



Durée: 2h. **Barème:** I (3 points), II (7 points), III (10 points).
Supports de cours (sur papier uniquement) autorisés.

Exercice I: Dépendances fonctionnelles et normalisation

Les questions suivantes sont indépendantes.

- Soit A, B, C et D des ensembles d'attributs.
Est-ce que les propositions suivantes sont vraies ? Si oui, le démontrer à l'aide des axiomes d'Armstrong, sinon donner un contre-exemple (en illustrant à l'aide de relations par exemple).
 - Si $A \rightarrow B$ et $C \rightarrow D$, alors $AC \rightarrow BD$.
 - Si $AB \rightarrow C$ et $B \rightarrow D$, alors $AD \rightarrow C$.
- Soit R une relation dont le schéma est composé des quatre attributs A, B, C et D .
Considérons les dépendances fonctionnelles suivantes : $ABC \rightarrow D$ et $D \rightarrow A$.
 - Lister toutes les clés possibles pour R .
 - Est-ce que R est en 3NF? Justifier.
 - Est-ce que R est en BCNF? Justifier.

Exercice II: Conception d'une base de données relationnelle

Un groupe d'agences immobilières vous demande de mettre en place une partie de sa base de données. Dans ce contexte, on vous communique les informations suivantes :

- Chaque agence est caractérisée par un code unique et une adresse, et est responsable de la gestion d'un ensemble d'appartements.
- Un appartement est caractérisé par sa surface, le nombre de pièces et son adresse.
- Un propriétaire et un locataire sont caractérisés par un nom, un numéro de téléphone et un email.
- Une location est caractérisée par une date de début et une date de fin.
- Pour des raisons de simplicité, on considère que le propriétaire d'un appartement ne change pas au cours du temps.

On souhaite représenter les données reliées aux agences, appartements, propriétaires et locataires ainsi qu'aux relations entre eux dans une base de données relationnelle.

Écrire l'ensemble des relations en BCNF, en précisant les dépendances fonctionnelles et les clés primaires. Il est recommandé de clairement énoncer les hypothèses posées dans le cadre de la conception de la base, comme celles liées à la cardinalité par exemple.

Exercice III: SQL

La médiathèque de Paris met à disposition de ses membres inscrits un ensemble de films sous forme de DVDs à louer. La base de données correspondante a le schéma suivant (avec la clé primaire soulignée) :

prod **Films** (Film ID, Titre, Réalisateur, Année_sortie, Durée)
estadele **Membres** (Membre ID, Nom, Email, Région, Ancienneté)
hell **Locations** (Membre ID, Film ID, Date début, Date fin)

Notes :

- L'attribut *Durée* indique la durée du film en minutes.
- L'attribut *Ancienneté* indique le nombre d'années depuis l'inscription d'un membre auprès de la médiathèque.
- L'attribut *Date_début* indique la date à laquelle le film a été loué et l'attribut *Date_fin* la date à laquelle le film a été rendu.

Exprimer les requêtes suivantes en SQL :

1. Lister les réalisateurs des films sortis en 2010 et ayant été loués par des membres de la région « 13^{ème} arrondissement ».
2. Lister les noms des membres qui ont loués tous les films fournis par la médiathèque.
3. Lister les IDs des films les plus longs (films qui ont la plus longue durée).
4. Lister les titres des films qui ont uniquement été loués par des membres qui ont plus de deux ans d'ancienneté.
5. Pour chaque membre ayant loué plus de quatre films, lister l'ID de ce membre ainsi que la date la plus récente à laquelle il a effectué une location.
6. Lister les noms des membres qui ont loués un film d'une durée supérieure à 120 minutes mais qui n'ont jamais loué un film du réalisateur « Quentin Tarantino ».
7. Lister les noms des membres qui n'ont loué aucun film et qui ont une ancienneté supérieure à l'ancienneté moyenne de tous les membres.
8. Lister les noms des membres qui ont loué un même film moins d'une semaine après l'avoir rendu.