

TP2 – Requêtes MySQL

Le fichier `definition-tables.sql` contient les définitions des tables suivantes :

- `departement(nom_dept, batiment, budget)` : les départements d’une université (américaine).
- `cours(id_cours, intitule, nom_dept, credits)` : les intitulés des cours, avec leur nombre d’ECTS et le département dont ils dépendent.
- `enseignant(ID, nom, nom_dept, salaire)` : les enseignants, avec leur no. d’identifiant.
- `salle_cours(batiment, no_salle, capacite)` : la table contenant les salles de cours avec leur caractéristiques.
- `filiere(id_cours, id_filiere, semestre, annee, batiment, no_salle, id_creneau)` : la liste des filières de l’université, avec tout leur emploi du temps codé dans les `id_creneau`.
- `resp_module(ID, id_cours, id_filiere, semestre, annee)` : les responsables de modules, pour chaque année.
- `etudiant(ID, nom, nom_dept, tot_cred)` : la liste de tous les étudiants (passés et présents) de cette université.
- `choix_option(ID, id_cours, id_filiere, semestre, annee, note)` : les choix des modules que les divers étudiants ont fait pendant leur cursus.
- `tuteur(ID_etud, ID_ens)` : les tuteurs de chaque étudiants.
- `creneau(id_creneau, jour, hr_debut, min_debut, hr_fin, min_fin)` : la table détaillant les créneaux horaires des cours.
- `prereq(id_cour, prereq_id)` : les prérequis (dépendances entre les cours).

Dans un premier temps vous allez importer le fichier `definition-tables.sql` dans votre base de données, puis le fichier `petites-tables.sql` et résoudre les questions suivantes. Dans un deuxième temps, vous allez réimporter `definition-tables.sql`, puis importer `grandes-tables.zip` (ce qui prendra un certain temps pour le faire !), et vous allez relancer les mêmes requêtes. Notez le temps de réponse à chaque fois.

Exercice 1: Questions basiques SQL :

1. Trouver tous les enseignants du département “Biology”.
2. Trouver les noms des cours de 3 ECTS du département d’info (“Computer Science” ou ressemblant à cela).
3. Pour l’étudiant dont le ID est 12345 afficher les identités (`id_cours`) et les intitulés des cours choisis.
4. Comme pour le point précédent, mais en affichant le nombre total de crédits pour tous les cours choisis par l’étudiant, mais sans afficher `tot_creds` (pourquoi ?).
5. Même question mais cette fois-ci produire le total des crédits pour *chacun* des étudiants, ensemble avec l’ID de chaque étudiant. Ne pas afficher les étudiants qui n’ont choisi aucun cours.
6. Trouver les noms des étudiants qui ont choisi un cours du département d’informatique (dont ressemblant à *Comp. Sci.*). Pas de duplicats.
7. Trouver les ID des enseignants qui n’ont jamais enseigné de cours.
8. Comme pour le point précédent, mais en affichant aussi les noms des enseignants.

Exercice 2: Questions SQL plus complexes sur la base de données de l’université américaine. Dans certains cas, on vous demande d’insérer des données dans certaines tables, afin de montrer certaines aspects.

1. Trouver le nombre maximal et minimal d’étudiants par filière, sans se soucier des filières qui n’ont pas eu d’étudiant.
2. Trouver les filières qui ont eu le nombre maximal d’étudiants, en affichant aussi ce nombre.
3. Comme pour le premier point, mais cette fois-ci afficher aussi les filières sans étudiant, en mettant 0 en tant que nombre d’étudiants pour ces filières.

Répondre à cette question de deux manières :

- (a) En utilisant une sous-requête.
 - (b) En utilisant une agrégation sur une jointure externe gauche.
4. Trouver tous les cours dont l'identifiant commence par "CS-1".
 5. Donner tous les enseignants qui ont enseigné dans les cours trouvés au point précédent.
 6. Insérer chaque enseignant en tant qu'étudiant dans son propre département, et avec *tot_creds* = 0.
 7. Maintenant supprimer tous ces "étudiants" fictifs, mais *sans* effacer les vrais étudiants qui ont *tot_creds* = 0.
 8. La valeur de l'attribut *tot_creds* peut ne pas être identique au nombre d'ECTS acquis par les étudiants. Mettre à jour cet attribut pour chaque étudiant.
On va considérer qu'un module est acquis si la note n'est pas "F".
 9. Modifier le salaire de chaque enseignant, en le mettant à 10000 fois le nombre de cours qu'ils ont enseigné.
-