

TP5 : Base de données et langage SQL

La compression des fichiers est une pratique aussi courante qu'indispensable en informatique. Elle consiste à stocker les données dans un nouveau format moins encombrant que l'original. Il y'a deux types majeurs de compressions: la compression sans perte et la compression avec perte.

Le taux de compression est une mesure de performance d'un algorithme de compression. Il est exprimé en pourcentage. Le taux de compression est le gain en volume rapporté au volume initial des données. Plus le taux de compression est élevé, plus la taille du fichier compressé résultant est faible. la formule de compression s'écrit:

$$\text{taux} = \left(1 - \frac{\text{taille du fichier compressé}}{\text{taille initiale du fichier}}\right) * 100$$

On dispose d'une base de données de chemin "C:\BDcompressions.sql" composé de trois tables:

Algorithme (format(texte),type(entier))

La table Algorithme contient différents formats d'algorithme de compressions. Le champ format est la clé primaire. Le champ type contient la valeur 1 si l'algorithme de compression est de type avec perte de données ou 0 sinon

format	type
7z	0
bz2	0
gz	0
jpeg	1
rar	0
zip	0
mp3	1
huffman	0

Fichier (numeroF (entier), nomF (texte), tailleF (entier))

La table Fichier contient des fichiers de différents types : texte, image, son, video,... . Le champ numeroF est la clé primaire. Le champ nomF contient le nom complet de chaque fichier. Le champ tailleF contient la taille originale du fichier, exprimée en Octet.

numeroF	nomF	tailleF
1	Cours Python.pdf	6420768
2	Nature.bmp	563786
3	Langage SQL.doc	335367
4	Chanson.wav	45457780
...

Compression (numeroF (entier), format (texte), tailleC (entier))

La table Compression contient les compressions de certains fichiers de la table Fichier. Les champs numeroF et format sont les clés étrangères. Le champ tailleC contient la taille du fichier compressé, exprimée en octet. Un fichier peut être compressé plusieurs fois, par différents algorithmes de compressions.

numeroF	format	tailleC
2	jpeg	667657
2	png	3434
1	zip	45656
1	cab	345346
4	mp3	4544
3	7z	66454
...

1. Ecrire la requête SQL qui permet d'ajouter dans la table Fichier, le document prepas.pdf ayant pour numéro 14 et tailleF égale à 14725.
2. Ecrire la requête SQL qui modifié la table Fichier en mettant à jour la taille du fichier numéro 10. La nouvelle taille est égale à 54201
3. Ecrire la requête SQL qui supprime de la table Fichier, le document dont le numéro est 14.
4. **Ecrire en algèbre relationnelle, une requête qui donne pour résultat: Les noms des fichiers dont la taille originale est comprise entre 1 Kilo-octet et 1 Méga-octet.**
5. Ecrire la requête SQL qui affiche le nom et la taille du fichier ayant la plus grande taille avant compression
6. Ecrire une requête SQL qui retourne les noms des fichiers compressés à l'aide du format "zip"
7. Ecrire la requête SQL qui affiche les noms des fichiers ayant une taille avant compression supérieur strictement à la moyenne de la colonne tailleF
8. Ecrire la requête SQL qui retourne les noms des formats n'ayant jamais été utilisé pour compresser un fichier.
9. Ecrire la requête SQL qui affiche les noms des fichiers dont la taille avant compression est égale à la taille des fichiers compressé au format zip
10. **Ecrire une requête SQL, qui donne pour résultat: Les noms et la taille des fichiers texte, dont le nom se termine par : .doc ou docx, triés dans l'ordre alphabétique des noms des fichiers.**
11. **Ecrire une requête SQL qui donne pour résultat: Les noms des fichiers compressés, les formats de compression, les types de compression, et le taux de compression de chaque fichier, dont le taux de compression dépasse 40%, triés dans l'ordre des numéros des fichiers**
12. **Ecrire une requête SQL qui donne pour résultat; Les algorithmes sans perte de données et le compte des fichiers compressés par ces algorithmes, dont ce compte est égal à 3,5 ou 8.**
13. **Ecrire une requête SQL qui donne pour résultat: Les 3 premiers grands taux de compressions, triés dans l'ordre croissant**