

**DS - Introduction aux Bases de  
Données Relationnelles  
(deuxième session)**

février 2013



---

Seul document autorisé : une feuille A4 manuscrite avec votre résumé du cours.

**Exercice 1 : Requêtes**

Dans le premier exercice, vous travaillerez avec la base d'un club de bateau à voile. Le club possède plusieurs bateaux, que ces adhérents peuvent utiliser pour faire des sorties d'une journée. Le club gère les *réservations* des *bateaux* par les *skippers* dans une base, avec le schéma suivant :

```
skippers(sid :integer,snom :varchar(30),niveau :integer, age :real)
bateaux(bid :integer,bnom :varchar(30),couleur :varchar(10))
reservations(sid :integer,bid :integer,jour :date)
```

Pour les bateaux, on connaît le nom et la couleur. Pour les skippers (les adhérents du club), on connaît le nom, le niveau, et l'âge.

Question 1.1 : Donnez quatre requêtes, avec quatre techniques SQL différentes, pour trouver les noms des skippers ayant réservé au moins un bateau.

Question 1.2 : En SQL, trouvez pour chaque niveau, l'âge du skipper le plus jeune de ce niveau.

Question 1.3 : En SQL, trouvez l'âge moyen des skippers ayant réservé le bateau 101 plus de cinq fois.

Question 1.4 : Considérez la question suivante : *Trouvez les noms des skippers avec un niveau supérieur à celui de tous les skippers d'âge  $\leq 21$ .* Les deux requêtes SQL suivantes sont des tentatives d'obtenir la réponse à cette question. Calculent-elles les deux le même résultat ? Si non, ou si oui, expliquez pourquoi. Sous quelles conditions calculent-elles le même résultat ?

Requête A :

```
SELECT S.snom FROM skippers S
WHERE NOT EXISTS ( SELECT *
                   FROM skippers S2
                   WHERE S2.age < 21
                   AND S.niveau <= S2.niveau )
```

Requête B :

```
SELECT S.snom
FROM skippers S
WHERE S.niveau > SOME (
  SELECT S2.niveau
  FROM skippers S2
  WHERE S2.age < 21 )
```

Question 1.5 : En TRC, trouvez les noms des skippers ayant réservé le bateau 103.

## **Exercice 2 : Modélisation**

Construire le diagramme conceptuel suivant en UML et le transformer en diagramme logique.

On souhaite gérer une salle de cinéma. Chaque salle présente une programmation hebdomadaire des films projetés et possède un numéro. Un film est caractérisé par un réalisateur, un titre, les horaires de début de la séance et la date et si le film est projeté en VO ou VF. Chaque film possède la liste des acteurs qui ont participé, chaque acteur étant caractérisé par son nom, prénom et age. Un film peut passer dans plusieurs salles. Un acteur peut participer à plusieurs films.

Réalisez les actions suivantes :

**Question 2.1 :** Construire le diagramme conceptuel. Identifiez les classes et les associations entre les classes. Ajoutez des cardinalités aux associations. Veillez à utiliser la notation UML pour les cardinalités.

**Question 2.2 :** A partir du diagramme conceptuel, appliquer l'algorithme de transformation du modèle conceptuel en modèle logique, et créer le modèle logique correspondant.

**Question 2.3 :** Écrivez en SQL le script de création des relations que vous avez obtenues dans l'étape précédente.

**Question 2.4 :** Ajoutez les contraintes aux relations (clé primaire, intégrité référentielle, nullité, unicité).

**Question 2.5 :** Proposez des commandes d'insertion qui permettront de tester les contraintes imposées.

## **Exercice 3 : Questions de connaissance et compréhension**

**Question 3.1 :** Discutez les façons d'optimiser des expressions de l'algèbre relationnelle, dans le cas où elles contiennent une projection d'une jointure entre deux tables A et B.

**Question 3.2 :** Expliquez avec un exemple l'élimination de l'implication dans une expression en TRC.

**Question 3.3 :** Dire qu'est-ce que c'est un attribut dans un document XML et faire un exemple.

**Question 3.4 :** Dire qu'est-ce que c'est un document XML valide.

**Question 3.5 :** Écrivez une requête en utilisant le langage de requête XPath pour récupérer les lieux des concerts d'un groupe musical nommé "Radiohead". D'abord, créez un document qui contient les éléments ci-dessus et après écrire la requête qui doit y être évaluée.

**Question 3.6 :** L'expression XPath suivante doit être évaluée :

`teacher[name="Mary"][cours="COO"]`

Expliquer sa sémantique sur un arbre ayant comme racine un noeud r. Dire aussi s'il s'agit d'une expression XPath absolue ou relative.