Pour vérifier vos requêtes, utilisez les bases de données associées.

1 CCINP 2019

une base de données médicale contient des informations administratives sur les patients et des informations médicales. On considère deux tables : PATIENT et MEDICAL.

La table PATIENT contient les attributs suivants :

- id : identifiant d'un individu (entier), clé primaire ;
- nom : nom du patient (chaîne de caractères) ;
- prenom : prénom du patient (chaîne de caractères) ;
- adresse : adresse du patient (chaîne de caractères) ;
- email (chaîne de caractères) ;
- naissance : année de naissance (entier).

La table MEDICAL contient les attributs suivants :

- id : identifiant d'un ensemble de propriétés médicales (entier), clé primaire ;
- data1 : donnée (flottant) ;
- data2 : donnée (flottant) ;
- ... ;
- idpatient : identifiant du patient représenté par l'attribut id de la table PATIENT (entier) ;
- etat : description de l'état du patient (chaîne de caractères).

Les attributs data1, data2 ...sont des données relatives à l'analyse médicale souhaitée. L'attribut etat permet d'affecter un label à un ensemble de données médicales : « normal », « hernie discale », « spondylolisthésis ».

Exercice 1 *CCP 2019*

Les questions du concours :

1. Écrire une requête SQL permettant d'extraire les identifiants des patients ayant une « hernie discale ».
2. Écrire une requête SQL permettant d'extraire les noms et prénoms des patients atteints de « spondylolisthésis ».
3. Écrire une requête SQL permettant d'extraire chaque état et le nombre de patients pour chaque état.

2 CCP 2018

Le système Hawk-Eye permet d'enregistrer en temps réel les trajectoires des balles lors d'un tournoi de tennis. Ces données sont stockées dans une base de données constituée de deux tables.

La table MATCHS contient les différents matchs du tournoi avec les attributs :

- id : identifiant de type entier, clé primaire ;
- nom : nom du match de type texte ;
- numero : numéro du match dans la planification du tournoi ;
- date : date où le match s'est déroulé ;
- joueur1 : nom du premier joueur ;
- joueur2 : nom du deuxième joueur (les joueurs 1 et 2 sont rangés par ordre alphabétique) :
- autres attributs non détaillés...

La table POINTS contient des entités correspondant aux points joués lors d'un match. Elle contient les attributs :

- id : identifiant de type entier, clé primaire ;
- mid : identifiant du match correspondant à la définition de type entier ;
- nombre : nombre d'échanges de type entier ;
- fichier : nom du fichier image de la trajectoire (stockée) correspondant au point ;
- autres attributs non détaillés...

Exercice 2 *CCP 2018*

Les questions du concours :

1. Rappeler la définition et l'intérêt d'une clé primaire.
2. Écrire une requête SQL permettant d'afficher les identifiants des matchs joués par Federer, joueur pris pour exemple.
3. Écrire une requête SQL permettant d'afficher le nombre d'échanges maximum lors du match dont l'identifiant est mid=4.
4. Écrire une requête SQL permettant de récupérer le nom des fichiers de toutes les images qui correspondent au match dont le nom est « Federer-Djokovic ».

Indications sur les résultats :

1. Voir le cours.
2. On trouve 6, 11 et 14
3. On trouve 30.
4. On trouve 12 vidéo.

3 Mines-Ponts 2017

On modélise un réseau routier par un ensemble de *croisements* et de *voies* reliant ces croisements. Les voies partent d'un croisement et arrivent à un autre croisement. Ainsi, pour modéliser une route à double sens, on utilise deux voies circulant en sens opposés.

La base de données du réseau routier est constituée des relations suivantes :

- Croisement (`id`,`longitude`,`latitude`)
- Voie (`id`,`longueur`,`id_croisement_debut`,`id_croisement_fin`)

Exercice 3 *MP 2019*

Les questions du concours :
Dans la suite on considère c l'identifiant (id) d'un croisement donné.

1. Écrire la requête SQL qui renvoie les identifiants des croisements atteignables en utilisant une seule voie à partir du croisement ayant l'identifiant c.
2. Écrire la requête SQL qui renvoie les longitudes et latitudes des croisements atteignables en utilisant une seule voie, à partir du croisement c.
3. Que renvoie la requête SQL suivante ?

```
SELECT V2.id_croisement_fin
FROM   Voie as V1
JOIN   Voie as V2
ON  V1.id_croisement_fin = V2.id_croisement_debut
WHERE V1.croisement_debut = c
```

Exercice 4 *MP 2019*

Indications sur les résultats :

1. avec $c = 9$ on trouve les croisements 3,4 et 7.
2. vérifier avec $c = 9$ qu'on retrouve bien les coordonnées des croisements précités.
3. Amusante, cette jointure de la table *Voie* avec elle-même, non ?

4 Mines-Ponts 2019

Au cours du développement des fonctions nécessaires à la manipulation des nombres premiers on s'aperçoit que le choix des algorithmes pour évaluer chaque fonction est primordial pour garantir des performances acceptables. On souhaite donc mener des tests à grande échelle pour évaluer les performances réelles du code qui a été développé. Pour ce faire on effectue un grand nombre de tests sur une multitude d'ordinateurs. Les données sont ensuite centralisées dans une base de données composée de deux tables.

La première table est ORDINATEURS et permet de stocker des informations sur les ordinateurs utilisés pour les tests. Ses attributs sont :

- `nom` TEXT, clé primaire, le nom de l'ordinateur.
- `gflops` INTEGER la puissance de l'ordinateur en milliards d'opérations flottantes par seconde.
- `ram` INTEGER la quantité de mémoire vive de l'ordinateur en Go.

La seconde table est FONCTIONS et stocke les informations sur les tests effectués pour différentes fonctions en cours de développement. Ses attributs sont :

- `id` INTEGER l'identifiant du test effectué.
- `nom` TEXT le nom de la fonction testée.
- `algorithme` TEXT le nom de l'algorithme qui permet le calcul de la fonction testée.
- `test_sur` TEXT le nom du PC sur lequel le test a été effectué.
- `temps_exec r` INTEGER le temps d'exécution en millisecondes.

Exercice 5 *MP 2019*

Les questions du concours :
Écrire des requêtes SQL permettant de :

1. Connaître le nombre d'ordinateurs disponibles et leur quantité moyenne de mémoire vive.
2. Extraire les noms des PC sur lesquels l'algorithme *rectangles* n'a pas été testé pour la fonction nommée *li*.
3. Pour la fonction nommée *Ei*, trier les résultats des tests du plus lent au plus rapide. Pour chaque test retenir le nom de l'algorithme utilisé, le nom du PC sur lequel il a été effectué et la puissance du PC.

Indications sur les résultats :

1. 500 bécanes, pour une moyenne de 29.52 Go
2. 481 ordinateurs sont concernés.
3. 96 tests sont concernés.

5 Mines-Ponts 2016

Pour suivre la propagation des épidémies, de nombreuses données sont recueillies par les institutions internationales comme l'O.M.S. Par exemple, pour le paludisme, on dispose de deux tables :

- La table **palu** recense le nombre de nouveaux cas confirmés et le nombre de décès liés au paludisme ; certaines lignes de cette table sont données en exemple (on précise que **iso** est un identifiant unique pour chaque pays) :

nom	iso	annee	cas	deces
Bresil	BR	2009	309316	85
Bresil	BR	2010	334667	76
Kenya	KE	2010	898531	26017
Mali	ML	2011	307035	2128
Ouganda	UG	2010	1581160	8431
...				

- La table **demographie** recense la population totale de chaque pays ; certaines lignes de cette table sont données en exemple :

pays	periode	pop
BR	2009	193020000
BR	2010	194946000
KE	2010	4090900
ML	2011	14417000
UG	2010	33987000
..		

Exercice 6

Mines-Ponts 2016

Les questions du concours :

1. Au vu des données présentées dans la table **palu**, parmi les attributs **nom**, **iso** et **annee**, quels attributs peuvent servir de clé primaire ? Un couple d'attributs pourrait-il servir de clé primaire ? (on considère qu'une clé primaire peut posséder plusieurs attributs). Si oui, en préciser un.
2. Écrire une requête en langage SQL qui récupère depuis la table **palu** toutes les données de l'année 2010 qui correspondent à des pays où le nombre de décès dus au paludisme est supérieur ou égal à 1000.

   On appelle *taux d'incidence d'une épidémie* le rapport du nombre de nouveaux cas pendant une période donnée sur la taille de la population-cible pendant la même période. Il s'exprime en général en « nombre de nouveaux cas pour 100 000 personnes par année ». Il s'agit d'un des critères les plus importants pour évaluer la fréquence et la vitesse d'apparition d'une épidémie.
3. Écrire une requête en langage SQL qui détermine le taux d'incidence du paludisme en 2011 pour les différents pays de la table **palu**.
4. Écrire une requête en langage SQL permettant de déterminer le nom du pays ayant eu le deuxième plus grand nombre de nouveaux cas de paludisme en 2010 (on pourra supposer qu'il n'y a pas de *ex æquo* pour les nombres de cas).

Quelques indications sur les résultats obtenus :

1. Voir le cours (?)
2. Seule la Chine répond aux critères de la requête.
3. 44 est des poussières pour l'Angola.
4. L'heureux gagnant est l'Indonésie.

6 X-ens mp & pc 2016

Une représentation simplifiée, réduite à deux tables, de la base de données d'un réseau social est donnée par :

- La table INDIVIDUS, qui répertorie les individus composant les réseau social à l'aide des champs :
  - *id* (clé primaire), un entier identifiant chaque individu.
  - *nom*, une chaine de caractères donnant le nom de famille de l'individu.
  - *prénom*, une chaine de caractères donnant le prénom de l'individu.
- La table LIENS, qui répertorie les liens d'amitiés entre individus et contient les champs :
  - *id1*, entier identifiant le premier individu du lien d'amitié.
  - *id2*, entier identifiant le second individu du lien d'amitié.

id	nom	prenom
1	Knuth	Donald
2	Dijkstra	Edsger
⋮	⋮	⋮

id1	id2
1	2
2	1
⋮	⋮

On supposera par ailleurs que pour tout couple (x, y) dans la table LIENS, le couple (y, x) est également présent dans la table.

Exercice 7 *X-ens mp & pc 2016*

Les questions du concours :

1. Écrire une requête SQL qui renvoie les identifiants des amis de l'individu d'identifiant x
2. Écrire une requête SQL qui renvoie les (noms,prénoms) des amis de l'individu d'identifiant x.
3. Écrire une requête SQL qui renvoie les identifiants des individus qui sont amis avec au moins un ami de l'individu d'identifiant x.

7 Mines-Ponts 2015

Les imprimantes sont des systèmes mécatroniques fabriqués en grande série dans des usines robotisées. Pour améliorer la qualité des produits vendus, il a été mis en place différents tests de fin de chaîne pour valider l'assemblage des produits. Pour un de ces tests, un opérateur connecte l'outil de test sur la commande du moteur de déplacement de la tête d'impression et sur la commande du moteur d'avance papier. Une autre connexion permet de récupérer les signaux issus des capteurs de position.

Différentes commandes et mesures sont alors exécutées. Ces mesures sont envoyées par liaison de données sous la forme d'une suite de caractères ASCII vers un ordinateur.

Cet ordinateur va effectuer différentes mesures pour valider le fonctionnement de l'electro-mécanique de l'imprimante. L'ensemble des mesures et des analyses est sauvegardé dans un fichier texte. Cette sauvegarde s'effectue dans une base de données et, afin de minimiser l'espace occupé, les fichiers sont compressés. La base de données permet à l'entreprise d'améliorer la qualité de la production par diverses études statistiques.

La suite des valeurs de mesure du courant en Ampère du moteur de la tête d'impression est contenue dans une liste. Les mesures ont été effectuées toutes les 2 ms. On calcule alors la moyenne I_{moy} de ces valeurs ainsi que leur écart-type I_{ec}.

Après son assemblage et avant les différents tests de validation, un numéro de série unique est attribué à chaque imprimante. A la fin des tests de chaque imprimante, les résultats d'analyse ainsi que le fichier contenant l'ensemble des mesures réalisées sur l'imprimante sont rangés dans la table *testfin* :

testfin

nSerie	dateTest	...	Imoy	Iec	...	fichierMes
230-588ZX2547	2012-04-22 14-25-45		0.45	0.11		mesure31025.csv
230-588ZX2548	2012-04-22 14-26-57		0.43	0.12		mesure41026.csv
⋮	⋮	⋮	⋮	⋮	⋮	⋮

Lorsqu'une imprimante satisfait les critères de validation, elle est enregistrée dans la table *production* avec son numéro de série, la date et l'heure de sortie de production ainsi que son type :

production

Num	nSerie	dateProd	type
20	230-588ZX2547	2012-04-22 14-25-45	JETDESK-1050
21	230-588ZX2549	2012-04-22 15-53-24	JETDESK-3050

Exercice 8

Mines-Ponts 2015

Questions du concours :

1. Rédiger une requête SQL permettant d'obtenir les numéros de série des imprimantes ayant une valeur de Imoy comprise strictement entre deux bornes Imin et Imax [a].
2. Rédiger une requête SQL permettant d'obtenir les numéros de séries, la valeur de l'écart type et le fichier de mesures des imprimantes ayant une valeur de Iec strictement inférieure à la valeur moyenne de la colonne Iec.
3. Rédiger une requête SQL qui permettra d'extraire à partir de la table *testfin* le numéro de série et le fichier de mesures correspondant aux imprimantes qui n'ont pas été validées en sortie de production.

Quelques indications sur les résultats obtenus :

1. Avec Imin = 0.42 et Imax= 0.46 on trouve 37 imprimantes.
2. Il y a 55 résultats.
3. On en trouve 11, à commencer par la $230 - 588ZX2548$ dont on avait remarqué qu'elle avait échoué aux tests.

a. question hors-programme qui implique des constantes SQL, mais quand même compréhensible