République Islamique de Mauritanie
Ministère de l'enseignement Supérieur
et de la Recherche Scientifique
Institut Supérieur du Numérique
Site : www.supnum.mr

الجمهورية الإسلامية الموريتانية
وزارة التعليم العالي والبحث العلمي
المعهد العالي للرقمية
الموقع الإلكتروني : www.supnum.m

Matière : SGBD 1 (DEV21)	**Session :** Examen Final	**Semestre :** S2
Responsable : Moussa Demba	**Date :** 17/6/2022	**Durée :** 1h30mn

Tous les documents et matériels électroniques sont interdits

On se donne la base de données suivante :

Etudiant(matricule, nom, prenom, moyGen)
Module(codeMod, nom, effectifMax, effectif, code_pole)
Prerequis(codeMod#, codeModPrereq#, noteMin)
Examen(codeEx, codeMod#, dateExamen, nbPresents)
Inscription(matricule#, codeMod#, date_inscription)
Resultats(matricule#, codeMod#, note)

L'attribut *moyGen* est la moyenne générale d'un étudiant, *effectifMax* est le nombre maximum d'étudiants autorisés à s'inscrire dans un module et *effectif* est le nombre total d'étudiants inscrits dans un module.
On dit qu'un module est saturé si son *effectifMax* = *effectif*.
La relation *Prerequis* exprime que pour s'inscrire dans le module *codeMod*, l'étudiant doit avoir compléter le module *codeModPrereq*. L'attribut *code_pole* dans Module prend l'une des valeurs : pôle de spécialité (PS), pôle de spécialité et technique (PST) ou pôle humanité et entreprise (PHE).

Répondre aux questions suivantes en utilisant des commandes SQL :

1. Ecrire une requête SQL qui affiche les *noms* et les *pôles* des modules pour lesquels l'étudiant 21099 n'a pas trouvé une note≥10 en utilisant le prédicat *EXISTS*.
2. Ecrire une requête qui affiche les codes des modules qui ont au moins 20 étudiants qui ont obtenu des notes ≥10.
3. Ecrire une requête qui affiche le nombre d'étudiants admis par module.
4. Compléter une procédure stockée ***Incr_effectif(codeM integer)*** qui permet d'augmenter l'effectif d'un module donné de 5 places si le module est saturé.
5. Ecrire une fonction ***checkPrerequis(codeM1, codeM2)*** qui retourne 1 si le module de code *codeM2* est le prérequis du module de code *codeM1* et 0 sinon.
6. Ecrire une vue ***v_moduleDisponible*** qui contient le code, nom et le nombre places restantes de chaque module non saturé.

7. Ecrire une vue ***v_examenPossible*** qui contient les codes des modules où on a au moins un étudiant inscrit dans ce module.
8. Ecrire une fonction ***checkInscription(matEt, codeMod)*** qui retourne 1 si l'étudiant numéro ***matEt*** est inscrit dans le module ***codeMod*** et 0 sinon. Donner un scenario d'exécution de la fonction avec les paramètres *21090* et '*DEV2'*.
9. Ecrire une vue ***v_exam*** qui affiche la liste des modules qui ont déjà fait des examens et une autre vue ***v_pas_exam*** pour ceux qui n'ont pas encore fait des examens.
10. Ecrire une procédure PL/SQL ***liste_modules(OUT nb1, OUT nb2)*** qui retourne le nombre de modules qui on fait des examens (respectivement qui n'ont pas encore fait des examens). Utiliser les vues créées précédemment et donner un exemple d'exécution de la procédure.
11. Créer une vue ***v_info*** qui donne le résultat suivant :

Code Module	Nom Module	Date Examen	Nombre de présents	Effectif	Nbre absents
DEV2	Programmation et développement	25/5/2022	57	75	18
SYR2	Architecture et systèmes	25/5/2022	63	75	12
MAI2	Outils mathématiques et informatiques	06/6/2022	72	75	3
DPR2	Développement personnel	10/6/2022	67	75	8

12. Ecrire une procédure PL/SQL ***calculMoyenne(matricule)*** qui calcule la moyenne générale d'un étudiant donné. Le calcul de la moyenne n'est possible que si l'étudiant a passé aux examens de tous les modules auxquels il est inscrit.
13. Créer une procédure qui permet de créer trois utilisateurs sur le serveur local: ***userPS***, ***userPST*** et ***userPHE***. L'utilisateur ***userPS*** ne peut accéder qu'aux modules de pôle PS, l'utilisateur ***userPST*** ne peut accéder qu'aux modules de pôle PST et l'utilisateur ***userPHE*** ne peut accéder qu'aux modules de pôle PHE. Attribuer à chaque utilisateur les privilèges de consulter ses modules.

- Bonne Chance -