

S.G.B.D. 2

TD n° 3

Pratique du langage SQL SQL*Plus ORACLE

Objectif : utiliser iSQLPlus et/ou SQLPlus de Oracle, créer des tables SQL, mettre à jour de données et effectuer les requêtes.

1. Démarrer avec SQL*Plus et iSQLPlus

Démarrer une session iSQLPlus depuis Internet Explorer

- 1.1. Ouvrir une session **ie** et lancer **isqlplus** depuis la page <http://quad.unice.fr:5560/isqlplus/>
- 1.2. Connecter à votre compte avec votre *identifiant* et *mot-de-passe* Oracle

Utilisation de l'interface isqlplus : exemple « Changer votre mot de passe Oracle »

- 1.3. Utiliser la commande **GRANT CONNECT** pour changer votre mot de passe Oracle.
 - a/ Saisir la commande suivante avec votre *identifiant* et votre NOUVEAU *mot de passe*


```
GRANT CONNECT TO identifiant IDENTIFIED BY mot-de-passe
```
 - b/ cliquer sur le bouton « exécuter » pour exécuter la requête.
 - c/ observer l'état d'exécution ou le résultat d'une requête affiché par Oracle
- 1.4. Quitter votre session iSQLPlus par le bouton déconnexion (en haut à droite)

Démarrer une session SQL*Plus, rechanger votre mot de passe

- 1.5. Ouvrir une session SQL*Plus à partir du menu démarrer-applications de votre poste. Connecter à votre compte Oracle avec votre *identifiant* et le *nouveau mot de passe* Oracle.
- 1.6. Utiliser encore une fois la commande **GRANT CONNECT** pour changer votre mot de passe ORACLE à l'ancien mot de passe.

Modifier les paramètres de votre session SQL

- 1.7. Lister la table V\$NLS_LANGUAGE pour connaître les valeurs courantes des paramètres de votre session.

```
SQL> SELECT * FROM V$NLS_PARAMETERS ;
```
- 1.8. Afficher la date système (SYSDATE) avec le jour (Day) et le mois (Month) en toutes lettres.

```
SQL> SELECT TO_CHAR(SYSDATE, 'Day DD Month YYYY') ToDay FROM DUAL ;  
TODAY  
-----  
Friday    17 September 1999
```

- 1.9. Changer la valeur du paramètre NLS_LANGUAGE, remplacer AMERICAN par FRENCH (ou inversement)


```
SQL> ALTER SESSION SET NLS_LANGUAGE = French ;
```
- 1.10. Vérifier la prise en compte du nouveau paramètre langue, en affichant à nouveau la date comme en 1.14


```
SQL> SELECT TO_CHAR(SYSDATE, 'Day DD Month YYYY') ToDay FROM DUAL ;
TODAY
-----
Vendredi 17 Septembre 1999
```
- 1.11. Quitter votre session SQL*Plus (commande **exit**).

2. Créer les tables SQL du schéma VIDEO

Le schéma VIDEO est décrit en Annexe en exécutant le script videobld.sql

Etudier le script videobld.sql

- 2.1. Ouvrir avec un éditeur de texte le fichier videobld.sql (récupérer le fichier depuis le répertoire réseau « supports cours/sgbd2 ») et examiner le texte de création des tables et d'insertion de données.
- 2.2. Sous SQL*Plus ou iSQLPlus, créer la table *film* avec la contrainte de clé primaire : copier l'instruction correspondant dans videobld et copier dans l'interface sqlplus d'oracle.

```
SQL> CREATE TABLE film
(
  id_film          VARCHAR2 (2)          PRIMARY KEY
,
  titre            VARCHAR2 (20)
,
  genre            VARCHAR2 (1)
) ;
```

- 2.3. Vérifier la structure de la table qui vient d'être créer en utilisant la commande **describe** de SQL*Plus.

```
SQL> desc film;
Name                               Null?    Type
-----
ID_FILM                            NOT NULL VARCHAR2(2)
TITRE                               VARCHAR2(20)
GENRE                               VARCHAR2(1)
```

- 2.4. Insérer la ligne ('F1', 'OISEAU BLEU', 'P') dans la table *film*.

```
SQL> INSERT INTO film VALUES ('F1', 'OISEAU BLEU', 'P') ;
```

Etudier la notion de transaction

- 2.5. Donner le droit de consultation sur la table *film* à votre binôme.


```
SQL> GRANT SELECT ON film TO binome;
```
- 2.6. Valider par COMMIT, les mises à jour effectuées depuis l'ouverture de votre session.


```
SQL> COMMIT ;
```
- 2.7. Faites vérifier par votre binôme qu'il peut consulter la table *film* sur laquelle vous lui avez donné le droit de lecture.

```
SQL> SELECT * FROM binome.film;
```

Ajouter 2 lignes supplémentaires dans votre table *film*, puis vérifier que ces lignes sont bien insérées :

```
SQL> SELECT * FROM film;
```

Faites vérifier par votre binôme depuis sa session qu'elles ont bien été insérées :

```
SQL> SELECT * FROM binome.film;
```

- 2.8. Comparer avec le résultat de votre sélection

Note : votre binôme ne verra la mise à jour qu'après que vous ayez fait un **COMMIT**.

Modification de la table film

- 2.9. Modifier la taille du champ titre dans la table film, vérifier.

```
SQL> ALTER TABLE film MODIFY
      ( titre VARCHAR2 (50)
      ) ;
```

- 2.10. Ajouter dans la table film un nouveau champ pour le réalisateur, vérifier.

```
SQL> ALTER TABLE film ADD
      ( realisateur VARCHAR2 (30)
      ) ;
```

Intégrité référentielle :

Création de la table cassette

- 2.11. Créer la table cassette, définir sa clé primaire ainsi que la clé étrangère référençant la clé primaire de la table film.

```
SQL> CREATE TABLE cassette
      (
        id_cassette VARCHAR2 (3) PRIMARY KEY ,
        id_film VARCHAR2 (2) REFERENCES film
      ) ;
```

- 2.12. Ajouter quelques lignes dans la table cassette, vérifier l'intégrité référentielle.

Création de la table emprunt

- 2.13. Essayer de créer la table EMPRUNT par copier/coller l'instruction correspondant sur l'interface SQLPLUS.

```
SQL> CREATE TABLE emprunt
      (
        id_cassette VARCHAR2 (3) PRIMARY KEY REFERENCES cassette,
        id_client VARCHAR2 (4) REFERENCES client,
        date-empt Date
      ) ;
```

Observer et expliquer le résultat de l'exécution.

3. Consulter le DD

- 3.1. Lister la vue **USER_TABLES** pour voir toutes les tables ou vues qui vous appartiennent.
- 3.2. Lister la vue **USER_CONSTRAINTS** pour voir toutes les contraintes sur vos tables.
- 3.3. Lister la vue **TABLE_PRIVILEGES** pour voir toutes les tables ou vues pour lesquelles vous avez des privilèges.

4. Requêtes SQL

Reconstruire la base de données VIDEO en exécutant la totalité du script videobld.sql

- 4.1. Exécuter le script *videobld.sql* pour reconstruire votre base de données.

```
SQL> start videobld (chemin vers le fichier est à préciser)
Ou
SQL> @videobld
```

Rédiger les requêtes, puis les exécuter depuis SQL*Plus ou iSQLPlus

- 4.2. Donner les genres et les titres des films triés par genres.
- 4.3. Donner les différents genres de films.
- 4.4. Donner les noms des clients qui comportent la lettre l en 2ème position.
- 4.5. Donner le nombre de genres différents de film.
- 4.6. Donner les noms des clients dont la date d'abonnement se termine le plus tôt.
- 4.7. Donner les références des cassettes des films policiers (genre=P).
- 4.8. Donner les noms des clients qui ont fait un emprunt avant le premier octobre.
- 4.9. Donner les références des cassettes et les titres des films empruntés au mois de sept.
- 4.10. Donner les genres des films empruntés par Smith ou Jones.
- 4.11. Donner les titres des films qui n'ont pas été empruntés en sept.-01.
- 4.12. Donner les nombres de cassettes pour chaque film.
- 4.13. Donner les nombres de cassettes non empruntées pour chaque film.
- 4.14. Donner les noms des clients qui ont emprunté à la fois un film policier (P) et un film d'aventure (A).
- 4.15. Donner les noms des clients qui n'ont emprunté que des films X.
- 4.16. Donner les noms des clients qui ont emprunté tous les films policiers (P).
- 4.17. Donner les noms des clients qui ont fait un emprunt un même jour que Smith.
- 4.18. Donner les noms des clients qui n'ont emprunté aucun film X ou E.
- 4.19. Donner les noms des clients qui ont emprunté au moins 2 cassettes en sept.-01.
- 4.20. Donner les nombres de cassettes empruntées pour chaque genre de film.

5. Scripts SQL*Plus

- 5.1. Créer un script SQL permettant de lister les titres et les nombres de cassettes des films d'un genre qui est donné en paramètre.
- 5.2. Créer un script SQL permettant de lister les cassettes empruntées par un client dont le nom est donné en paramètre.

Annexe

Le schéma VIDEO

FILM (no_film, titre, genre)

CASSETTE (no_cassette, no_film)

CLIENT (no_client, nom, date_fin_ab, nb_max_empt)

EMPRUNT (no_cassette, no_client, date_empt)

Jeu d'essai

FILM

id_film	titre	genre
F1	OISEAU BLEU	P
F2	BALEINE BLANCHE	A
F3	SOLEIL VERT	F
F4	HERBE ROUGE	E
F5	CHAMBRE JAUNE	P
F6	LAPIN ROSE	X

CASSETTE

id_cassette	id_film
K1	F1
K2	F2
K3	F2
K4	F2
K5	F3
K6	F4
K7	F5
K8	F5
K9	F6
K0	F6

CLIENT

id_client	nom	date_fin_ab	nb_max_empt
C1	Smith	31-DEC-03	5
C2	Jones	30-NOV-03	4
C3	Blake	30-NOV-03	3
C4	Clark	31-DEC-03	2
C5	Adams	31-DEC-03	1

EMPRUNT

id_cassette	id_client	date_empt
K1	C1	20-SEP-03
K2	C1	20-SEP-03
K5	C1	01-OCT-03
K6	C1	20-SEP-03
K8	C1	01-OCT-03
K3	C2	20-SEP-03
K7	C2	01-OCT-03
K9	C3	02-OCT-03

Script videobld.sql

```
rem $ videobld.sql
/* Suppression des tables */
DROP TABLE emprunt ;
DROP TABLE client ;
DROP TABLE cassette ;
DROP TABLE film ;

/* Creation des tables */
CREATE TABLE film
(
    id_film          VARCHAR2 (2)    PRIMARY KEY
    titre            VARCHAR2 (20)
    genre            VARCHAR2 (1)
) ;

CREATE TABLE cassette
(
    id_cassette      VARCHAR2 (3)    PRIMARY KEY
    id_film           VARCHAR2 (2)    REFERENCES film
) ;

CREATE TABLE client
(
    id_client        VARCHAR2 (3)    PRIMARY KEY
    nom               VARCHAR2 (14)   NOT NULL
    date_fin_ab      DATE            NOT NULL
    nb_max_empt      NUMBER (3)      NOT NULL
) ;

CREATE TABLE emprunt
(
    id_cassette      VARCHAR2(3)      PRIMARY KEY REFERENCES cassette
    id_client        VARCHAR2(3)      NOT NULL REFERENCES client
    date_empt        DATE            NOT NULL
) ;

/* Chargement des tables */
INSERT INTO film VALUES ('F1', 'OISEAU BLEU', 'P') ;
INSERT INTO film VALUES ('F2', 'BALEINE BLANCHE', 'A') ;
INSERT INTO film VALUES ('F3', 'SOLEIL VERT', 'F') ;
INSERT INTO film VALUES ('F4', 'HERBE ROUGE', 'E') ;
INSERT INTO film VALUES ('F5', 'CHAMBRE JAUNE', 'P') ;
INSERT INTO film VALUES ('F6', 'LAPIN ROSE', 'X') ;

INSERT INTO cassette VALUES ('K1', 'F1') ;
INSERT INTO cassette VALUES ('K2', 'F2') ;
INSERT INTO cassette VALUES ('K3', 'F2') ;
INSERT INTO cassette VALUES ('K4', 'F2') ;
INSERT INTO cassette VALUES ('K5', 'F3') ;
INSERT INTO cassette VALUES ('K6', 'F4') ;
INSERT INTO cassette VALUES ('K7', 'F5') ;
INSERT INTO cassette VALUES ('K8', 'F5') ;
INSERT INTO cassette VALUES ('K9', 'F6') ;
INSERT INTO cassette VALUES ('K0', 'F6') ;

INSERT INTO client VALUES ('C1', 'Smith', '31-DEC-03', 5) ;
INSERT INTO client VALUES ('C2', 'Jones', '30-NOV-03', 4) ;
INSERT INTO client VALUES ('C3', 'Blake', '30-NOV-03', 3) ;
INSERT INTO client VALUES ('C4', 'Clark', '31-DEC-03', 2) ;
INSERT INTO client VALUES ('C5', 'Adams', '31-DEC-03', 1) ;

INSERT INTO emprunt VALUES ('K1', 'C1', '20-SEP-03') ;
INSERT INTO emprunt VALUES ('K2', 'C1', '20-SEP-03') ;
INSERT INTO emprunt VALUES ('K5', 'C1', '01-OCT-03') ;
INSERT INTO emprunt VALUES ('K6', 'C1', '20-SEP-03') ;
INSERT INTO emprunt VALUES ('K8', 'C1', '01-OCT-03') ;
INSERT INTO emprunt VALUES ('K3', 'C2', '20-SEP-03') ;
INSERT INTO emprunt VALUES ('K7', 'C2', '01-OCT-03') ;
INSERT INTO emprunt VALUES ('K9', 'C3', '02-OCT-03') ;

COMMIT ;
rem EXIT
```