

# Partiel Bases de Données

2024

**Durée : 2h Une feuille A4 manuscrite avec des notes de cours/TD/TP autorisée, calculatrice erdite**

## Exercice 1

Dans cet exercice, on désire modéliser de façon très simplifiée le système d'une base de données pour gérer le personnel et leur travail dans une piscine.

Une piscine d'une ville est composée de un ou plusieurs bassins. Chaque bassin peut être connecté à un ou plusieurs autres bassins (accès, exemple piscine de Belfort, le bassin extérieur de 50 m est connecté au bassin intérieur de 25m par des portes fenêtres). Les bassins sont d'un certain type (25m). Les surveillants donnent des cours ou surveillent des bassins. C'est la ville qui emploie chaque surveillant.

*Les clés primaires sont soulignées et les clés étrangères sont précédées par un caractère "#".*

*La convention de nommage (typographique) utilisée est le **snake\_case** , il y a un « \_ » entre les mots dans les noms des attributs (colonnes) d'une table et certains noms de table.*

Soit le modèle relationnel suivant :

piscine = (id\_piscine , nom , surface , estPublique , #id\_ville);  
surveillant = (id\_surveillant , nom\_surveillant , prenom );  
ville = (id\_ville , nom\_ville );  
connexion = (#id\_bassin\_1, #id\_bassin\_2, type\_connexion );  
type\_bassin = (id\_type\_bassin , libelle );  
bassin = (id\_bassin , longueur , largeur , #id\_type\_bassin, #id\_piscine);  
emplois = (#id\_ville, #id\_surveillant, date\_debut , date\_fin , salaire );  
cours = (#id\_piscine, #id\_surveillant, date\_debut , nom\_cours , duree );  
surveillance = (#id\_piscine, #id\_surveillant, date\_debut , date\_fin , horaires );

\* **id\_bassin\_1** est un entier qui identifie un bassin.

\* **connexion** définit les bassins connectés et le type de connexion (couloir intérieur, qui permet à un usager (client) de passer d'un bassin à l'autre).

\* **duree** est un numérique qui représente **le temps en minute d'un cours.**

### Question 1

Proposer un modèle conceptuel des données associé au modèle relationnel précédent : modèle relationnel complet. **Utiliser le document réponse.**

Ne pas utiliser d'identifiant relatif sauf si il n'y a pas d'autre solution dans une relation binaire.

### Question 2

\* Proposer le script de création de la table « **cours** » dans un SGBD respectant strictement la norme SQL avec des contraintes d'**intégrité référentielle** sur les autres Tables.

\* Quelles hypothèses préalables doivent être vérifiées avant de lancer l'exécution de ce script ? (Quelles sont les tables à créer **avant au minimum, indiquer l'ordre de création** )

\* Indiquer quelles sont les tables qu'il n'est pas nécessaire de créer avant cette table

### Question 3

Écrire les requêtes SQL suivantes (bien numéroter vos requêtes) :

ATTENTION :

- faire des **sauts de ligne** au minimum avant les mots clés « SELECT », « FROM », « INNER JOIN », « WHERE », « GROUP BY », « ORDER BY » (sinon pénalité)
- **souligner** les mots clés ci-dessus (sinon pénalité)
- écrire le code complet de chaque requête (sinon pénalité importante)
- utiliser uniquement les tables nécessaires
- ne pas utiliser de jointure dite « naturelle » (« natural join »)
- ne **pas** utiliser le mot clé « **limit** »

1. Donner (afficher) le nombre de surveillants qui ont donné un cours durant l'année 2023. (*affichage : un numérique uniquement*). *Utiliser le minimum de table.*
2. Donner le nombre de surveillants employés le 21 décembre 2023 dans la ville de "Belfort". (*affichage : un numérique uniquement*). *Faire attention, si la colonne « date\_fin » n'est pas renseignée, le surveillant travaille toujours sur cet emploi.*
3. Afficher la longueur maximale (des bassins) par piscine (*affichage : un numérique qui correspond à la longueur, le nom de la ville, le nom de la piscine ; trier le résultat par ville croissant puis par nom de piscine croissant*).
4. Donner le nombre de piscines dans chaque ville pour les villes où il y a plus de 2 piscines. (*affichage : nom des villes, nombre de piscines, le résultat sera trié selon l'ordre lexicographique inverse sur le nom des villes*).

5. Donner le nom des surveillants employés à belfort avant ou en 2023 qui n'ont pas donné un cours en 2023. ( affichage : nom du (des) surveillant(s), le résultat sera trié selon l'ordre lexicographique (alphabétique) inverse ).
6. Donner le nom du (des) surveillant(s) de belfort qui gagne(nt) plus que la moyenne des salaires des surveillants de montbéliard. ( affichage : nom du (des) surveillant(s) et leur(s) salaire(s), le résultat sera trié selon l'ordre lexicographique inverse du nom des surveillants). Pour ce calcul, les surveillants sont tous encore en activité si leur date de fin de contrat n'est pas renseignée (CDI) ou si cette date n'est pas encore passée.
7. Donner le nom des piscines où il y a eu plus de 10 heures de cours au mois d'octobre 2023 et qui n'ont donné aucun cours en aout 2023. La durée des cours est en minutes dans la base de données. Il est demandé d'essayer d'utiliser le mot clé « EXISTS » pour la sous requête. ( affichage : nom de la (des) piscines(s) et le nombre d'heure(s) de cours)
8. Donner le nom du surveillant de belfort qui a donné le plus grand nombre de cours à la piscine de Belfort. ( affichage : nom du surveillant et le nombre de cours)

#### Question 4 : modification du MCD

On désire modifier dans le MCD,

Proposer une solution qui fait apparaître les séances de cours. Un cours peut être composé de au moins une ou plusieurs séances. Les séances ont lieu à une date avec une heure de début et une durée. Un cours est défini par un nom, un nombre de séance(s) maxi, une date de début et de fin. Un cours est d'un certain type (« collectif », « individuel »). Un cours possède un prix et a un surveillant comme responsable. Une séance a lieu dans une piscine et est encadrée par un et un seul surveillant.

**1. Utiliser le deuxième document réponse (verso).** Dessiner un nouveau schéma MCD avec les entités/associations et cardinalités utiles. Dans les entités, vous pouvez faire apparaître uniquement les clés et des petits points pour indiquer qu'il y a d'autres attributs si ça ne nuit pas à la compréhension.

Si vous supprimez des attributs, faites les apparaître et rayez les.

Tenir compte des informations ci-dessous.

**2. Donner les modifications dans le Modèle Logique de Données initial (page1) :** donner le nom des tables à créer en plus ainsi que les attributs en indiquant bien avec les mêmes règles :

\* Les clés primaires sont soulignées et les clés étrangères sont précédées par un caractère "#".

\* La convention de nommage (typographique) utilisée est le **snake\_case** , il y a un « \_ » entre les mots dans les noms des attributs (colonnes) d'une table et certains noms de table.

## Exercice 2

Le contenu de la base représente des clients achetant des places sur des vols. Son schéma est constitué de 4 tables :

Client(id\_client, nom, miles)

Vol (id\_vol, intitulé, capacité, prix\_place, nb\_places\_total, nb\_places\_disponibles, prix\_unitaire\_place, distance)

Reservation(id\_reservation, id\_vol, id\_client, prix\_total, nb\_places)

Log\_reservation (id\_log, id\_vol, id\_client, nb\_places, libelle\_operation, date\_log)

Remarque :

\* distance est en « miles »

\* nb est l'abréviation de « nombre de »

### Exercice 2.1

Écrire une procédure PL/SQL qui reçoit en paramètres l'identifiant du client, l'identifiant du vol et le nombre de places à réserver ; Cette procédure réalise les opérations suivantes :

\* cette procédure vérifie que le nombre de places disponibles sur le vol est supérieur ou égal au nombre de place à réserver ; si ce n'est pas le cas une exception est levée et la procédure se termine

\* Sinon créer une transaction et dans cette transaction, effectuer les opérations suivantes :

\* le nombre de places disponibles est décrémenté du nombre de places à réserver

\* une réservation est créée avec l'identifiant du vol, l'identifiant du client et le prix total de la réservation ( prix unitaire d'une place dans un vol fois le nombre de place à réserver )

\* le nombre de « miles » du client est crédité de la distance du vol multiplié par le nombre de place réservé

### Exercice 2.2

Créer un trigger : après l'opération de créer une réservation dans la base de données, le trigger (déclencheur) sauvegarde dans une table de « log » : « Log\_reservation » les informations de la réservation. Le libellé de l'opération a pour contenu « achat de place(s) de M. xxxx sur le vol yyyy à la date zzzz ».

Remarque : la difficulté réside surtout à bien écrire le libellé de l'opération car « xxxx » est le nom du client, « yyyy » est l'intitulé du vol.