1. Les indexs
2. Expliquer à quoi sert un index en base de données.

Un indexe permet d’améliorer la rapidité des requêtes SQL. On peut mettre un Index sur un attribut souhaité et on pourras y faire référence plus facilement.

2. Trouver les requêtes SQL afin de :

- compter le nombre d'étudiants qui sont dans la maison "Gryffindor" ;

- mesurer le temps de la requête avec la commande SHOW PROFILE

|  |  |
| --- | --- |
| starting | 0.000088 |
| Executing hook on transaction | 0.000003 |
| starting | 0.000006 |
| checking permissions | 0.000005 |
| Opening tables | 0.000044 |
| init | 0.000004 |
| System lock | 0.000008 |
| optimizing | 0.000013 |
| statistics | 0.000021 |
| preparing | 0.000027 |
| executing | 0.000141 |
| end | 0.000004 |
| query end | 0.000003 |
| waiting for handler commit | 0.000011 |
| closing tables | 0.000007 |
| freeing items | 0.000056 |
| cleaning up | 0.000013 |

(https://dev.mysql.com/doc/refman/8.0/en/show-profile.html) ;

- ajouter un index sur la colonne "house\_id" de la table "students" ;

- mesurer à nouveau le temps de la requête après l'ajout de l'index ;

|  |  |
| --- | --- |
| starting | 0.000066 |
| Executing hook on transaction | 0.000004 |
| starting | 0.000006 |
| checking permissions | 0.000005 |
| Opening tables | 0.000036 |
| init | 0.000005 |
| System lock | 0.000006 |
| optimizing | 0.000021 |
| statistics | 0.000058 |
| preparing | 0.000020 |
| executing | 0.000034 |
| end | 0.000003 |
| query end | 0.000002 |
| waiting for handler commit | 0.000009 |
| closing tables | 0.000006 |
| freeing items | 0.000048 |
| cleaning up | 0.000009 |

- mesurer à nouveau le temps de la requête mais sans index.

Attention : dans MySQL, un index est utilisé par défaut par le SGBD pour optimiser les

requêtes. Si vous souhaitez mesurer les performances d'une requête sans utiliser l'index que

vous venez de créer, vous pouvez utiliser IGNORE INDEX.

|  |  |
| --- | --- |
| starting | 0.000080 |
| Executing hook on transaction | 0.000003 |
| starting | 0.000006 |
| checking permissions | 0.000005 |
| Opening tables | 0.000047 |
| init | 0.000005 |
| System lock | 0.000007 |
| optimizing | 0.000011 |
| statistics | 0.000072 |
| preparing | 0.000022 |
| executing | 0.000048 |
| end | 0.000004 |
| query end | 0.000002 |
| waiting for handler commit | 0.000010 |
| closing tables | 0.000007 |
| freeing items | 0.000055 |
| cleaning up | 0.000009 |
|  |  |
|  |  |

3. Pour les requêtes suivantes, vous devez dire à quoi correspond chaque requête.

Ensuite, vous devez mesurer le temps de la requête, rajouter un index, mesurer encore

une fois le temps de la requête.

**Requête a**

SELECT houses.house\_name, courses.course\_name, COUNT(\*) AS

num\_students

FROM students

JOIN houses ON students.house\_id = houses.house\_id

JOIN courses ON students.course\_id = courses.course\_id

GROUP BY houses.house\_name, courses.course\_name

ORDER BY num\_students DESC;

Cette requette compte le nombre détudiant de chaque classe qui suivent le même cours et range le résultats dans l’ordre décroissant.

|  |  |  |  |
| --- | --- | --- | --- |
| starting | 0.000080 |  |  |
| Executing hook on transaction | 0.000003 |  |  |
| starting | 0.000005 |  |  |
| checking permissions | 0.000002 |  |  |
| checking permissions | 0.000001 |  |  |
| checking permissions | 0.000001 |  |  |
| checking permissions | 0.000002 |  |  |
| Opening tables | 0.000051 |  |  |
| init | 0.000004 |  |  |
| System lock | 0.000007 |  |  |
| optimizing | 0.000008 |  |  |
| statistics | 0.000049 |  |  |
| preparing | 0.000013 |  |  |
| Creating tmp table | 0.000047 |  |  |
| executing | 0.000851 |  |  |
| end | 0.000005 |  |  |
| query end | 0.000002 |  |  |
| waiting for handler commit | 0.000020 |  |  |
| closing tables | 0.000009 |  |  |
| freeing items | 0.000070 |  |  |
| cleaning up | 0.000011 |  |  |

**Requête b**

SELECT student\_name, email

FROM students

WHERE course\_id IS NULL;

Cette requette affiche les étudiants et leur email qui ne sont inscrit à aucun cour.

**Requête c**

SELECT houses.house\_name, COUNT(\*) AS num\_students

FROM students

JOIN houses ON students.house\_id = houses.house\_id

WHERE EXISTS (

SELECT \*

FROM courses

WHERE course\_name IN ('Potions', 'Sortilèges', 'Botanique')

AND course\_id = students.course\_id

)

GROUP BY houses.house\_name;

Cette requête compte le nombre d’étudiants par maison qui participent à un cour.

**Requête d**

SELECT s.student\_name, s.email

FROM students s

JOIN (

SELECT student\_id, year\_id, COUNT(DISTINCT course\_id) AS

num\_courses

FROM students

GROUP BY student\_id, year\_id

) AS sub

ON s.student\_id = sub.student\_id AND s.year\_id = sub.year\_id

JOIN (

SELECT year\_id, COUNT(DISTINCT course\_id) AS num\_courses

FROM students

GROUP BY year\_id

) AS total

ON s.year\_id = total.year\_id AND sub.num\_courses =

total.num\_courses

WHERE sub.num\_courses = total.num\_courses;

Cette requette affiche : le **nom**, **l’email**, **l’année** et **l’id** de l’étudiant ainsi que le nombre de cours diférent qu’il participe et le nombre de cours diférent que participe les élèves de son année à condition que les deux dernier nombre soit identiques.