

Objectifs

Fouille de données à l'aide de fonctions OLAP et de Window Functions SQL sous Postgres.

On considère la table suivante, obtenue par jointure des informations pertinentes d'un système de gestion de notes

```
notes(mat, ncont, netu, groupe, note);
```

Récupérez sur Moodle le script `tp04.sql` contenant à la fois la structure et quelques exemples de données et exécutez le.

1 Upsert

Ecrire en SQL les requêtes suivantes :

1. Ecrire deux requêtes Insert rajoutant une note de 5 au contrôle 1 de gestion pour les étudiants 10 et 11 du groupe k,k
2. Ecrire quatre requêtes de type Upsert au contrôle 1 de gestion pour les étudiants 10,11,12,13 respectivement dans les groupes k,k,l,l avec une note de 10 si ce (mat,ncont,netu) existe, et une note aléatoire entre 0 et 20 s'il n'existe pas.
3. La table `ma j` contient des données à ajouter ou à modifier selon les cas. Intégrez l'ensemble de ces données à la table `notes` une seule requête.

2 Valeurs nulles

Ecrire en SQL les requêtes suivantes :

1. Afficher uniquement les notes de l'étudiant 13. Combien de lignes sont affichées ? combien de notes sont affichées ?
2. Calculer la moyenne des notes de l'étudiant 11
3. Calculer la moyenne des notes de l'étudiant 11 si on considère 0 aux notes non saisies
4. Calculer la moyenne des notes de l'étudiant 11 si on considère 20 aux notes non saisies
5. Calculer la moyenne des notes de l'étudiant 11 si on considère qu'il a la moyenne globale des notes à ses notes non présentes. Vous noterez les différences entre les résultats des 4 requêtes précédentes.
6. Affichez les lignes contenant une valeur nulle
7. Triez la table `notes` par valeur de note croissante, en mettant les valeurs nulles en tête de liste
8. Triez la table `notes` par valeur de note décroissante, en mettant les valeurs nulles en fin de liste.

3 Requêtes complexes

Ecrire en SQL les requêtes suivantes :

1. Afficher la moyenne du nombre de notes non nulles de chaque étudiant
2. Liste des étudiants à la moyenne la plus élevée (maximum)

4 OLAP

Ecrire en SQL les requêtes suivantes. vous prendrez garde pour chaque question au nombre de lignes obtenues

1. Moyenne par mat et ncont
2. Moyenne par mat, ncont, groupe
3. idem mais en ajoutant les moyennes de tous les sous-ensembles de (mat,ncont,groupe) ordonnés
4. idem mais avec les moyennes de toutes les permutations possibles. Combien cela fait-il de lignes ?
5. idem mais avec toutes les moyennes des produits cartésiens entre (mat,ncont) et (groupe)

5 Window functions

Ecrire en SQL la requête permettant d'afficher la table des notes, avec comme colonne supplémentaire :

1. une colonne contenant la somme de toutes les notes
2. une colonne contenant le pourcentage de chaque note sur le total des notes
3. une colonne contenant la somme des notes du groupe correspondant
4. une colonne contenant le rang décroissant de cette note sur l'ensemble des notes; quel rang possède la note suivant une égalité ?
5. une colonne contenant le rang dense (sans trous en cas d'égalité) de cette note sur l'ensemble des notes; quel rang possède la note suivant une égalité ?
6. idem, mais le rang dans le groupe correspondant
7. une colonne contenant la valeur médiane de la note par rapport à la totalité des notes non nulles
8. idem mais la médiane dans le groupe correspondant
9. une colonne contenant le quartile de la note, uniquement pour les étudiants du groupe k, pour les notes non nulles
10. Afficher uniquement les étudiants du premier quartile (les meilleurs) de chaque groupe