

DS 4

Informatique pour tous, première année

Julien REICHERT

Seules les réponses à la dernière question à choix multiples doivent être justifiées.

Exercice 1 : QCM.

1. Dans une requête simple utilisant les six mots-clés ci-dessous, lequel vient en dernier ?
a) FROM b) WHERE c) HAVING d) GROUP BY e) JOIN f) SELECT
2. ... et lequel vient en troisième ?
a) FROM b) WHERE c) HAVING d) GROUP BY e) JOIN f) SELECT
3. Laquelle de ces propositions ne désigne pas une fonction d'agrégation implémentée en SQL partout (ou du moins presque) ?
a) MAX b) MIN c) AVG d) SUM e) PROD f) COUNT
4. Que peut-on toujours dire d'une clé primaire ?
a) Elle ne concerne qu'un attribut.
b) N'importe quelle clé d'une table peut être sa clé primaire.
c) Elle accélère la recherche dans une table.
d) Elle est incrémentée automatiquement quand on fait une insertion sans préciser sa valeur.
e) Elle utilise le type INT.
f) Une table doit forcément en avoir une.

Exercice 2 : On considère une table `Resultats` pouvant matérialiser des épreuves quelconques. Les attributs sont limités à une date `Date` de type chaîne de caractères de taille limitée à 20, un identifiant `Id` et à un score `Score`, tous les deux de type entier. Le couple `(Date, Id)` est une clé.

Pour chacun des exercices, si on demande de récupérer des informations, il n'y a pas de pénalité au cas où des informations supplémentaires sont collectées, sous réserve de ne pas provoquer d'ambiguïté voire d'erreur SQL.

On utilisera exclusivement une version allégée de SQL (malgré une certaine tolérance pour MySQL).

Question 2.1 : Écrire une requête qui permet de récupérer le score réalisé par la personne d'identifiant `n` à la date `d`.

Question 2.2 : Écrire une requête qui permet de récupérer le score maximum réalisé par la personne d'identifiant `n`.

Question 2.3 : Écrire une requête qui permet de déterminer le nombre de personnes ayant participé à la date `d`.

Question 2.4 : Écrire une requête qui permet de déterminer la date à laquelle le nombre de personnes ayant participé était le plus grand ainsi que ce nombre. En cas d'égalité, on peut indifféremment retourner un seul enregistrement ou tous.

Exercice 3 : Nous allons faire le lien entre la théorie des ensembles et les bases de données.

On considère un ensemble fini D et une table T avec un attribut noté Elt dont les enregistrements parcourent l'ensemble D . On considère aussi une relation R entre éléments de D , qu'on représentera par une fonction $f : D \times D \rightarrow \{0, 1\}$ telle que $f(x, y) = 1$ ssi xRy . La fonction f est l'indicatrice de la relation, et on suppose qu'elle est implémentée en SQL de sorte qu'un appel à $f(x, y)$ renvoie 1 ou 0 suivant le cas, dès lors que x et y seront des valeurs pour l'attribut Elt .

Question 3.1 : Écrire une requête dont le résultat est la table dérivée décrivant la relation R .

On suppose maintenant que la table qu'on souhaitait obtenir existe déjà et s'appelle TT . Comme la relation n'est pas forcément symétrique, l'ordre des attributs est important, et on les appellera Elt1 et Elt2 , tous les deux de domaine D .

Question 3.2 : Écrire une requête qui permet de récupérer le nombre d'éléments y de D tels que xRy , à x fixé et utilisable dans la requête.

Question 3.3 : Écrire une requête qui permet de récupérer le nombre d'éléments y de D tels que yRy .

Question 3.4 : Écrire une requête qui permet de récupérer le nombre d'éléments y de D tels que xRy ou yRx , à x fixé et utilisable dans la requête. Attention, on demande bien $\text{Card}(\{y \in D \mid xRy \vee yRx\})$.