

Cartouche du document

Année : ING 2

Matière : Base de données 2A

Activité : Examen

Objectifs

Cet examen de base de données porte sur tous les TDs de Base de données pratiqués en salle.

Tous les documents papiers sont autorisés. Votre ordinateur n'est autorisé qu'en local.

La durée de l'examen est de 1h00 heure. Vous êtes notés sur 24.

Sommaire des exercices

1 - Des matières, des activités et des documents en SQL pur

2 - Des matières, des activités et des documents avec le type XMLTYPE

Corps des exercices

1 - Des matières, des activités et des documents en SQL pur

Énoncé :

Dans le site Arel, on a défini :

- des documents : document(idDocument, titre, contenu, EtatPublication)
- des activités : activite(idActivite, titre, contenu, EtatPublication)
- des matières : matiere(idMatiere, titre, syllabus)
- des dépôts de document dans une matière : lien_mat_doc(idMatiere, idDoc)
- des attachements de documents à une activité : lien_act_doc(idActivite, idDoc)
- des travaux à rendre dans une activité : travaux(idTravaux, titre, dateRemise, idActivite)
- des acteurs : acteur(login, nom, prenom, statut). Pour tout acteur, on a créé dans la BDD un compte dont le login est celui qui apparaît dans la table acteur. Le statut peut prendre les valeurs 'etudiant', 'professeur', 'administratif'.

Question 1)

Énoncé de la question

barème 8 pts

On désire tracer la vie d'un document. Dans une table nommée VieDocument(utilisateur, idDocument, dateMAJ) , on enregistre toutes les créations ou mise à jour d'un document. Sous Oracle, il existe les deux fonctions suivantes :

- sysdate qui renvoie la date courante;
- user qui renvoie le login de l'utilisateur.
- 1) Définir l'objet Oracle (le type de l'objet et le code PL/SQL associé) approprié qui permet d'alimenter la table VieDocument, afin de tracer la vie d'un document
- 2) Que faut-il changer dans l'objet précédent, si dans le cas d'une mise à jour d'un document existant

on ne trace que si le contenu change ?

Solution de la question

Il faut définir un trigger associé à l'insertion ou la modification d'un document.

```
CREATE OR REPLACE TRIGGER maj_document AFTER INSERT OR UPDATE ON
document
FOR EACH ROW
BEGIN
  INSERT INTO VieDocument VALUES (USER, :new.idDocument, SYSDATE);
END;
/
```

Si la trace ne se fait que dans le cas d'une mise à jour du contenu, alors le code du trigger devient :

```
CREATE OR REPLACE TRIGGER maj_document AFTER INSERT OR UPDATE ON
document
FOR EACH ROW
BEGIN
  IF INSERTING OR (UPDATING AND :new.contenu <> :old.contenu) then
    INSERT INTO VieDocument VALUES (USER, :new.idDocument, SYSDATE);
  END IF;
END;
/
```

Question 2)

Énoncé de la question

barème 8 pts

Ecrire une procédure qui permet d'ajouter un document dont on passe en paramètre le titre, le contenu et l'identifiant de la matière. Si l'utilisateur est un étudiant, on doit faire de telle sorte que l'ajout soit annulé et insérer dans une table nommée Interdit(login,dateAction) une ligne pour signaler qu'une action interdite a été effectuée par un étudiant.

Solution de la question

```
CREATE SEQUENCE seq_idDocument
/
CREATE OR REPLACE PROCEDURE ajout_document(ptitre varchar2,
pcontenu varchar2, pidMatiere number) IS
verr_etudiant : EXCEPTION;
vstatut acteur.statut%TYPE;
vidDocument document.idDocument%TYPE;
BEGIN
  SELECT statut INTO vstatut; FROM acteur WHERE login = USER;
  IF vstatut = 'etudiant' THEN RAISE verr_etudiant;
  SELECT seq_iddocument.nextval INTO vidDocument FROM dual;
  INSERT INTO document(vidDocument,ptitre,pcontenu,'brouillon');
  INSERT INTO lien_mat_doc(pidMatiere,vidDocument);
```

```
EXCEPTION
WHEN verr_etudiant THEN
  INSERT INTO Interdit(USER,SYSDATE);
WHEN OTHERS then
  -- Pour traiter l'erreur d'un id de matière inexistant
  DBMS_OUTPUT.PUT_LINE('Ajout echoué');
END;
/
```

2 - Des matières, des activités et des documents avec le type XMLTYPE

Énoncé :

On reprend l'exercice précédent. On désire stocker une activité dans une seule table en utilisant le type XMLTYPE. Les tables lien_act_doc et travaux n'existent plus.

Question 1)

Énoncé de la question

barème 4 pts

Redéfinir la table **activite** en utilisant de la meilleure façon possible le type XMLTYPE : on définira précisément la structure de l'arbre XML. On justifiera la réponse.

Solution de la question

On crée la table ActiviteXML :

```
CREATE TABLE ActiviteXML(idActivite number, DescriptifXML XMLTYPE,
CONSTRAINT pk_acitiviteXML PRIMARY KEY(idActivite));
```

Pour chaque activité, on stockera dans le champ descriptif un arbre XML. On décrit cet arbre par :

- un noeud noté descriptif pour le noeud racine;
- un attribut idMatiere dans le noeud descriptif;
- un noeud texte noté titre dans le noeud descriptif;
- un noeud texte noté contenu dans le noeud descriptif;
- de 0 à plusieurs noeuds notés travail dans le noeud descriptif;
- un noeud texte noté dateRemise dans un noeud travail;
- un noeud texte noté titre dans un noeud travail;
- de 0 à plusieurs noeuds vides notés documentAttache dans le noeud descriptif;
- un attribut obligatoire noté idDocument dans un noeud documentAttache.

Question 2)

Énoncé de la question

barème 4 pts

Ecrire une requête d'interrogation en utilisant SQL et XPATH qui permet d'afficher le contenu des documents d'une activité dont on connaît le titre. Ces contenus ne sont affichés que s'il y a au moins

un travail à rendre dans l'activité.

Solution de la question

```
SELECT EXTRACT(descriptifXML, 'descriptif/contenu').getStringVal()  
FROM activiteXML  
WHERE  
EXTRACT(descriptifXML, 'descriptif[titre = 'Examen BDD  
2']/travail').getStringVal() <> '';
```