

## fiche 4

---

Cette fiche se fonde sur les tables `joueurs`, `jeux` et `parties` de la fiche précédente.

**Exercice 1 :****Q 1 .**

Indiquez les commandes SQL permettant de recueillir les résultats suivants. Entre parenthèses, sont précisées, si nécessaire, les informations que l'on souhaite voir apparaître dans la réponse.

NB : Retour sur la fonction **count** (et autres fonctions de regroupement) :

l'expression `count(distinct nom_d_attribut)` compte le nombre de valeurs distinctes pour cet attribut.

Cette forme s'applique également aux autres fonctions de regroupement (mais c'est moins utile).

1. la liste des jeux d'un prix  $> 100$  ayant été joués plus de 1000 fois. (id du jeu, nombre de parties)
2. la liste des jeux d'un prix  $> 100$  ayant été joués par plus de 50 joueurs différents. (id du jeu, nombre de joueurs)
3. le nombre de points totalisés par l'ensemble des joueurs (points)
4. le nombre de points totalisés par l'ensemble des joueurs, par année (année, points)
5. le nombre de points totalisés par l'ensemble des joueurs, par mois (année, mois, points)
6. le nombre de points totalisés par joueur et par année (année, pseudo, points)
7. la liste des joueurs n'ayant jamais joué (pseudo)
8. la liste des joueurs ayant joué à des jeux distincts (pseudo)
9. la liste des jeux ayant été joués moins de 10 fois. (id du jeu, nombre de parties)

**Q 2 .**

- proposez une requête affichant la liste des joueurs classés par ordre décroissant du nombre de points qu'ils ont gagné.
- la clause `LIMIT n` (où  $n$  est un entier) permet de limiter à  $n$  (maxi) le nombre de tuples renvoyés par la requête. L'utilisation de `LIMIT` n'a réellement de sens que pour des requêtes où l'ordre des tuples est fixé par `ORDER BY` (si non cela reviendrait à renvoyer  $n$  tuples plus ou moins aléatoirement)  
Utilisez la clause `LIMIT` pour ne renvoyer que le « top 3 » (3 joueurs ayant gagné le plus de points) dans l'hypothèse où il n'y a pas d'ex-aequo.
- cette requête convient-elle en cas d'ex-aequo? (illustrez votre réponse par un ou plusieurs exemples)
- la notion même de « top 3 » a-t-elle toujours un sens dans ce cas?

NB : Une bonne réponse pour le cas général fait appel à la fonction `rank() over(...)`, et donc aux opérations de fenêtrage (ou partition), hors du programme du cours.