

---

M1104: INTRODUCTION AUX BASES DE DONNÉES.

**Feuille de TP n°6**

*Requêtes conjonctives*

---

**Question 1.** Vous allez trouver sur le site <https://dbis-uibk.github.io/relax/calc.htm?lang=en#> un calculateur pour l'algèbre relationnelle. Nous vous proposons quelques requêtes à tester sur ce calculateur, pour mieux comprendre l'algèbre. Néanmoins, remarquer que le calculateur propose non seulement les opérateurs de l'algèbre relationnelle, mais aussi d'autres, construits sur les opérateurs de base ou avec un pouvoir d'expression plus grand que l'algèbre relationnelle. Nous allons nous restreindre aux opérations de SPJR.

Considérons la base UIBK-PS (à choisir via le bouton en haut à gauche). Écrire, en utilisant le calculateur, les requêtes ci-dessous. Vous allez exécuter les requêtes sur le calculateur. Pour chaque requête observez et répondez :

- Le calculateur montre un plan d'exécution pour la requête. En effet, le plan d'exécution d'une requête SQL est normalement exprimé via un arbre sur des opérateurs de l'algèbre relationnelle. Ce plan doit être analysé du bas vers le haut. L'arbre montre une proposition de l'ordre de l'exécution des différentes opérations présentes dans la requête. Pour chaque requête, analysez le plan en informant dans quel ordre le plan propose que les opérations soient effectuées.
- En Oracle, un *optimiser* "étudie" différents plans d'exécution (éventuellement en ré-écrivant la requête) pour choisir le moins cher. Pouvez vous proposer un plan d'exécution différent de celui proposé par le calculateur ?

**REQUÊTES**

1. Personnes (nom et age) qui fréquentent une pizzeria.
2. Personnes de plus de 20 qui fréquentent une pizzeria.
3. Nom et age des personnes fréquentant une pizzeria qui sert une pizza qu'ils mangent. Afficher aussi le nom de la pizza.
4. Nom et age des personnes fréquentant une pizzeria qui sert une pizza pepperoni.

**Question 2.** Soit une base de données qui stocke des informations sur les coureurs du Tour de France. Les clés primaires sont soulignées et les clés étrangères sont en italique.

EQUIPES [NomEquipe, DirecteurSportif]  
COUREURS [NumCoureur, NomCoureur, PrenomCoureur, *NomEquipe*, *CodePays*]  
ETAPES [NumEtape, DateEtape, VilleDépart, VilleArrivée, NbKm]  
TEMPS [*NumCoureur*, *NumEtape*, TempsRéalisé]  
PAYS [CodePays, NomPays]

**À partir des informations concernant le schéma ci-dessus, donner la liste de dépendances fonctionnelles qui doivent être valides.**

Les scripts SQL pour la création de la base et l'insertion des tuples est disponible sur l'ENT.

## Rappels :

- Les scripts SQL doivent avoir l'extension `.sql` et ils s'exécutent en entrant la commande `@ NomDuScript`.
- Vérifier que le schéma de la relation est bien créé avec la commande `describe <Nomtab>`.
- Vérifier l'instance de la base de données en regardant les instances de chaque table avec `select * from <NomTable>`. Vous pourrez vérifier le résultat de vos requêtes en regardant ces valeurs.

Écrire les requêtes suivantes en SQL :

1. Lister les noms des équipes.
2. Lister les noms des directeurs sportifs.
3. Lister les noms et les prénoms des coureurs.
4. Trouver l'équipe du coureur ULLRICH.
5. Donner les noms des coureurs de l'équipe COFIDIS.
6. Lister tous les coureurs français.
7. Quel temps le coureur 'JALABERT' a fait dans l'étape 3 ?
8. Trouver le temps réalisé par les coureurs sous l'encadrement de *Manolo SAIZ* dans les étapes dont la ville de départ est *Rouen*.
9. Lister les pays ayant des coureurs dans l'étape dont l'arrivée est *Plumelec*.
10. Trouver des couples de coureurs du même pays.

Écrire les requêtes suivantes en SQL :

1. Lister tous les coureurs français. Afficher (en concaténant) prénom et nom. La colonne doit afficher Coureur.
2. Afficher les coureurs dont les prénoms commencent par J.
3. Afficher les noms des coureurs en ordre alphabétique.
4. Afficher les résultats des étapes ordonnés par le numéro de l'étape, le temps réalisé et le nom du coureur.

**Question 3.** Considérez le schéma de base de données suivant :

```
CLIENT [ IdCli, NomCli, PrenomCli, Adresse ]
PRODUIT [ Ref , NomProd, Prix ]
COMMANDE [ NumCom, DateCom, IdCli ]
DETAILCOMMANDE [ NumCom, Ref, Quantite ]
```

Vous trouverez sur ENT des scripts SQL pour créer les tables correspondant à la BD décrite par le schéma ci-dessus et pour créer une instance de la base de données, c'est-à-dire, pour insérer des tuples dans les tables.

Remarquer qu'aucune contrainte a été implémentée. Que cela signifie ? Pensez-vous intéressant d'ajouter des contraintes ? Proposez-en quelques-unes et implantez-les en modifiant le script de création des tables. Vérifiez-les avec des exemples.

Écrire les requêtes suivantes en SQL :

1. Lister les noms des clients.

2. Lister les noms et les prénoms des clients, dans l'ordre alphabétique de nom.
3. Lister les noms de produits et leur prix dans l'ordre décroissante de prix.
4. Donner la liste des villes (sans répétition) où la société a des clients.
5. Donner les noms des clients d'Orléans.
6. Donner la liste des produits valant plus de 2 Euros.
7. Donner pour chaque commande le nom du client.
8. Donner le détail (NumCom, Ref, NomProd, Quantite, Prix ) des commandes passées le 10 août 2013.
9. Donner pour chaque client le nom des produits qu'il a commandés. Organiser le résultat par ordre alphabétique du nom et prénom de client et, ensuite, par nom du produit.
10. Afficher les lignes de commandes, classées par date et numéro de commande. Pour chaque ligne, on veut les informations du produit ainsi que le total de la ligne.
11. Classer les commandes par date en suivant le format 'YYYY/MON/DD' . Afficher la date (avec nom de la colonne DateCommande), le numéro de la commande et le nom du Client.
12. Même requête (11) mais avec le nom du jour de la semaine en plus.
13. Donner l'heure d'enregistrement des commandes de Marcel Auger.
14. Donner les commandes qui ont été enregistrées il y a plus de 3 ans. Afficher le numéro de la commande et le nombre d'années de différence par rapport à la date d'aujourd'hui.
15. Donner les couples de clients habitants dans la même ville.
16. Donner le numéro des commandes ayant au moins trois produits différents achetés.
17. Liste de clients dont le nom contient le motif 'Du'.