

TP 2 : SQL (Articles scientifiques)

Le métier de chercheur consiste, entre autres, à faire connaître ses résultats par l'intermédiaire d'articles de recherche. Le nombre des articles de recherche publiés croît extrêmement rapidement, ce qui rend difficile à la fois le travail de collecte d'articles sur un sujet en particulier et le stockage de ces masses d'informations. L'objectif de ce sujet est de créer une application permettant aux chercheurs de gérer leur bibliographie (ensemble d'articles) à l'aide d'une base de données en leur permettant d'avoir une description rapide des articles disponibles mais aussi d'avoir accès à des notes et brefs commentaires associés à ces articles.

Description des articles. Un article possède plusieurs attributs comme son titre, qui doit être unique, ses auteurs, un résumé et un type comme "demo", "recherche-long", "recherche-court" etc. Un article avec un titre donné peut être publié sous différents supports de publication : conférence, journal, thèse, livre etc. Tout support a donc un type (conférence, journal etc.) et un nom. On doit pouvoir déterminer l'année de publication d'un article sur un support donné, sachant qu'un article peut être publié dans plusieurs conférences et/ou journaux la même année ou à des années différentes. Néanmoins, un article donné, sur un support donné ne sera publié qu'une seule année. Un auteur est caractérisé par son adresse email. De plus, un nom, un prénom, un site web personnel, et le ou les laboratoires auquel il appartient (en effet un chercheur peut travailler pour différents laboratoires en même temps) lui sont associés. Un laboratoire est caractérisé par son nom et possède un sigle, une adresse, et éventuellement une adresse url indiquant son site web.

Notation et Annotation des articles. Les utilisateurs de notre outil sont des auteurs (mais tous les auteurs ne sont pas nécessairement utilisateurs). On aimerait permettre aux utilisateurs de notre outil de noter et d'étiqueter les articles gérés par le système (ces notes et tags ne dépendent pas du support de publication). La provenance (l'utilisateur ayant donné ce tag ou cette note) des tags et des notes est stockée dans le système. Cette provenance peut être utilisée dans les requêtes. Il est possible de trouver par exemple tous les auteurs ayant noté leurs propres articles. Un utilisateur peut apposer des tags sur n'importe quel article. Un tag n'est caractérisé que par sa valeur. A priori, n'importe quelle chaîne de caractères peut faire office de valeur à un tag. Chaque utilisateur peut aussi attribuer une note (entier compris entre 0 et 5) à un article en particulier. Un utilisateur ne peut noter qu'une seule fois chaque article.

1 Modélisation

À partir du code SQL fourni de création des tables et de spécification des contraintes, donnez une modélisation en diagramme de classes UML de l'énoncé.

Appliquez ensuite le script de création des tables, puis celui de remplissage de la base.

2 Requêtes de sélection

Répondez aux questions suivantes par une requête sur la base fournie :

1. Quel est l'e-mail de Peter Buneman ?
2. Quels sont les articles dont Susan Davidson est l'un des auteurs ?
3. Quels sont les laboratoires qui ont la même url (donnez la liste des laboratoires qui partagent une url commun avec un autre laboratoire) ?
4. Quels sont les co-auteurs de Susan Davidson ?
5. Pour chaque auteur, donnez le nombre de laboratoires dans lesquels ils travaillent.
6. Quels sont les auteurs ayant annoté au moins un article ?
7. Quelle est la moyenne des notes données par Luc Segoufin ?
8. Pour chaque article, quelle la moyenne des notes données à cet article par ses propres auteurs ?
9. Quels sont les auteurs n'ayant annoté aucun de leurs propres articles ?
10. Quels sont les auteurs qui ont noté tous les articles ?
11. Listez par ordre décroissant les auteurs en fonction du nombre d'articles qu'ils ont publiés.
12. Quel est le premier auteur de la liste de la question précédente ?
13. Quels sont les auteurs ayant le plus publié ?