

TP 1

(Séances 1, 2 et 3)

1 Connexion à Oracle

Le serveur Oracle est sur une machine serveur sur laquelle on ne se connecte pas (a priori). Chaque machine sous linux possède un client **SQLPLUS** (interpréteur de commandes **SQL** et module d'Oracle). La documentation relative à l'environnement **SQLPLUS** est accessible par un "butineur" à l'URL **http ://neptune/oracle10**.

Comme l'éditeur de commandes sous **SQLPLUS** est "ligne à ligne", il est conseillé d'utiliser un éditeur externe (exemple **xemacs**). Le fichier a pour extension **.sql**.

Pour se connecter, dans une fenêtre **xterm** lancer **SQLPLUS** par la commande

```
sqlplus login/password@venus/licence
```

Après le signe d'invite SQL> taper les commandes SQL. Pour exécuter le fichier nomfichier.sql, vous tapez *start nomfichier.sql*.

Les minuscules ne sont pas différenciées des majuscules pour ce qui concerne la syntaxe réservée.

2 Session SQL interactif

L'exécutable **SQLPLUS** est un interpréteur/éditeur de commandes, chaque commande **SQL** pouvant être saisie sur une ou plusieurs lignes, le point virgule obligatoire termine la commande. Attention à la syntaxe qui est très rigoureuse. Voir l'exemple suivant : attention aux parenthèses, virgules et surtout au point virgule qui termine une commande.

```

CREATE TABLE acteur
(
idA NUMERIC(5) PRIMARY KEY,
nom CHARACTER(40) NOT NULL,
prenom CHARACTER(40),
nationalite CHARACTER(40)
);

```

3 La base de données

Soit une base de données représentée par les tables suivantes :

- acteur(idA, nom, prénom, nationalité) : “idA” est la clé primaire
- réalisateur(idR, nom, prénom, nationalité) : “idR” est la clé primaire
- genre(idG, description) : “idG” est la clé primaire
- film(idF, titre, année, pays, nbspectateurs, idRéalisateur, idGenre) : “idF” est la clé primaire, “nbspectateurs” est > 0
- jouer(idActeur, idFilm, salaire) : (idActeur, idFilm) est la clé primaire, “salaire” est > 0

4 Création de la base de données

1. Créer les tables données ci-dessus (ne pas oublier les contraintes associées de domaine, de clés primaires et d'intégrité référentielle). Pour ce faire, vous allez sous Xemacs créer un fichier script contenant vos ordres SQL. Sauvegardez votre fichier avec l'extension `.sql`. Réalisez des insertions dans chaque table créée. Exécuter ce script dans la session **SQL** interactive par la commande `start “votreNomDeFichier”`. Vérifier la définition et le contenu des tables créées.
2. “idRéalisateur” et “idGenre” dans la table “film” sont des clés étrangères.

5 Insertion de données

Insérer quelques données.

6 Requêtes de définition et de mise à jour

- Insérer deux acteurs différents avec le même identifiant. Que se passe-t-il ?
- Réaliser la mise à jour qui consiste à augmenter le salaire dans la table “jouer” de 20%.
- Supprimer un réalisateur.

7 Requêtes de consultation

Traduire les requêtes suivantes en SQL :

1. Les acteurs ayant joué dans des comédies ou des films policiers
2. Les films tournés par Daroussin depuis 2000
3. Les couples (réalisateurs,acteurs) par ordre réalisateurs, acteurs
4. Les genres des films dans lesquels a joué Al Pacino
5. Les réalisateurs de comédies musicales par ordre chronologique
6. Le nombre de films enregistrés dans la base de données
7. L’année du film le plus ancien
8. Le salaire maximum, minimum et moyen des acteurs
9. Le nombre maximum, minimum et moyen de spectateurs qui sont allés voir un film comique
10. Le ratio entre le salaire moyen d’Al Pacino et le nombre de spectateurs moyen qui ont vu ses films

Pour préparer le prochain TP, créer la base de données suivante (les clés primaires sont soulignées et les clés étrangères sont en *italique*) :

- Opérateur(numéro, nom, âge)
- Machine(référenceM, désignationM)
- Pièce(référenceP, désignationP, *référence_machine*, *numéro_opérateur*, quantité)

– Qualifié_sur(numéro_opérateur, référence_machine, date)

L'ensemble des données (schéma physique) est présenté dans les tables suivantes :

numéro	nom	âge
OP42	Robert	32
OP10	Sophie	41
OP78	Lucette	25
OP22	Albert	25
OP57	Marc	38

TAB. 1 – Table “Opérateur”.

référenceM	désignationM
M12	Perceuse
M13	Ponceuse
M14	Tour numérique

TAB. 2 – Table “Machine”.

référenceP	désignationP	référence_machine	numéro_opérateur	quantité
P1	Pièce1	M12	OP10	250
P2	Pièce2	M12	OP22	600
P3	Pièce3	M14	OP22	200
P4	Pièce4	M13	OP78	150

TAB. 3 – Table “Pièce”.

numéro_opérateur	référence_machine	date
OP10	M12	15/01/00
OP22	M12	20/05/01
OP10	M13	10/10/99
OP42	M13	17/01/02
OP78	M12	19/07/98
OP10	M14	04/05/01

TAB. 4 – Table “Qualifié_sur”.

TP 2

Séance 4

Exercice 1

On reprend la base de données (acteur, film, jouer, réalisateur, genre). Traduire les requêtes suivantes en SQL :

1. le nombre de films qui ont été tournés chaque année.
2. pour chaque année, le nombre maximum de spectateurs qui ont vu un film.
3. depuis 1990, le nombre de films par année.
4. pour chaque année, afficher le nombre de films pour les années où chacun des films a eu de 1000000 spectateurs.

Exercice 2

On reprend la base de données de l'usine (pièce, opérateur, machine, qualifié_sur). Traduire les requêtes suivantes en SQL :

1. La quantité de pièces fabriquées par chaque employé.
2. La liste des employés, désignés par leur nom, qui ont fabriqué plus de 500 pièces.
3. Pour chaque machine, le nombre d'opérateurs qui y sont qualifiés pour travailler.
4. Pour chaque machine, le nombre de pièces qui y sont fabriquées.
5. Pour chaque employé, le nombre de machines sur lesquelles il est qualifié.
6. La quantité de pièces fabriquées sur chaque machine.

TP 3

Séance 5

Soit une base de données représentée par les tables suivantes :

acteur(idA, nom, prénom, nationalité)

film(idF, titre, année, pays, nbspectateurs, *idRéalisateur*, *idGenre*)

jouer(idActeur, idFilm, salaire)

réalisateur(idR, nom, prénom, nationalité)

genre(idG, description)

Traduire les requêtes suivantes en SQL :

1. les films (désignés par leur titre) dont la fréquentation dépasse la moyenne.
2. les réalisateurs (désignés par leur nom et prénom) dont au moins un film dépasse la fréquentation moyenne.
3. les acteurs (désignés par leur nom et prénom) participant au film ayant eu le plus de succès.
4. les acteurs (désignés par leur nom et prénom) n'ayant joué dans aucun film.
5. les genres jamais utilisés.
6. les réalisateurs (désignés par leur nom et prénom) qui n'ont rien tourné depuis 2002.
7. les genres (désignés par leur description) dont la fréquentation est toujours supérieure à 50000.
8. les genres (désignés par leur description) dont la fréquentation est toujours supérieure à 50000 et dont les réalisateurs sont tous américains.

TP 4

Séance 6

Soit une base de données représentée par les tables suivantes (les clés primaires sont soulignées et les clés étrangères sont en *italique* :

personne(id, nom, prenom)

voiture(immat, marque, annee, prix, *idProp*). Le prix est > 0 .

Exercice 1 (Création de la base)

- Créer les tables ci-dessus avec les contraintes de domaine, de clés primaires et d'intégrité référentielle.
- Insérer les données suivantes :

personne	IdProprio	Nom	Prenom
	Id01	Martin	Paul
	Id02	Duval	

voiture	Immatriculation	Marque	Annee	Prix	IdProprio
	1111AA01	Peugeot	1997	16 000	Id01
	2222BB02	Peugeot	2000	41 200	Id01
	4444DD13	Fiat	1995	30 300	Id02

Exercice 2 (Requêtes de consultation)

Traduire les requêtes suivantes en SQL :

1. Nombre total de personnes.
2. Nombre de personnes dont le prénom est inconnu.
3. Afficher le nombre de voitures par identifiant de propriétaire.
4. Afficher le capital de chaque propriétaire désigné par son nom et son prénom.
5. Afficher le prix moyen par marque.
6. Afficher le nombre de voitures de plus de 20 000 euros pour chaque propriétaire désigné par son nom et prénom.
7. Afficher le capital et l'identifiant des propriétaires dont le capital est supérieur à 50 000 euros.
8. Capital de chaque propriétaire par année (afficher l'identifiant du propriétaire, l'année et le capital).
9. Idem, mais afficher le nom et le prénom à la place de l'identifiant.