

TP1 : Définition de relations sous Oracle

Création des tables

Les tables sont créées avec les instructions suivantes :

```
CREATE TABLE Clients(idcl number,  
                      nom varchar2(20),  
                      pren varchar2(15),  
                      adr varchar2(30),  
                      tel varchar2(12),  
                      CONSTRAINT clients_pk PRIMARY KEY (idcl));
```

```
CREATE TABLE Livres(refl varchar2(10),  
                     titre varchar2(30),  
                     auteur varchar2(20),  
                     genre varchar2(15),  
                     CONSTRAINT livres_pk PRIMARY KEY (refl));
```

```
CREATE TABLE Achats(idcl number,  
                     refl varchar2(10),  
                     dateachat date check(dateachat between to_date('01-01-2008', 'DD-MM-  
YYYY') and to_date('31-12-2013', 'DD-MM-YYYY')) not null,  
                     CONSTRAINT achats_pk PRIMARY KEY (dateachat, refl, idcl),  
                     CONSTRAINT achats_pf1 FOREIGN KEY (refl) REFERENCES Livres(refl),  
                     CONSTRAINT achats_pf2 FOREIGN KEY (idcl) REFERENCES Clients(idcl));
```

```
CREATE TABLE Avis(idcl number,  
                   refl varchar2(10),  
                   note number(4,2),  
                   commentaire varchar2(50),  
                   CONSTRAINT avis_pk PRIMARY KEY (refl, idcl),  
                   CONSTRAINT avis_pf1 FOREIGN KEY (refl) REFERENCES Livres(refl),  
                   CONSTRAINT avis_pf2 FOREIGN KEY (idcl) REFERENCES Clients(idcl));
```

Les instructions contenant les mots-clefs « Primary key » permettent de déclarer un attribut de la table en clé primaire. Celles contenant les mots-clefs « Foreign Key » permettent de déclarer des clefs étrangères. La ligne qui déclare l'attribut dateachat de la table Achats est la suivante : « *dateachat date check(dateachat between to_date('01-01-2008', 'DD-MM-YYYY') and to_date('31-12-2013', 'DD-MM-YYYY')) not null* ». Elle met en place la contrainte sur la date, énoncée dans le sujet (« Cette date d'achat est comprise entre le 1er janvier 2008 et le 31 décembre 2013. »).

Remplissage des tables

Les tables sont remplies avec les instructions suivantes :

```
INSERT INTO Clients VALUES ('00', 'Dupont', 'Francis', '18 rue du la pendaison', '0614562312');
INSERT INTO Clients VALUES ('01', 'Huster', 'Francis', '20 rue du la pendaison', '0685749645');
INSERT INTO Clients VALUES ('02', 'Colonel', 'Moutarde', '22 rue du la pendaison',
'0678126345');
```

```
INSERT INTO Livres VALUES ('AA', 'Le chat du rabin', 'Joann Sfar', 'BD');
INSERT INTO Livres VALUES ('AB', 'Les Bidochons', 'Christian Binet', 'BD');
INSERT INTO Livres VALUES ('AC', 'Le petit prince', 'Saint-Exupery', 'fantastique');
INSERT INTO Livres VALUES ('AD', 'La mecanique du coeur', 'Mattias Malhzieu', 'fantastique');
```

```
INSERT INTO Achats VALUES ('00', 'AB', TO_DATE('01-01-2011','DD-MM-YYYY'));
INSERT INTO Achats VALUES ('00', 'AD', TO_DATE('01-02-2009','DD-MM-YYYY'));
INSERT INTO Achats VALUES ('01', 'AA', TO_DATE('12-06-2013','DD-MM-YYYY'));
INSERT INTO Achats VALUES ('01', 'AD', TO_DATE('05-07-2009','DD-MM-YYYY'));
INSERT INTO Achats VALUES ('01', 'AC', TO_DATE('05-07-2010','DD-MM-YYYY'));
INSERT INTO Achats VALUES ('02', 'AB', TO_DATE('25-06-2012','DD-MM-YYYY'));
```

```
INSERT INTO Avis VALUES('00', 'AD', '18', 'Super livre');
INSERT INTO Avis VALUES('01', 'AD', '15', 'Chouette');
INSERT INTO Avis VALUES('01', 'AA', '12', 'Bof');
INSERT INTO Avis VALUES('02', 'AB', '12', '');
```

Requêtes SQL

- 1) Les meilleures ventes : les livres (titre, auteur, genre) qui ont été achetés en plus de 10000 exemplaires.

```
SELECT titre, auteur, genre
FROM Livres FULL JOIN Achats ON Livres.refl=Achats.refl
GROUP BY Achats.refl, titre, auteur, genre
HAVING COUNT(Achats.refl)>1;
```

Dans ce cas, la requête est testée avec un nombre d'achat supérieur à 1. On obtient la réponse suivante :

TITRE	AUTEUR	GENRE
La mecanique du coeur	Mattias Malhzieu	fantastique
Les Bidochons	Christian Binet	BD

2) Les livres qui obtiennent une note moyenne supérieure à 16.

```
SELECT titre, SUM(note)/COUNT(Avis.refl)
FROM Livres FULL JOIN Avis ON Livres.refl=Avis.refl
GROUP BY Avis.refl, titre
HAVING SUM(note)/COUNT(Avis.refl)>16;
```

La réponse donnée par Oracle est la suivante :

TITRE	SUM(NOTE)/COUNT(AVIS.REFL)
La mecanique du cœur	16.5

3) Les clients qui n'ont pas renseigné l'attribut commentaire d'un livre (quel client, livre, note).

```
SELECT nom, pren, titre, note
FROM Livres INNER JOIN (Avis FULL JOIN Clients ON Avis.idcl=Clients.idcl) ON
Livres.refl=Avis.refl
WHERE commentaire is NULL;
```

NOM	PREN	TITRE	NOTE
Colonel	Moutarde	Les Bidochons	12