

TP1 : PL/SQL

But du TP :

- Environnement SQLDeveloper et Dictionnaire des Données
- Procédures PL/SQL
- Gestion simple des curseurs



Cette série de TPs concerne des données musicales (Albums, Chansons, Artistes, Concerts).

Une chanson, dont on conserve le titre et l'année de composition, est écrite et composée par un artiste, qui n'est pas forcément le même. Un artiste peut être une personne physique ou bien un groupe, et est décrit par ses nom et prénom (pas de prénom dans le cas d'un groupe). Une personne pouvant faire partie de plusieurs groupes au cours du temps, on conserve ses dates de début et de fin d'appartenance éventuelles au(x) groupe(s). Un album est défini par son titre, son année de sortie et son label musical. Un album peut contenir des chansons d'un même artiste ou des chansons d'artistes différents. Il peut être disponible sur plusieurs supports (CD, vinyl, MP3), avec un prix de vente conseillé pour chaque support. Les chansons qui le composent, enregistrées sur des pistes, possèdent chacune une durée exprimée en secondes. Une chanson sur un album peut être interprétée par plusieurs artistes, de même qu'une chanson, en fonction des différents albums sur lesquels elle est, peut avoir des durées différentes.

Les concerts, dont on conserve le lieu et la date, sont composés de plusieurs chansons, qui peuvent être interprétées soit par un seul artiste, soit par plusieurs artistes différents.

On considère les relations suivantes, traduisant les contraintes exprimées ci-dessus :

Support (idSupport, nomSupport, TVA)

Relation décrivant les différents supports d'albums (Vinyl, MP3,...)

idSupport, nomSupport: identifiant et nom de support

TVA : TVA associée au support

Concert (idConcert, nomConcert, lieuConcert, dateConcert)

Relation décrivant les différents concerts

idConcert, nomConcert, lieuConcert, dateConcert : identifiant, nom, lieu et date du concert

InterpreterEnConcert (#idConcert,#idChanson, #idArtiste)`

Relation décrivant les différentes chansons interprétées dans un concert

idConcert : référence Concert(idConcert)

idChanson : référence Chanson(idChanson)

idArtiste : référence Artiste(idArtiste)

Album (idAlbum, titreAlbum, anneeAlbum, label)

Relation décrivant les différents albums

idAlbum, titreAlbum, anneeAlbum : identifiant, titre et année de l'album

label: label de l'album

Chanson (idChanson, titreChanson, anneeChanson, #idAuteur, #idCompositeur)

Relation décrivant les chansons

idChanson, titreChanson, anneeChanson: identifiant, titre et année de composition de la chanson

idAuteur : identifiant de l'auteur, référence Artiste(idArtiste)

idCompositeur : identifiant du compositeur, référence Artiste(idArtiste)

Artiste (idArtiste, nomArtiste, prenomArtiste)

Relation décrivant les artistes et les groupes

idArtiste, nomArtiste : identifiant et nom de l'artiste

prenomArtiste : prénom de l'artiste, nul si groupe

AppartenirGroupe (#idArtiste, #idGroupe, dateDebut, dateFin)

Relation décrivant l'appartenance des artistes à des groupes

idArtiste, idGroupe : référencent Artiste(idArtiste)

dateDebut : date de début de l'appartenance de l'artiste au groupe

dateFin : date de fin de l'appartenance de l'artiste au groupe (nul si l'artiste appartient toujours au groupe)

Piste (idPiste, noPiste, duree, #idChanson, #idAlbum)

Relation décrivant les différentes pistes composant un album

idPiste : identifiant de la piste, unique sur la BD

noPiste : numéro de la piste sur l'album

duree : durée en secondes de la piste

idChanson : identifiant de la chanson enregistrée sur la piste, référence Chanson(idChanson)

idAlbum : identifiant de l'album sur lequel se trouve la piste, référence Album(idAlbum)

Interpréter (#idPiste, #idArtiste)

Relation décrivant les artistes interprétant les pistes d'un album

idPiste : référence Piste(idPiste)

idArtiste : référence Artiste(idArtiste)

DisponibleSurSupport (#idAlbum, #idSupport, prixDeVente)

Relation décrivant les types de supports disponibles pour un album

idAlbum= identifiant de l'album, référence Album(idAlbum)

idSupport : identifiant du support, référence Support(idSupport)

prixDeVente : prix de vente conseillé de l'album sur le support

[Q0] Récupérez le fichier *creabase.sql* sous Moodle et copiez-le sur votre compte. Il contient les ordres de création des tables et leur remplissage. Examinez-le et exécutez-le.

Pour la bonne suite des TPs, il est très important que vous compreniez à fond le schéma de la BD.

Remontez au niveau conceptuel et faites le schéma entité-association de la base ou le diagramme UML.

Vous pouvez également représenter le schéma relationnel de façon « graphique ».

En cas de doute, n'hésitez pas à poser la question et à faire valider votre schéma.

Voyez notamment pourquoi on n'associe pas directement un artiste interprète à un album.

[Q1] En utilisant le Dictionnaire des Données :

- donner la requête permettant d'obtenir la liste des tables créées
- donner la requête permettant d'obtenir une description des colonnes (nom et type) de la table *Chanson*
- donner la requête permettant d'obtenir le nom et le type des contraintes créées

Note : Les deux premières questions sont « triviales » sous SQLDeveloper, car l'interface graphique permet l'affichage du Dictionnaire des Données. Essayez cependant de faire les requêtes d'interrogation, cela vous servira si un jour vous vous retrouvez à la ligne de commande sous SQL*Plus.

[Q2] Pour les besoins d'une application, on souhaite connaître le nombre d'albums disponibles sur le support 'Vinyl', ainsi que la liste de ces albums (*titre, année, label*) trié par ordre chronologique décroissant de parution.

A – Ecrire les deux requêtes SQL permettant d'afficher ces caractéristiques.

B – « Encapsulez » ces deux requêtes dans une procédure stockée PL/SQL *afficheSupport* effectuant la même chose. Vous coderez trois versions différentes de la procédure :

- en utilisant un curseur manuel
- en utilisant un curseur semi-automatique
- en utilisant un curseur automatique

Attention, on n'utilisera un curseur que quand cela est nécessaire !

Tester l'appel de cette procédure.

Remarque : On rappelle l'intérêt de la procédure PL/SQL dans ce cas :

- simplicité d'utilisation pour l'utilisateur : uniquement appel à la procédure, pas d'exécution de requêtes
- l'utilisateur ne « voit » pas le code.

[Q3] Créer une nouvelle procédure stockée reprenant la précédente mais permettant de paramétrer le nom du support pour lequel on veut le nombre d'albums et la liste de ces albums.

[Q4] A l'aide du dictionnaire des données, affichez le code source de vos programmes stockés.