

Travail Pratique MCD Merise, SQL2, PLSQL

MBDS-HAITI

1. Projet MCD MERISE, SQL2, PLSQL

FAIRE DES GROUPES DE 3. IDENTIFIER UN SUJET D'INTERET POUR VOUS OU VOTRE ENTREPRISE. OU CHOISIR UN DES SUJETS SCOLASTIQUES SUIVANTS :

1.1 Choix du sujet Vous choisirez, si ce n'est déjà fait, une application appartenant à la liste suivante :

- | | |
|--|--------------------------------------|
| 1) Agence de location de bateaux | 10) Gestion d'un labo. de chercheurs |
| 2) Agence immobilière | 11) Cabinet de notaires |
| 3) Agence bancaire | 12) Gestion de la bourse |
| 4) Gestion d'une bibliothèque | 13) Gestion d'un cabinet médical |
| 5) Tournois de tennis | 14) Gestion d'une agence de voyages |
| 6) Gestion d'une promotion d'étudiants | 15) Gestion de salons |
| 7) Gestion des menus d'un restaurant | 16) Gestion des anciens étudiants |
| 8) Gestion d'un club sportif | 17) Gestion d'une boîte de nuit |
| 9) Tournois de trampoline | 18) autres sujets de votre choix |

1.2 Spécification, Analyse et conception

Le résultat de cette phase doit être mis dans un fichier appelé `specification_analyse_conception_Nom1_Nom2_Nom3_Nom4.pdf` Il s'agit des noms des membres d'un même groupe.

Ce fichier doit contenir :

- **Une page de garde** : avec le titre du projet et les noms des membres
- **La description du sujet** : ce que doit faire cette application, décrire textuellement ces structures et ses services
- **La description textuelles des requêtes de mise à jour** (2 requêtes impliquant 1 table, 2 requêtes impliquant 2 tables, 2 requêtes impliquant plus de 2 tables)
- **La description textuelles des requêtes de suppression** (2 requêtes impliquant 1 table, 2 requêtes impliquant 2 tables, 2 requêtes impliquant plus de 2 tables)

- **La description textuelles des requêtes de consultation** (5 requêtes impliquant 1 table dont 1 avec un group By et une avec un Order By, 5 requêtes impliquant 2 tables avec jointures internes dont 1 externe + 1 group by + 1 tri, 5 requêtes impliquant plus de 2 tables avec jointures internes dont 1 externe + 1 group by + 1 tri)

- **Le dictionnaire de données MERISE.** Pour chaque entité décrire chacune des propriétés : Titre / description / format des données / type / Identifiant / contraintes

- **La description textuelles des associations** : Décrire textuellement les associations entre entités

- **La définition du Modèle Entité-Association MERISE** (en utilisant le logiciel Poweramc de SYBASE/SAP ou manuellement). Vous devez vous limiter à 10 entités maximum et 5 minimum. Vous devez ici prendre en compte les contraintes identifiées lors de la description du dictionnaire de données. Exemple de liens d'association pour deux entités A et B ayant une liaisons 1 : N ou N-M (exemple UN PILOTE ASSURE 0, 1 ou plusieurs VOL, un VOL est assuré par 1 et 1 PILOTE au plus)

- **La définition du modèle logique de Données ou schéma relationnel** (en utilisant le logiciel Poweramc de SYBASE/SAP ou manuellement) un schéma de données logique en respectant les contraintes d'intégrités d'entité (PRIMARY KEY), de domaine (CHECK, NOT NULL, ...) et de référence (REFERENCES / foreign key). Générable automatiquement avec POWERAMC si vous avez décrites au niveau MCD

- **Spécification des traitement avec des packages PLSQL** (Modèle de traitements) . Choisir parmi vos tables deux d'entres (A et B) elles sur lesquelles les fonctions suivantes vont être spécifiées puis implémentées :

Sur la table A, définir un package plsql ayant le nom de la dite table:

- ajouter une nouvelle occurrence à A : fonction AInsérer;
- supprimer une occurrence à A (Attention : les enregistrements liés dans B doivent aussi être supprimés) : fonction ASupprimer;
- modifier des informations sur de A : fonction AmodifierF1, AmodifierF2 (texte requêtes correspondantes plus haut);
- lister toutes les occurrences de A: fonction ALister;
- fournir le nombre total des occurrences de A : fonction ATotal ;
- Proposer aussi 3 fonctions avec des requêtes de consultation impliquant 2 ou 3 tables au moins (jointure, groupe, tri) : fonction Af1, Af2, Af3. f1, f2, f3 sont des noms à définir AmodifierF2 (texte requêtes correspondantes plus haut);

Sur la table B, définir un package plsql ayant le nom de la dite table :

- ajouter une nouvelle occurrence à B : fonction BInsérer
- supprimer une occurrence à B : fonction BSupprimer;

- modifier des informations sur de B : fonction BModifierF1, BModifierF2 (texte requêtes correspondantes plus haut);

- lister toutes les occurrences de B pour une occurrence de A donnée: fonction Blister

Nota : Seule la partie spécification de chaque package est nécessaire ici. bien choisir les paramètres des méthodes. Bien nommer les méthodes. Remplacer F1 à Fn par des noms appropriés.

- **Spécification des trigers**

Vous devez définir textuellement aux moins deux triggers. Ces triggers vous permettant devront vous permettre de gérer aux moins deux règles de gestions qui ne peuvent être prises en compte à travers le schéma de données.

1.3 Génération du schéma physique de données

Le schéma doit être généré dans un fichier de script ayant le nom suivant :

Schema_physique_NomProjet_Nom1_Nom2_Nom3_Nom4.sql

Nom1 à NomN correspondent aux membres du groupe.

Définir le schéma physique consiste à produire les ordres SQL de création des tables, indexes etc.. Vous pouvez générer ces tables en convertissant le schéma logique vers des ordres SQL avec POWERAMC ou manuellement. Toutefois, il faut privilégier POWERAMC.

Si vous avez une base de données Oracle locale, il faut créer un utilisateur Oracle si ce n'est déjà fait ou utiliser le compte Oracle qui vous a été fourni sur une base distante.

Cet utilisateur sera le propriétaire de tous les objets de votre application (des tables, indexes, packages, ...).

Connectez vous avec le compte Oracle choisi (local ou distant).

Exécutez ensuite le script suivant :

Schema_physique_NomProjet_Nom1_Nom2_Nom3_Nom4.sql pour créer tous les objets qui s'y trouvent.

Les actions peuvent être mises en œuvre pas à pas.

1.4 Insertion des lignes dans des tables

Les données doivent être rendues dans un fichier ayant le nom suivant :

Insertions_lignes_NomProjet_Nom1_Nom2_Nom3_Nom4.sql Nom1 à NomN correspondent aux membres du groupe.

Il s'agit d'effectuer manuellement des insertions de lignes dans chacune de vos tables. Insérer 10 à 20 lignes par tables. Bien gérer les contraintes d'intégrités (primary key, foreign, et check).

Connectez vous avec le compte Oracle choisi dans la section 3 (local ou distant).

Exécutez ensuite le script suivant :

Insertions_lignes_NomProjet_Nom1_Nom2_Nom3_Nom4.sql pour créer tous les objets qui s'y trouvent.

Les actions peuvent être mises en œuvre pas à pas.

1.5 Mise à jour et consultation des données

Les requêtes de mise à jour (modification, suppression) et de consultation doivent être rendues dans un fichier ayant le nom suivant :

Maj_Consultation_De_Donnees_NomProjet_Nom1_Nom2_Nom3_Nom4.sql

Nom1 à NomN correspondent aux membres du groupe.

Les requêtes de mise à jour (modification, suppression) et de consultation à écrire sont celles définies dans la section 2.

Connectez vous avec le compte Oracle choisi dans la section 3 (local ou distant).

Exécutez ensuite le script suivant :

Maj_Consultation_De_Donnees_NomProjet_Nom1_Nom2_Nom3_Nom4.sql pour créer tous les objets qui s'y trouvent.

Les actions peuvent être mises en œuvre pas à pas.

1.6 Définition et implémentation des packages PLSQL

Les programmes et les tests de chaque méthodes doivent être rendues dans un fichier ayant le nom suivant :

Package_plsql_NomProjet_Nom1_Nom2_Nom3_Nom4.sql

Nom1 à NomN correspondent aux membres du groupe.

Il s'agit définir les packages spécifications et d'implémenter le code des packages bodies introduits au chapitre 2.

Vous devez aussi proposer le code de test de chacun de ces packages.

Vous devez dans ce même fichier programmer les spécifications des deux triggers décrits dans le chapitre 2.

Connectez vous avec le compte Oracle choisi dans la section 3 (local ou distant).

Exécutez ensuite le script suivant :

Package_plsql_NomProjet_Nom1_Nom2_Nom3_Nom4.sql pour créer tous les objets qui s'y trouvent.

Les actions peuvent être mises en œuvre pas à pas.

1.7 Travail à rendre

Travail à rendre: un dossier contenant :

- Le fichier `specification_analyse_conception_Nom1_Nom2_Nom3_Nom4.pdf` contenant le résultat des activités proposées 2
- Le fichier `Schema_physique_NomProjet_Nom1_Nom2_Nom3_Nom4.sql` contenant les activités proposées dans la section 3
- Le fichier `Insertions_lignes_NomProjet_Nom1_Nom2_Nom3_Nom4.sql` contenant les activités proposées dans la section 4
- Le fichier `Maj_Consultation_De_Donnees_NomProjet_Nom1_Nom2_Nom3_Nom4.sql` contenant les activités proposées dans la section 5
- Le fichier `Package_plsql_NomProjet_Nom1_Nom2_Nom3_Nom4.sql` contenant les activités proposées dans la section 6