

Bases de Données

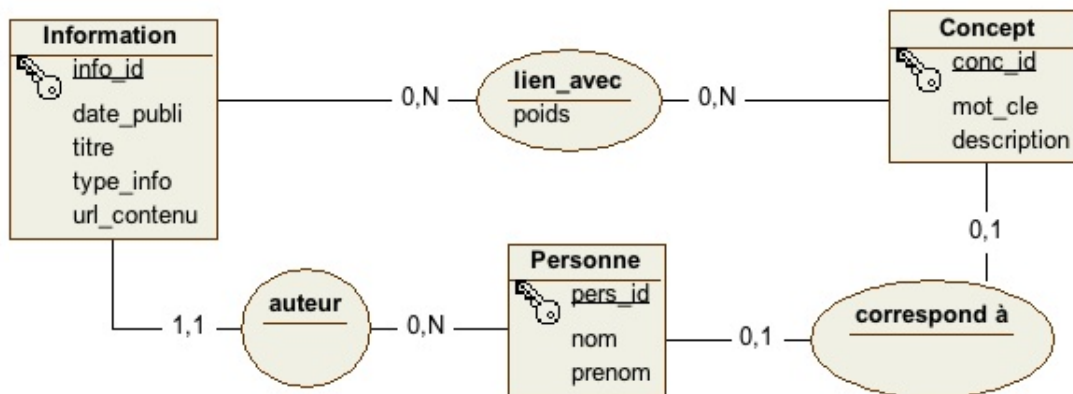
le 18 décembre 2013

Devoir Surveillé

durée 3h, documents autorisés, appareils mobiles de communication interdits.

Exercice 1 : *Ce sujet porte sur la conception d'un site d'information.*

La société "Daily Planet" souhaite arrêter son édition papier et passer à une version web. Les informaticiens de la société proposent le modèle conceptuel suivant :



Dans ce modèle, une **information** peut être de plusieurs types (attribut `type_info`) : du texte, du son, une image ou une vidéo. Dans tous les cas, le contenu de l'information sera conservé dans un fichier, et la base mémorise simplement l'URL de ce contenu. Toute information possède obligatoirement un identifiant numérique, une date de publication (date+heure) et un titre. Une information possède un **auteur**, cet auteur est une personne avec un nom, un prénom, et pour laquelle on définira un identifiant numérique `pers_id`. Afin de faciliter une recherche par mot-clé, on définit des **concepts**. Chaque concept possède un identifiant numérique, une description, et un mot-clé. La description n'est pas obligatoire, le plus important étant le mot-clé. Il n'y a pas deux concepts avec le même mot-clé. Une information parle de concepts, mais tous ces concepts n'ont pas forcément la même importance pour cette information. On va donc définir des **poinds**, qui représentent l'importance de tel concept pour telle information. Ces poids ont une valeur de 1 à 5 (du moins important au plus important). Une personne peut être un concept, à partir du moment où on donne une information qui la concerne. On a donc dans ce cas une association entre la personne et son concept. Pour traduire ce modèle conceptuel en modèle relationnel les informaticiens ont choisi de représenter cette association "correspond à" par un attribut `conc_id` dans la table `Personne`. On obtient la liste suivante de tables :

```

CONCEPT(conc_id, mot_cle, description)
PERSONNE(pers_id, nom, prenom, conc_id)
INFORMATION(info_id, date_publi, titre, url_contenu, type_info, auteur)
LIEN_AVEC(info_id, conc_id, poinds)

```

Question 1.1 : Créer les tables en pensant aux contraintes d'intégrité.

Question 1.2 : Donner les requêtes SQL qui permettent d’obtenir les informations suivantes, en respectant le schéma du résultat :

1. Les informations qui sont des vidéos. SCHEMA : (info_id, titre, date_publi)
2. Les titres des informations dont l’auteur est Clark Kent. SCHEMA : (titre)
3. Les personnes qui n’ont publié aucune information. SCHEMA : (pers_id, nom, prenom)
4. Le nombre d’informations publiées par auteur, en ne tenant pas compte des personnes qui n’ont rien publié. SCHEMA : (nom, prenom, nb_info)
5. La moyenne du nombre d’informations publiées par auteur, en tenant compte des personnes qui n’ont rien publié. SCHEMA : (moyenne_nb_info)
6. Les concepts avec leur total de poids attribués, concepts triés par poids total décroissant. SCHEMA : (conc_id, mot_cle, poids_total)
7. Le concept qui a le plus grand poids total. SCHEMA : (conc_id, mot_cle)

La suite du sujet comporte des questions de programmation en PL/SQL. Pour information, la violation d’une contrainte de clé primaire déclenche l’erreur de code -1 et la violation d’une contrainte de clé étrangère déclenche l’erreur de code -2291.

Les informaticiens se rendent compte qu’ils doivent souvent écrire des requêtes pour connaître le poids total d’un concept, ou le nombre d’informations en lien avec un concept. Pour éviter ces calculs fréquents, ils décident de mémoriser ces informations. Ils ajoutent une colonne `poids_total` et une colonne `nb_infos` à la table `Concept` :

```
alter table concept
add (poids_total number(5) default 0 not null,
nb_infos number(4) default 0 not null);
```

Question 1.3 : Quelle requête SQL doivent-ils exécuter pour que ces colonnes prennent les bonnes valeurs, sachant qu’il y a déjà des données dans les tables `Concept` et `Lien_avec` ?

Question 1.4 : Définir un trigger qui maintienne cette cohérence entre les données de la table `Lien_avec` et les colonnes `poids_total` et `nb_infos` de la table `Concept`.

On complète le schéma par un package stocké. Dans un premier temps, ce paquetage contient une procédure permettant d’ajouter un lien entre un concept et une information, **ou** de modifier le poids d’un lien existant. Il est bien sûr beaucoup plus fréquent d’ajouter un lien que d’en modifier le poids. Voici la spécification du paquetage :

```
create or replace
package pack_info as

    procedure definir_lien(l_info information.info_id%type,
                           le_concept concept.conc_id%type,
                           le_poids lien_avec.poids%type);

end pack_info;
```

Question 1.5 : Ecrire le corps de cette procédure.

Ajout de commentaires

Le site fonctionne bien, mais une enquête auprès des lecteurs montre qu'ils aimeraient pouvoir réagir par rapport aux informations qu'ils trouvent sur le site. L'équipe informatique s'adapte à cette nouvelle demande en ajoutant à la base une table de commentaires¹ :

COMMENTAIRE(*com_id*, *info_id*, *reponse_a*, *date_publi*, *texte*, *pseudo*)

- *com_id* est l'identifiant du commentaire
- *info_id* fait référence à l'information commentée, sauf si ce commentaire est en réponse à un autre commentaire, auquel cas *reponse_a* fait référence au commentaire auquel on répond. Il n'est pas possible de répondre à un commentaire qui est déjà une réponse, donc *reponse_a* contient l'identifiant d'un commentaire direct à une information. Pour ne pas mélanger les commentaires directs et les réponses aux commentaires, on définit une contrainte : l'une des 2 colonnes *info_id* et *reponse_a* n'est pas *null*, mais il y en a forcément une à *null*.
- La suppression d'une information entraîne la suppression de tous ses commentaires et de leurs réponses.
- Les attributs restant sont assez simples : *date_publi* est une date+heure de publication, le *texte* et le *pseudo* sont des chaînes de caractères. On limite le texte du commentaire à 256 caractères.

Voici par exemple un extrait de la table **Information** :

info_id	date_publi	titre	url_contenu	auteur	type_info
4	11/12/13	Chasse aux Rhinocéros	http://www.mesPhotos.com/rhino.jpg	11	image
6	14/12/13	Concours de maisons fleuries sur M6	http://www.monsite.com/concours.html	12	texte

et un extrait de la table **Commentaire** :

com_id	info_id	reponse_a	date_publi	pseudo	texte
8	6	null	14/12/13 03:12:19	gaston	C'est une très bonne idée, merci M6 !
9	null	8	14/12/13 03:12:33	lucette	bonne idée ... surtout pour les horticulteurs.
10	4	null	14/12/13 03:12:10	gaston	blabla
11	null	10	14/12/13 03:12:51	marguerite	reblabla
12	null	8	14/12/13 03:12:49	marguerite	personne ne vous oblige à regarder cette émission
13	6	null	14/12/13 04:12:03	jules	Encore un concours, ça commence à être lassant
14	null	13	14/12/13 04:12:41	julie	oui, après les cuisiniers, les pâtisseries, les chanteurs, ...

Question 1.6 : Donner l'instruction SQL qui permet de créer cette table.

On complète le paquetage **PACK_INFO** afin d'afficher la dernière information avec ses commentaires.

Question 1.7 :

1. Ecrire une procédure **afficher_reponses** qui prend en paramètre l'identifiant d'un commentaire, et affiche les réponses à ce commentaire (ou ne fait rien si ce commentaire n'a pas de réponse). Cette procédure a pour signature :

```
procedure afficher_reponses(le_com commentaire.com_id%type)
```
2. Ecrire une procédure **afficher_derniere_info** sans paramètre, qui affiche l'identifiant et le titre de la dernière information - i.e. celle qui a la date de publication la plus récente - ainsi que ses commentaires et leurs réponses (utiliser la procédure **afficher_reponses**).

Par exemple, avec les données de la table **Commentaire**, sachant que la dernière information est celle d'identifiant 6, la commande `execute pack_info.afficher_derniere_info;` entraîne un affichage du style de celui-ci :

¹C'est un stagiaire qui a trouvé cette solution, ce n'est pas forcément la meilleure.

```

derniere info : Concours de maisons fleuries sur M6
commentaires :
gaston : C'est une très bonne idée, merci M6 !
    reponse de lucette : bonne idée ... surtout pour les horticulteurs.
    reponse de marguerite : personne ne vous oblige à regarder cette émission
jules : Encore un concours, ça commence à être lassant
    reponse de julie : oui, après les cuisiniers, les pâtisseries, les chanteurs, ...

```

JDBC

Question 1.8 : Ecrire en java la classe `Journal` qui possède

- une variable d'instance `connect` de type `Connection`.
- une variable d'instance `stmt` de type `Statement`
- une variable d'instance `info_stmt` de type `PreparedStatement` (cf la méthode `selection_info`)
- un constructeur qui initialise `connect`, `stmt`, `info_stmt` et donc qui réalise la connexion à la base. Utiliser les mêmes paramètres de connexion que ceux de la salle TP.
- une méthode `menage` sans paramètre qui supprime les informations dont la date de publication est vieille de plus d'une année. Cette fonction renvoie le nombre d'informations supprimées. On supposera que la suppression d'une information entraîne automatique la suppression des liens avec les concepts, et tous les commentaires qui s'y rapportent.
- une méthode `selection_info` qui prend en paramètre un mot clé et affiche tous les titres et dates de publication des informations qui ont un lien avec ce mot clé. Pour cette méthode, vous utiliserez une requête préparée.

Exercice 2 : On considère une relation $R(A, S, L, G, C)$. Un nuplet (a, s, l, g, c) a la signification suivante : l'animateur a enseigne le sport s dans le lieu l au groupe g pendant le créneau horaire c . Soit \mathcal{F} l'ensemble des dépendances fonctionnelles (DF) suivantes :

$$\begin{array}{l}
 A \rightarrow S \\
 AC \rightarrow G \\
 CL \rightarrow A \\
 CG \rightarrow L
 \end{array}$$

Question 2.1 : D'après ces dépendances fonctionnelles,

1. un animateur donné peut-il enseigner des sports différents ?
2. un animateur donné peut-il encadrer des groupes différents ?
3. deux sports différents peuvent-ils avoir lieu simultanément au même endroit ?

Justifiez vos réponses.

Question 2.2 : Quelles sont les clés de la relation R ? Justifiez votre réponse.

Question 2.3 : Est-ce que R est 3NF ? Si R n'est pas 3NF, appliquez l'algorithme de normalisation pour obtenir une décomposition 3NF de R .