les **Systeme de Gestion de Bases de Données (SGBD)**
2. Apprendre les bases du **langage SQL**
3. Apprendre les bases de programmation d'applications bases de données : **langage PL/SQL**

Organisation de l'enseignement

A) Cours : 12 H (6 séances)

- Diffusion par vidéo-projecteur
- Supports de cours fournis en début de chapitre
- Explications et illustrations détaillées chaque séance

B) Travaux dirigés (TD) : 6H (3 séances)

- Séries d'exercices en lien avec le cours
- Corrections pendant la séance de TD

C) Travaux Pratiques (TP) : 14 H (7 séances)

- Sujets de TP associés à chaque chapitre du cours
- Réalisation par monôme
- Dépôt de comptes rendus sur Moodle. Les consignes seront données sur le sujet de TP.

Modalités de Contrôle des Connaissances

1^{ère} session

- **Contrôle Continu (CC)** : 2 Comptes rendus (CR) de TP au choix notés (CC_1 , CC_2)



Outre le contenu du CR, la notation des CC comprendra :

- l'assiduité
- la lisibilité du CR
- Le respect des délais de remise du CR

- **Contrôle Terminal (CT)** : épreuve écrite (2H)

$$\text{Note} = 0 \cdot 3 (CC_1 \cdot 0.5 + CC_2 \cdot 0.5) + 0.7 \cdot CT$$

2^{ème} session

- **Contrôle Terminal (CT)** : Epreuve écrite (2H)
- **Pas de report de la note de CC**

Espace de l'UE-Moodle

UE ELMAB5F1 : Langages de requêtes

- Dépôt des support de cours
- Dépôt des supports de TD
- Dépôt des supports de TP
- Dépôt des CR de TP
- Forum : questions, discussions, informations, ...



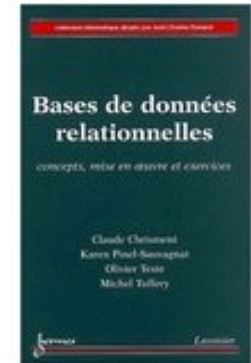
Il faut s'inscrire à l'UE dès aujourd'hui !

Chapitres du cours

- Chapitre 1 : Introduction aux BD et SGBD
- Chapitre 2 : Le langage SQL
- Chapitre 3 : Bases du langage PL/SQL

Bibliographie

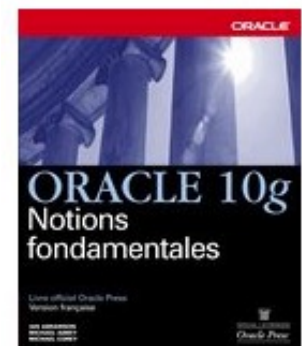
- *Bases de données relationnelles : concepts, mise en œuvre et exercices.* C.Chrisment et al, Edition Hermes Lavoisier, 2008



- *Bases de données.* G. Gardarin, Editions Eyrolles, 2003



- *ORACLE9i notions fondamentales.* M. Abbey et al, CAMPU Press, 2002



Planification des séances de cours/TD

Séance	Contenu
1	Introduction aux BD-SGBD; introduction au LDD
2	LDD : création et mise à jour de tables- TD 1
3	LMD, TD 1 suite
4	LID : bases de la requête SELECT; TD 2
5	LID : jointures, regroupement, synchronisation; TD 2 suite
6	PL/SQL : bases, exceptions
7	Gestion des curseurs, TD 3
8	PL/SQL : procédures, fonctions, paquetages
9	TD3 suite

Planification des séances de TP

Séance	Contenu
1	Mise en œuvre BD : LDD
2	Manipulation, interrogation de BD : LMD LID
3	Interrogation de BD : LID (suite)
4	PL/SQL : principes de base, exceptions
5	PL/SQL : gestion des curseurs
6	PL/SQL : Procédures
7	PL/SQL : Fonctions; paquetages

Chapitre 1

Introduction aux BD et SGBD

Plan du chapitre 1

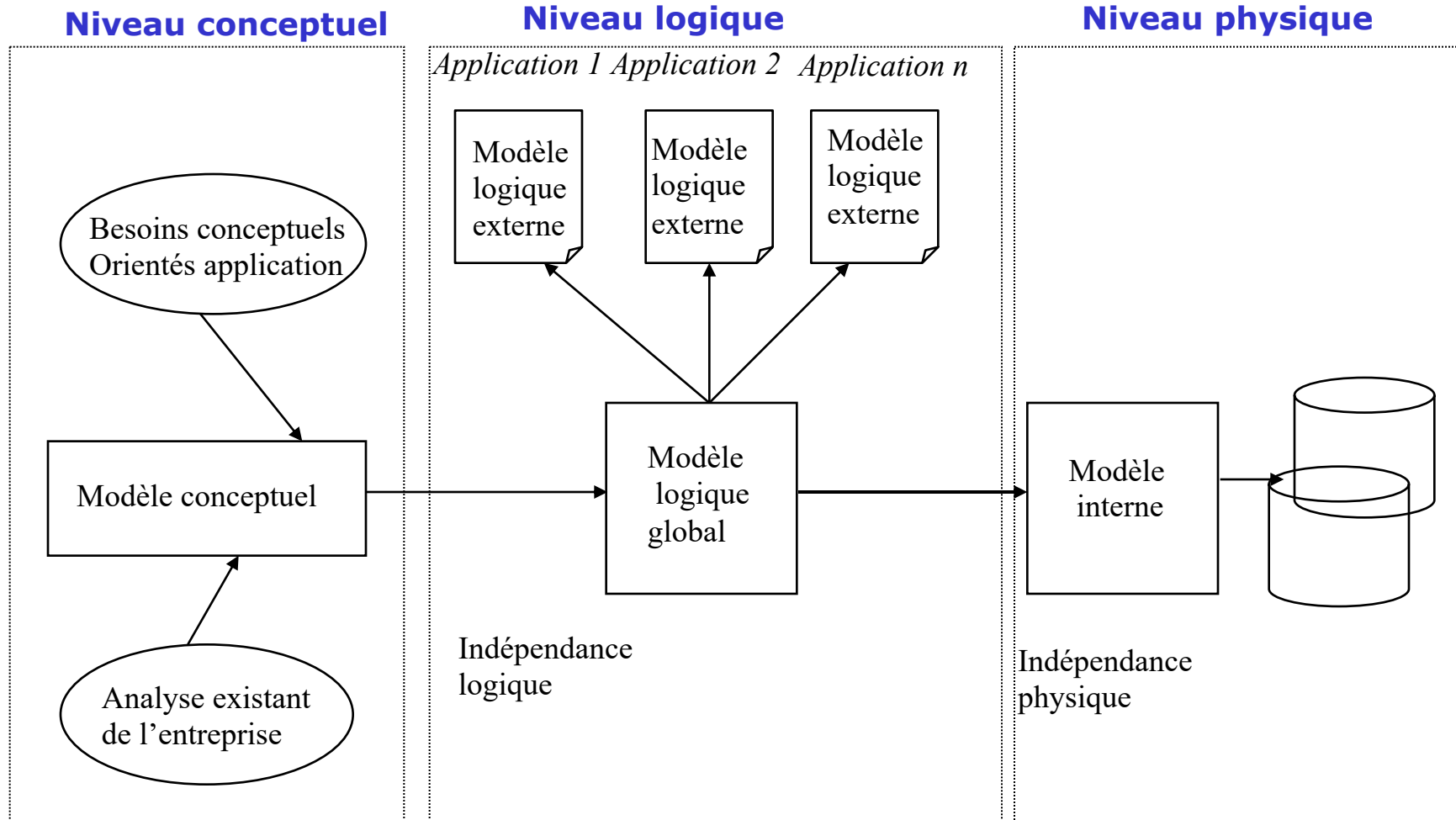
- Notions de BD et SGBD
- Éléments du modèle relationnel
- SQL et modes d'accès à Oracle

Bases de Données (BD) : c'est quoi ?

□ Base de Données (*BD*)?

- Un ensemble structuré d'informations agrégées ou élémentaires accessibles par une communauté d'utilisateurs [Chrisment,08]
- Une collection de données qui intègre :
 - une structure intégrée,
 - des liaisons sémantiques entre données,
 - des contraintes d'intégrité,
 - des vues de différents utilisateurs.
- Une collection de données qui supporte des opérations de manipulation et de recherche de données :
 - cohérente,
 - sécurisée,

Bases de données : c'est quoi ?

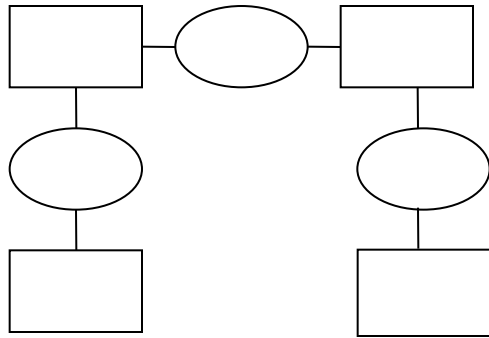


Niveaux de modélisation dans une Base de Données

Bases de Données (BD) : c'est quoi ?

□ Base de Données (*BD*)?

Ensemble d'**entités**
liées par des
associations

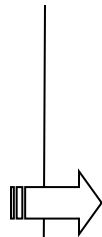


Ensemble de **relations**
décrites par des **attributs**

R1 (att1, att2,..)

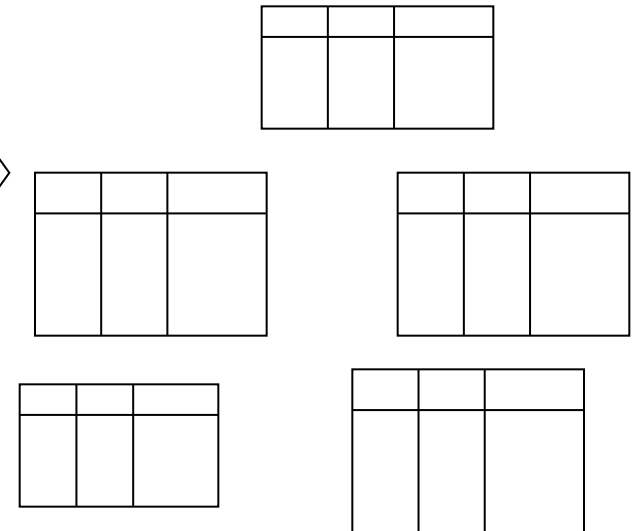
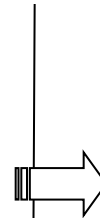
R2 (att1, att2,..)

Rn (att1, att2,..)



Modèle relationnel

Ensemble de **tables**
décrites par des colonnes



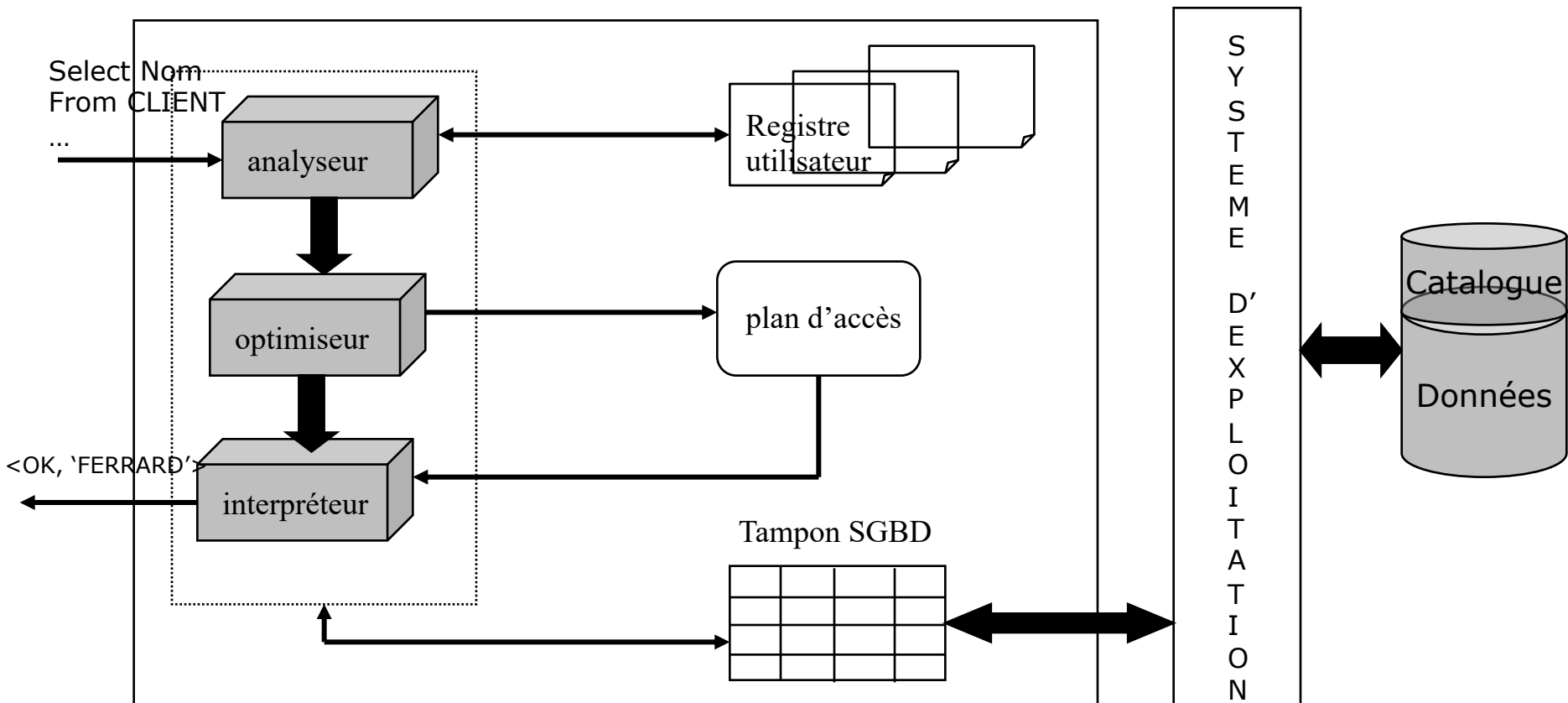
Modèle physique

MCD Entité-Association

Système de Gestion de Bases de données (SGBD) : c'est quoi ?

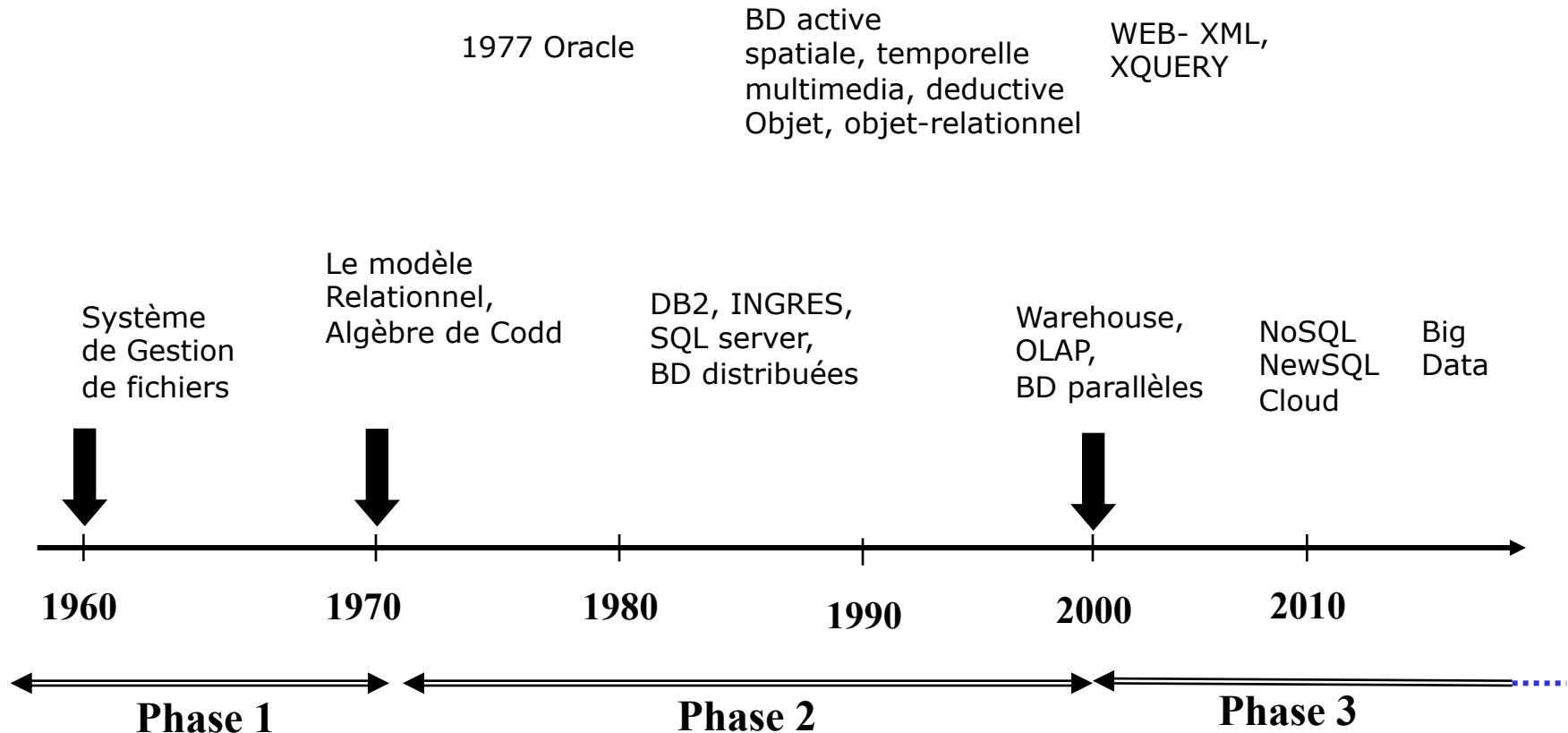
□ Système de Gestion de Base de Données (SGBD)?

- Un système (logiciel) qui implante les fonctions de sauvegarde, d'interrogation, de recherche et de mise en forme des données de la base de données



*Eléments extraits de Tutoriel,
M. Adiba, EDBT 2013*

BD et SGBD : Historique



Systeme de Gestion de Bases de données : *quoi autour ?*

□ SGBD : les outils

- **Les outils frontaux (4-GL)**
 - Générateurs : de formes, de rapports, des applications
 - Intégrés au SGBD ou externes
 - Powerbuilder, Borland...
 - Interfaces WEB : HTML, XML...
 - Interfaces OLAP & Data Mining
 - Intelligent Data Miner (IBM)
- **Utilitaires : chargement, statistiques, aide à la conception...**

Systeme de Gestion de Bases de données :

quoi autour ?

□ SGBD : les usagers

- **Interactifs (ad-hoc)**
 - Cherchent les informations, sans connaître la structure de la BD, utilisent des :
 - Interfaces visuelles: 4-GL, Web...
 - Langages assertionnels (QBE)
- **Développeurs d'application**
 - Construisent les programmes de traitement des données + interfaces pour les usagers interactifs
 - Utilisent des langages SQL, PL/SQL, etc...
- **Administrateur**
 - Utilisateur prioritaire
 - Définit et maintient la structure de la BD

Systeme de Gestion de Bases de données : *quoi autour ?*

☐ SGBD : les plus populaires



Firebird



TERADATA.



SGBD : *pourquoi ?*

□ **SGBD : *principaux objectifs***

- Indépendance physique/logique des programmes aux données
- Manipulation des données par des langages non procéduraux
- Administration aisée des données
- Efficacité de l'accès aux données
- Partage des données
- Cohérence des données
- Redondance contrôlée des données
- Sécurité des données

SGBD : *quelles fonctions ?*

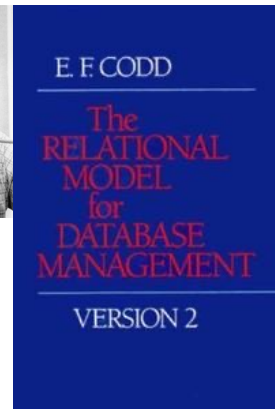
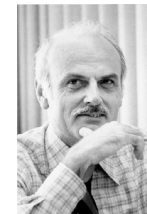
□ Principales fonctions

- Description des objets
- Recherche et mise à jour des objets (accès + vues+optimisation)
- Contrôle des données (intégrité, autorisation)
- Gestion de transactions et sécurité (concurrency+sécurité)

BD et SGBD : Le modèle relationnel

□ Modèle relationnel

- Inventé par *Edgar Franck Codd* en 1970
- Fondements :
 - Algèbre relationnelle, logique de prédicat de 1^{er} ordre
 - Indépendance des données, vue tabulaire
 - Langages : SQL, QUEL, QBE
 - Dépendances fonctionnelles, formes normales
- Prototypes SGBDR (1975)
- SGBDR commercialisés en 1980



BD et SGBD : Le modèle relationnel

□ Notions de base

- Théorie des ensembles E. F. Codd, 1970
- BD : ensemble de relations $BD = \{\text{Relation}\}$
- Relation = {Schéma, Tuples}
- Schéma de relation est la description d'une relation à l'aide :
 - d'**attributs** caractéristiques associés à des **domaines** de valeurs,
 - Tuple : **instance** de relation
- Attribut, Domaine
- Schéma de relation, Tuple

Attribut, Domaine

□ Attribut

- Caractéristique de l'objet décrit par la relation
- Propriété d'une relation qualifiée par un nom
- Le i ème attribut d'une relation R est noté A_i
- Exemples : Nom (de l'étudiant), Age (de l'étudiant), Ville (de l'étudiant)...
- Attribut

□ Domaine

- Un attribut prend une valeur comprise dans un ensemble de valeurs (fini ou infini) appelé domaine, cad :
- $Dom(A_i)$ est l'ensemble de valeurs associé à l'attribut A_i
- Exemples : $Dom(Nom) = \{Dupont, Durant, Didier, \dots\}$,
 $Dom(Age) = \{15, 23, 56, \dots\}$, $Dom(Ville) = \{Paris, Bordeaux, Toulouse, \dots\}$

Schéma de relation, Tuple

□ Schéma de relation

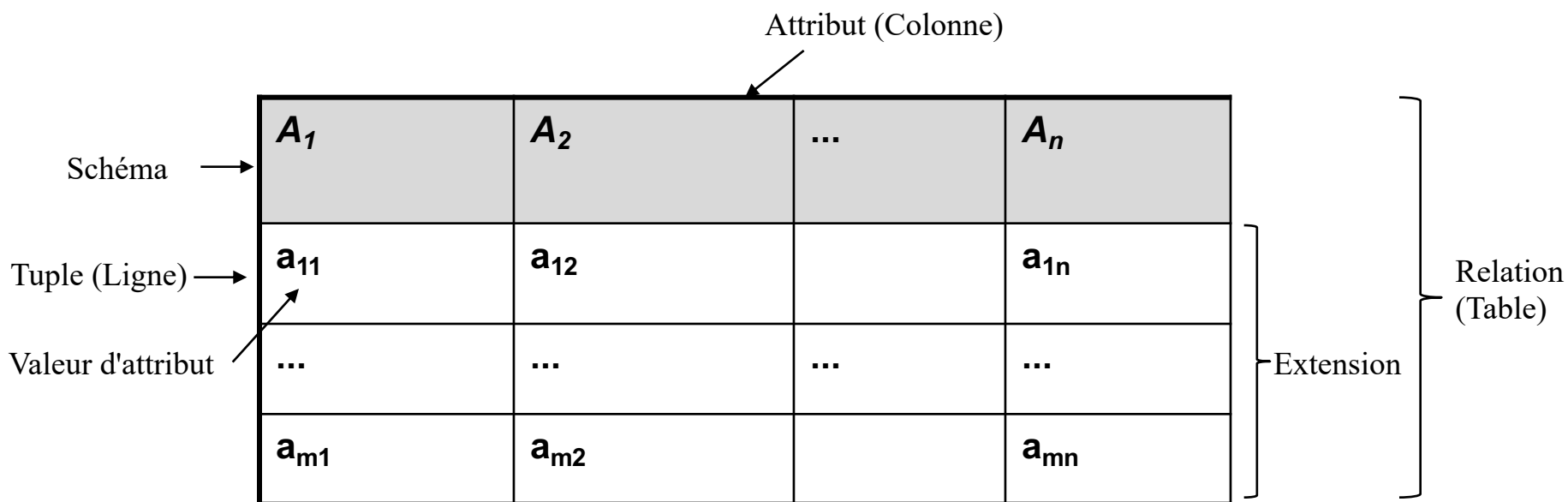
- Le **schéma** d'une relation est sa description en **intention**
- Le schéma d'une relation est déterminé par l'ensemble des attributs de cette relation
- Le schéma SR de la relation R est noté $R [A_1, A_2, \dots, A_n]$ où R est le nom de la relation.
- Le **degré** de la relation R est déterminé par le nombre de ses attributs, soit n

□ Tuple

- Un tuple T est « une réalisation » de la relation R obtenue par attribution d'une **valeur atomique** a_i à chacun des attributs A_i constituant son schéma : $T = \langle a_1, a_2, \dots, a_n \rangle$ où $a_i \in \text{Dom}(A_i)$. On note $a_i = T[A_i]$
- L'ensemble des tuples d'une relation $ER = \{T\}$ est sa description en **extension**, il n'existe pas deux tuples égaux dans R !
- La **cardinalité** d'une relation est déterminée par le nombre de ses tuples

Exemple

Représentation de relation sous forme tabulaire : ordre des lignes sans importance



Exemple

RefProd	Designation	Catégorie	Prix	Stock
18	Tabouret	A	56,78	200
23	Chaise	B	13,00	350

Schéma : R (RefProd, Designation, Catégorie, Prix, Stock)

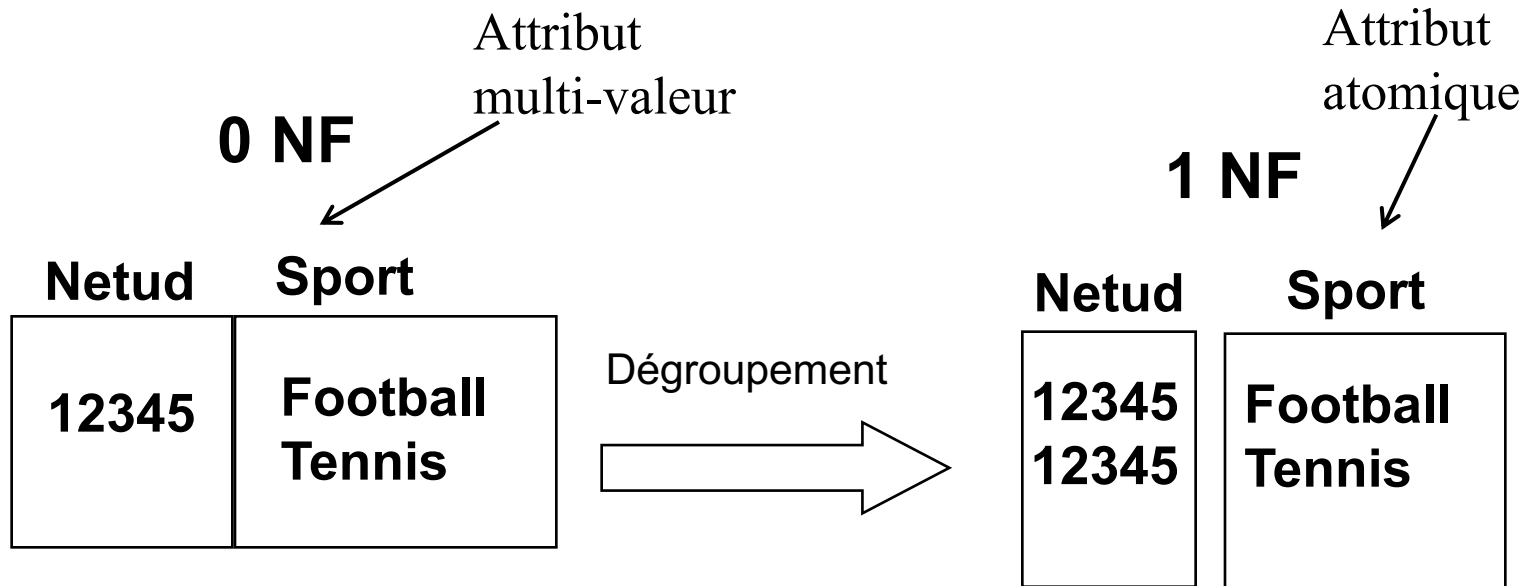
Degré : Degré(R) = 5

Tuples (R) : <18, 'Tabouret', 'A', 56,78, 200>;
<23, 'Chaise', 'B', 13,00, 350>;

Cardinalité : Cardinalité (R) = 2

Toute relation est en 1^{ère} forme normale

Un étudiant peut pratiquer plusieurs sports



Notion de clé primaire

□ **Objectif** : identifier pour **chaque relation** et de **manière unique** chaque tuple. Une clé est **unique** et **non-nulle**

Une clé est définie pour chaque relation

□ **Définition**

Une clé C de R est une combinaison d'attributs de R telle que :

(1) dans tout tuple T de R , on a $T(C)$ valeurs clé de T , identifie T

($\forall T, T'$ tuples de $R, T(C) \neq T'(C)$)

(2) il n'existe pas de sous-combinaison de C avec cette propriété (1)

□ **Autrement dit**

Soit - $R [A_1, \dots, A_n]$ et

- $R_b [A_k, A_l]$: les relations issues de la bipartition de R

On dit que A_k est clé de R ssi :

(1) $A_k \rightarrow A_l$

(2) $\forall A' \subset A_k, A' \rightarrow A_l$



Notation : la clé est soulignée

Notion de clé primaire - Exemples

PRODUIT

RefProd	Designation	Categorie	Prix	Stock
18	Tabouret	A	56,78	200
23	Chaise	B	13,00	350
25	Table	B	123,00	36

LigneComm

RefCom	RefProd	Quantite
21A12	23	18
34B10	25	23

Commande

RefCom	CodCli
1A	18
2T	2

Client

CodCli	Nom	Prenom
1	TOTO	Martin
2	TITI	Amélie

Catégorisation des clés, attributs clés

❑ **Clé atomique / composite**

- Clé **atomique** : clé composé d'un seul attribut
- Clé **composite** : clé composée de plusieurs attributs. Le nombre d'attributs détermine la cardinalité de la clé

❑ **Attribut clé / Attribut non clé**

- Tout attribut d'une clé est dit attribut-clé
- Tout autre attribut est un attribut non clé

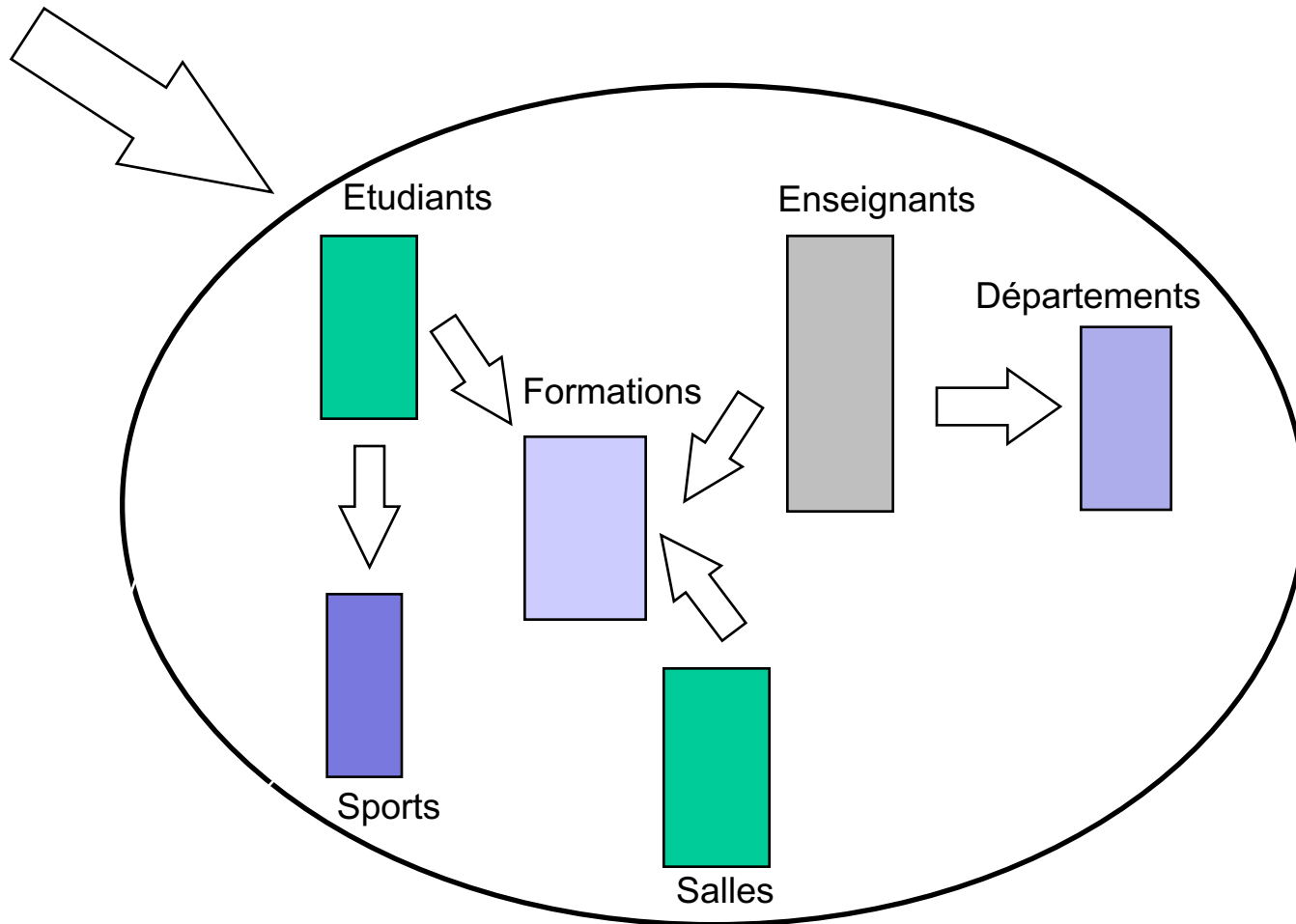
❑ **Clé primaire/ secondaire** : une relation peut avoir plusieurs clés

- Une clé est choisie : c'est la clé primaire
- Les autres clés sont candidates

❑ **Clé minimale** : une relation peut avoir plusieurs clés de cardinalités différentes

- La clé minimale est la clé de plus petite cardinalité

Dépendance vs Indépendance des données ?



Contrainte d'intégrité référentielle

□ Clé étrangère d'une relation : c'est quoi ?

- C'est une contrainte sémantique d'inclusion qui :
 - ✓ possède une source composée d'un ou plusieurs attributs s_i (clé étrangère)
 - ✓ possède une cible composée d'un ou plusieurs attribut clés c_j (clé)
 - ✓ traduit une relation d'inclusion $s_i \text{ inclus } c_j$



La clé primaire et la clé étrangère peuvent être dans la même relation !



Notation : l'attribut qui référence la clé est précédée du symbole #

Retour sur l'exemple

RefProd	Designation	Categorie	Prix	Stock
18	Tabouret	A	56,78	200
23	Chaise	B	13,00	350
25	Table	B	123,00	36

PRODUIT

RefCom#	RefProd#	Quantite
21A12	23	18
34B10	25	23

LigneComm

RefCom	CodCli#
21A12	18
34B10	2

Commande

CodCli	Nom	Prenom
18	TOTO	Martin
2	TITI	Amélie

Client

Notion de Méta-base (Méta-base)

❑ Ensemble de tables en lecture seule

❑ Contenu des tables

- les informations concernant les structures logiques et physiques de la base : tables, vues, procédures, etc.
- les utilisateurs d'une base Oracle
- les informations concernant les contraintes d'intégrité définies sur les tables
- l'espace alloué pour un objet du schéma ainsi que l'espace réellement utilisé

❑ Un dictionnaire de données est créé dès la création de la base

- Automatiquement mis à jour par Oracle en réponse à des actions spécifiques (`CREATE`, `DROP`, etc.)

Notion de méta-base

❑ Vues interrogeables par les utilisateurs

- Vues préfixées par «dba_»
 - ✓ Contiennent les informations sur l'ensemble des objets de la BD. Ces vues sont interrogeables par les administrateurs
 - ✓ Exemple : dba_tables liste toutes les tables contenues dans la BD
- Vues préfixées par «all_»
 - ✓ Contiennent les informations sur les objets accessibles par l'utilisateur
 - ✓ Exemple : all_tables liste les tables que l'utilisateur peut manipuler
- Vues préfixées par «user_»
 - ✓ Contiennent les informations sur les objets dont l'utilisateur est propriétaire (owner)
 - ✓ Exemple : user_tables liste les tables dont l'utilisateur est propriétaire

SQL ? (1)

□ Structured Query Language = Langage d'interrogation structuré

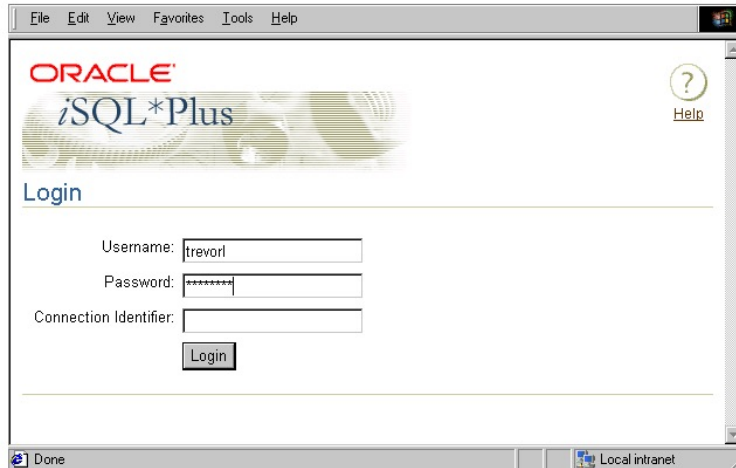
- Langage de gestion de bases de données relationnelles
- Permet :
 - ✓ de **définir les données** : création/suppression de tables, index, modification de structures des tables, ...
 - ✓ de **manipuler des données** : mises à jour des données
 - ✓ d'**interroger des données** : consulter, synthétiser ..
 - ✓ de **contrôler des données** : gestion des droits objets de la BD

SQL ? (2)

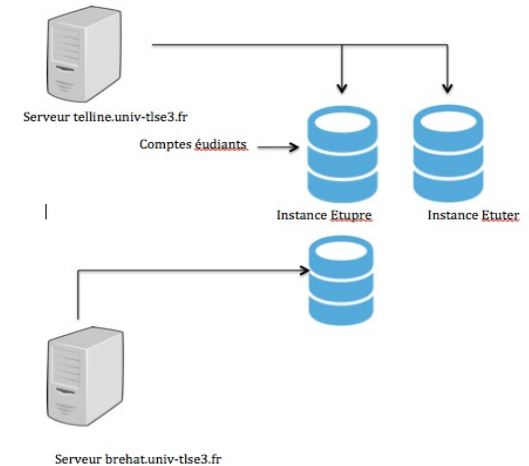
❑ Sous langages SQL

- **LDD : Langage de Définition des Données**
 - ✓ Création, modification, suppression du schéma de la BD, des composants du schéma de la BD (colonne, contrainte etc.)
CREATE, ALTER, RENAME, DROP
- **LMD : Langage de Manipulation des Données**
 - ✓ Manipulation des données : ajout, suppression, modification
INSERT, UPDATE, DELETE
- **LID : Langage d'Interrogation des Données**
 - ✓ Sélection de données
INSERT
- **LCD : Langage de Contrôle des Données**
 - ✓ Contrôle de l'accès aux données : droits d'accès, vues etc.
GRANT, REVOKE
- **LCT : Langage de Contrôle des Transactions**
 - ✓ Validation et annulation des transactions
COMMIT, ROLLBACK

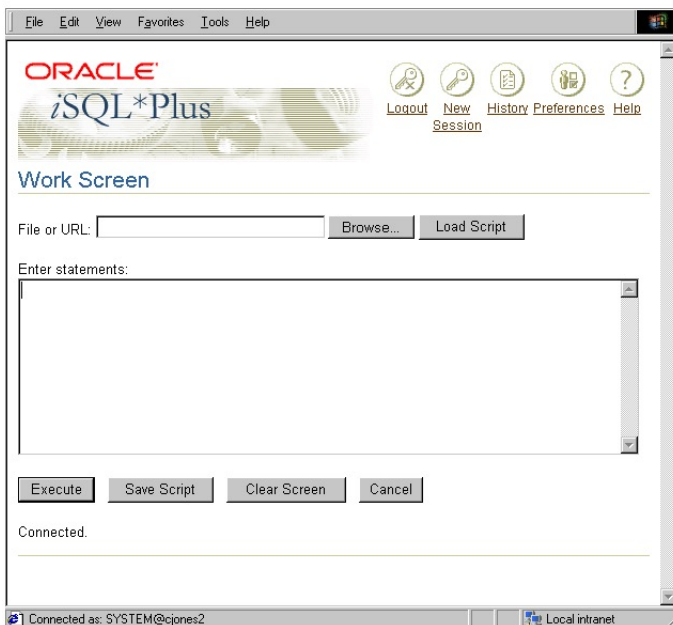
Le client léger iSQL*plus via navigateur web



Interface de connexion iSQL* à une base Oracle

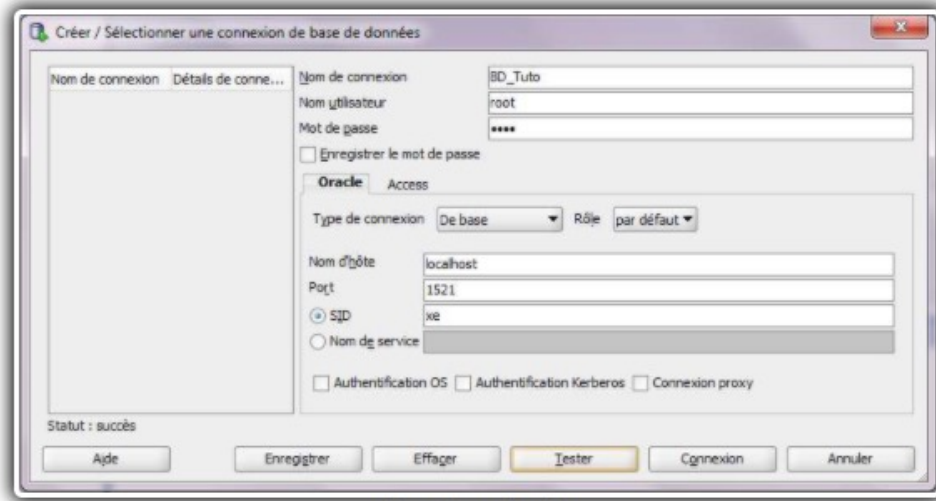


Architecture de votre environnement en TP : instance ETUPRE

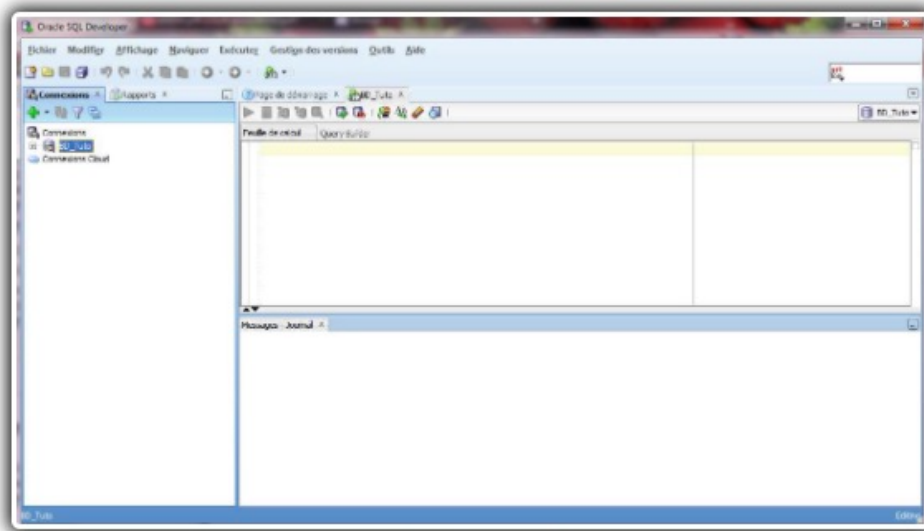


Interface espace de travail iSQL*

Le client riche Oracle SQL Developer



Interface de connexion SQL Developer à une base Oracle



Interface espace de travail SQL Developer