Index

En étudiant le dump MySQL **db\_space\_invaders.sql vous** constaterez que vous ne trouvez pas le mot clé **INDEX.**

1. **Pourtant certains index existent déjà. Pourquoi ?**

Ils sont créés sur la ou les colonnes qui serviront à filtrer les données. Les contraintes « Primary KEY », « Foreign KEY » et « Unique » créent automatiquement des index sur les colonnes concernées (pour les Foreign KEY, les index sont souvent utilisés pour accélérer les opérations de jointure et de vérification des contraintes). Les index sont utilisés pour assurer l’intégrité de la table, ensuite pour une recherche plus rapide comme un liste bien triée et par ordre alphabétique. Certains index ne sont pas écrits dans les requêtes sql mais sont quand même présente dans la db.

1. **Quels sont les avantages et les inconvénients des index ?**

**Avantages :**

* **Amélioration de la performance de recherche :** Les index permettent de réduire grandement le temps nécessaire pour retrouver les données dans les grandes tables.
* **Efficacité des jointures :** Les index sont très bien agencés pour améliorer la performance des requêtes qui impliquent des jointures entre plusieurs tables.
* **Optimisation de tri et de regroupement :** Les index sont capables d’accélérer les opérations de tri avec comme : (ORDER BY et GROUP BY).

**Inconvénient :**

* **Augmentation de l’espace de stockage :** Pour chaque index mis de l’espace de stockage sera pris par celui-ci.
* **Ralentissement de certaine requête :** Les requêtes comme : INSERT, UPDATE, DELETE, sont ralentis par les index car en plus de la table qu’il faut mettre à jour, l’index demandera aussi une mise à jour pour être toujours équivalent à la table.

1. **Sur quel champ (de quelle table), cela pourrait être pertinent d’ajouter un index ? Justifier votre réponse.**

Les index sont très utiles si nous voulons l’utiliser dans une table qui souhaiterait avoir la possibilité d’avoir un tri (GROUP BY), les tables qui ont des jointures (Join) et aussi des moments où la clause WHERE est utilisé.

Backup/ Restore

Nous souhaitons réaliser une sauvegarde (Backup) de la base de données db\_space\_invaders.

Ensuite, nous souhaitons nous assurer que cette sauvegarde est correcte en la rechargeant dans MySQL (opération de restauration).

Donner la commande permettant de faire :

1. **Un Backup de la base de données db\_space\_invaders ?**

Depuis le docker directement

« mysqldump -u root -proot --databases db\_space\_invaders > db\_space\_invaders\_backup.sql »

Depuis l’invite de commande sans docker

« docker exec -i db mysqldump -u root -proot --databases db\_space\_invaders > db\_space\_invaders\_backup.sql »

1. **Un restore de la base de données db\_space\_invaders ? En expliquant en détail chaque commande utilisée.**

Depuis le docker directement

« mysql -u root -proot < db\_space\_invaders\_backup.sql »

* mysql : Utilitaire MySQL pour accéder à la base de données.
* -u root : Spécifie l'utilisateur MySQL.
* -p root : Mot de passe de l'utilisateur.
* db\_space\_invaders : Nom de la base de données où les données seront restaurées.
* < db\_space\_invaders\_backup.sql : Utilisation du fichier de sauvegarde pour restaurer les données dans la base.

Depuis l’invite de commande sans docker

« docker exec -i db mysql -u root -proot < db\_space\_invaders\_backup.sql »