

TD n° 4 : bases de données et requêtes SQL

Exercice 1. Clé primaire

Un institut suit quotidiennement la propagation d'une pandémie au cours d'une année non bissextile. Pour cela il dispose d'une table constituée de trois colonnes :

- `pays` (TEXT) : le nom du pays ;
- `jour` (INT) : le numéro du jour de l'année (de 1 à 365) ;
- `cas` (INT) : le nombre de cas confirmés.

Quel(s) attribut(s) peu(ven)t constituer une clé primaire ?

Exercice 2. Clé primaire, clé étrangère

Un CDI gère son fond à l'aide d'une base de données. Celle-ci est constituée de deux tables :

- `Livres` ayant pour attributs `id`, `titre`, `auteur`, `id_élève`,
- `Elèves` ayant pour attributs `id`, `nom`, `prénom`, `classe`.

On suppose qu'il n'y a pas de livre en double et qu'un élève peut emprunter aucun, un ou plusieurs livres.

1. Pour chaque table, donner la clé primaire et les éventuelles clés étrangères.
2. Peut-on enregistrer deux élèves ayant le même nom et le même prénom dans la table `Elèves` ?
3. Comment pourrait-on modifier cette base de données pour faciliter le suivi des emprunts ?

Exercice 3. Premières requêtes au CDI

On considère de nouveau la base de données de l'exercice 2. On précise que les attributs `id` et `id_élève` sont de type entier et tous les autres de type chaîne de caractères.

1. Écrire une requête permettant d'obtenir tous les enregistrements de la table `Livres`.
2. Écrire une requête permettant d'obtenir les titres des livres écrits par Spinoza.
3. Écrire une requête permettant d'obtenir tous les auteurs (différents) dont au moins un ouvrage est au CDI.
4. Écrire une requête permettant d'obtenir les nom et prénom de tous les élèves de la classe TSI2, classés par ordre alphabétique.

Exercice 4. Météo : les bases

On dispose d'une base de données météorologiques pour l'année 2024 constituée d'une seule table `Météo` ayant pour attributs `ville` (TEXT), `mois` (INT), `jour` (INT), `Tmin` (FLOAT), `Tmax` (FLOAT), `précipitations` (FLOAT). On suppose qu'il n'y a qu'un enregistrement par ville et par jour.

1. Écrire une requête permettant d'obtenir les températures minimales et maximales à Troyes le 1^{er} septembre.
2. Écrire une requête permettant d'obtenir les jours où il a plu à Troyes.
3. Écrire une requête permettant d'obtenir l'amplitude de température à Troyes chaque jour d'octobre.
4. Écrire une requête permettant d'obtenir le nom des villes et les jours où il a fait plus de 40 °C. On classera les résultats par température maximale décroissante.
5. Que renvoie la requête suivante ?

```
1 SELECT ville, jour, mois
2 FROM Météo
3 WHERE Tmax <= 0 AND précipitations = 0
```

↪ Faire les épreuves 1.06 (*Retrouver une note*) et 3.10 (*Qui a majoré le DS ?*) du challenge info.

Exercice 5. Retour au CDI : fonctions d'agrégation

On considère de nouveau la base de données de l'exercice 2.

1. Écrire une requête permettant d'obtenir le nombre de livres dont dispose le CDI.
2. Écrire une requête permettant d'obtenir les auteurs ainsi que leur nombre de livres pour chacun d'eux.
3. Écrire une requête permettant d'obtenir les auteurs dont au moins cinq livres différents sont au CDI.

Exercice 6. Météo : fonctions d'agrégation

On reprend la base de données de l'exercice 4.

1. Écrire une requête permettant d'obtenir la quantité de pluie tombée à Troyes au cours de l'année.
2. Écrire une requête permettant d'obtenir la température maximale moyenne à Troyes chaque mois.
3. Écrire une requête permettant d'obtenir les mois où il a gelé à Troyes.
4. Écrire une requête permettant d'obtenir pour chaque ville le nombre de jours où il a fait plus de 40 °C.
5. Modifier la requête précédente pour ne garder que les dix villes qui ont connu le plus de jours atteignant ou dépassant 40 °C.
6. Que renvoie la requête suivante ?

```
1 SELECT mois, MIN(Tmin), MAX(Tmax), SUM(précipitations)
2 FROM Météo
3 WHERE ville = 'Chaumont'
4 GROUP BY mois
```

↪ Faire l'épreuve 2.06 (*Calcul moyenne*) du challenge info.

Exercice 7. Premières jointures au CDI

On reprend la base de données de l'exercice 2.

1. Écrire une requête permettant d'obtenir le nom et le prénom des élèves ayant emprunté « Lorenzaccio ».
2. Écrire une requête permettant d'obtenir les classes des élèves ayant emprunté un livre écrit par Edith Wharton.
3. Écrire une requête permettant d'obtenir le nombre de livres différents empruntés par des élèves de TSI2.

Exercice 8. Plein de jointures avec des films

Comme son nom l'indique, l'Internet Movie Database, abrégé en IMDb, est une base de données¹ sur les productions cinématographiques mondiales. Une version très simplifiée de celle-ci comprend trois tables :

— **Actors** ayant pour attributs :

- **id** : identifiant, type entier, clé primaire,
- **first_name** : prénom, type texte,
- **last_name** : nom, type texte.

— **Movies** ayant pour attributs :

- **id** : identifiant, type entier, clé primaire,
- **title** : titre du film, type texte,
- **year** : année de sortie du film, type entier.

— **Casts** ayant pour attributs :

- **id_actor** : identifiant d'un acteur, type entier,
- **id_movie** : identifiant du film, type entier,
- **role** : nom du personnage, type texte.

1. Une partie est téléchargeable gratuitement sur <https://www.imdb.com/interfaces/>.

1. Écrire une requête permettant d'obtenir le nom des personnages du film *Imitation Game*.
2. Écrire une requête permettant d'obtenir le nom de tous les personnages joués par Benedict Cumberbatch.
3. Écrire une requête permettant d'obtenir le nom des films dans lesquels a joué Benedict Cumberbatch ainsi que le nom de son personnage pour chacun.
4. Écrire une requête permettant d'obtenir le nombre de films dans lesquels a joué Taraji Henson.
5. Écrire une requête permettant d'obtenir, pour chaque année, le nombre de films dans lesquels a joué Taraji Henson.
6. Écrire une requête permettant d'obtenir les identifiants des acteurs n'ayant joué dans des films qu'avant 1900.

↪ Faire les épreuves 3.15 (*Combien de bonnes notes ?*) et 4.06 (*Moyenne coefficientée*) du challenge info.

Exercice 9. Deux CDI et opérateurs ensemblistes

Les lycées les Lombards et Chrestien de Troyes veulent comparer les livres de leurs CDI respectifs. Pour cela on dispose de deux tables :

- **Lombards** (id, titre, auteur, année, nb_pages),
- **Chrestien** (id, titre, auteur, année, nb_pages).

1. Écrire une requête permettant d'obtenir tous les livres présents dans au moins un des lycées.
2. Écrire une requête permettant d'obtenir les livres de Spinoza présents dans les deux lycées. On triera les résultats par ordre alphabétique du titre.
3. Écrire une requête permettant d'obtenir les livres de plus de 500 pages présents aux Lombards mais pas à Chrestien.
4. Écrire une requête permettant d'obtenir les titres et les auteurs des livres pour lesquelles les Lombards disposent d'une édition plus récente que Chrestien.

Exercice 10. Des requêtes imbriquées et des films

On reprend la base de données décrite dans l'exercice 8.

1. Répondre de nouveau aux questions 1 et 2 de l'exercice 8 mais en utilisant des requêtes imbriquées au lieu de jointures.
2. Écrire une requête permettant d'obtenir le ou les titres des films de la première année de l'histoire du cinéma.
3. ★ Écrire une requête permettant d'obtenir l'année de sortie et le titre des films dans lesquels jouent Joaquin Phoenix et Scarlett Johansson. *On pourra procéder à une autojointure des tables Casts et Actors.*
4. ★ Que renvoie la requête suivante :

```

1 SELECT first_name, last_name, title
2 FROM Actors AS a
3 JOIN Casts AS c ON a.id = c.id_actor
4 JOIN Movies AS m ON c.id_movie = m.id
5 WHERE year >= 1950
6 GROUP BY id_actor, id_movie
7 HAVING COUNT(role) > 1

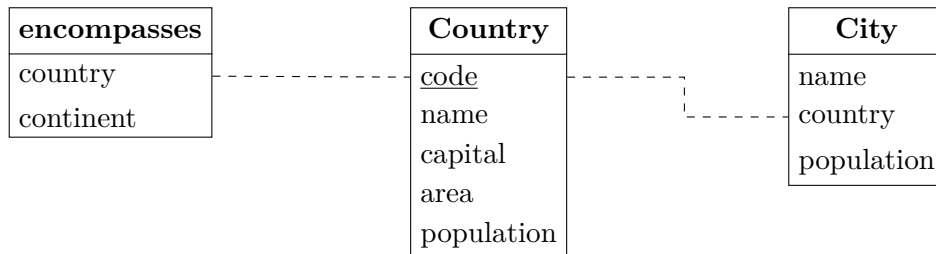
```

Exercice 11. Requêtes illimitées dans la Mondial Database

La Mondial database regroupe de (très) nombreuses informations sur les pays du monde, leur géographie, leur population, leur économie, etc. La dernière version comporte 37 tables dont les noms et les attributs sont listés ici : <http://www.dbis.informatik.uni-goettingen.de/Mondial/mondial-RS.pdf> ou de façon plus visuelle : <http://www.dbis.informatik.uni-goettingen.de/Mondial/mondial-abh.pdf>.

On peut l'interroger en ligne² grâce à un site de l'Université de Göttingen (Allemagne) : <https://www.semwebtech.org/sqlfrontend/>

Dans les question ci-dessous, on s'intéresse seulement à une partie des tables **Country**, **encompasses** et **City** comme décrit dans ce schéma :



Notons que l'attribut **country** des tables **encompasses** et **City** correspond au code du pays (et non pas son nom), et donc à la clé primaire de la table **Country**.

1. Écrire une requête permettant d'obtenir le nom et la capitale de tous les pays.
 2. Écrire une requête permettant d'obtenir la population et la surface du Canada.
 3. Écrire une requête permettant d'obtenir la population mondiale et le nombre de pays.
 4. Écrire une requête permettant d'obtenir le nom et la densité (= population / surface) de chaque pays. Classer les résultats par ordre décroissant de densité.
 5. Écrire une requête permettant d'obtenir la liste des pays d'Asie.
 6. Écrire une requête permettant d'obtenir la population et le nombre de pays de chaque continent.
 7. Écrire une requête permettant d'obtenir le pays le moins peuplé du monde.
 8. Écrire une requête permettant d'obtenir les pays d'Afrique d'au moins cinquante millions d'habitants. Classer les résultats par ordre alphabétique.
 9. Écrire une requête permettant d'obtenir le nombre de villes en Chine comptant plus d'un million d'habitants.
 10. Écrire une requête permettant d'obtenir les pays du monde dont la population est supérieure à celle de la France.
 11. Écrire une requête permettant d'obtenir les noms des villes qui sont aussi des noms de pays.
 12. Écrire une requête permettant d'obtenir la capitale et la population de celle-ci pour chaque pays.
 13. Écrire une requête permettant d'obtenir le nom et la population de la ville la plus peuplée d'Europe.
- À vous de continuer sur le site pour répondre à toutes les questions que vous vous posez (ou pas).

2. Attention, il y a quelques subtilités dans l'écriture des requêtes. Par exemple on ne peut pas utiliser la clause **LIMIT** et les renommages doivent se faire sans la clause **AS** : on écrira **FROM Country c** et non **FROM Country AS c**.