

M1 Informatique  
Bases de données avancées  
Contrôle terminal de session 1  
7 mai 2020

## PARTIE 1

### Triggers

Écrire le ou les triggers nécessaires à réaliser les actions suivantes (il faut écrire aussi bien les fonctions PL/pgSQL qui implémentent les triggers, que les commandes `create trigger`.)

**Question 1** Soit le schéma de base de données suivant

Compte (**no\_compte**, nom\_agence)  
Titulaires (**nom\_client**, **no\_compte**)  
Agence\_client (**nom\_agence**, **nom\_client**, nombre)

La clef de chaque table est en gras.

Écrire un trigger pour imposer automatiquement une contrainte référentielle entre la table Titulaires et la table Compte : quand un compte est effacé da la table Compte, effacer les lignes de la table Titulaires qui font référence à ce compte.

**Question 2** Avec référence au schéma de base de données de la Question 1, remarquer qu'un compte peut avoir plusieurs titulaires (comptes joints). Nous voulons que la table Agence\_client contienne à tout moment, pour chaque client et chaque agence, le nombre de comptes dont le client est titulaire qui sont gérés par cette agence, si ce nombre est supérieur à zéro.

Ecrire le trigger qui met à jour automatiquement la table Agence\_client après chaque insertion dans la table Titulaires.

**Question 3** Soit le schéma de base de données suivant :

Enseigne (**id\_enseignant**, **id\_cours**)  
EDT (**id\_cours**, **jour**, salle, horaire\_debut, horaire\_fin)

Écrire un trigger qui empêche toute modification de la table EDT (INSERT ou UPDATE) qui impliquerait qu'un même enseignant se trouve en même temps dans deux salles différentes. Remarquer qu'un enseignant devrait se trouver en même temps dans deux salles différentes s'il enseigne deux cours prévus le même jour dans deux salles différentes, à des créneaux horaires qui se superposent.

**Question 4** Soit le schéma de base de données suivant :

Cours(**id-cours**, nom-cours, credits)  
Etudiant(**num-etu**, nom, tot-credits)  
Notes (**num-etu**, **id-cours**, note)

La note d'un examen est comprise entre 0 et 20, ou bien elle a valeur NULL. On suppose qu'un étudiant a le droit de repasser un examen plusieurs fois, mais la table Notes décrit uniquement la dernière note obtenue (elle est mise à jour chaque fois qu'une nouvelle note est obtenue).

Écrire le ou les triggers nécessaires à calculer automatiquement les crédits obtenus par les étudiants. Plus précisément, à chaque insertion dans la table Notes, ainsi qu'à chaque modification de l'attribut Notes.note, le trigger exécute les actions suivantes : si la nouvelle note est supérieure ou égale à 10 et l'ancienne note était inférieure à 10, ou absente (note absente veut dire note avec valeur NULL, ou bien trigger déclenché par une insertion), le nombre de crédits du cours validé est ajouté au total des crédits obtenus par l'étudiant (attribut Etudiant.tot-credits).

## PARTIE 2

### Normalisation

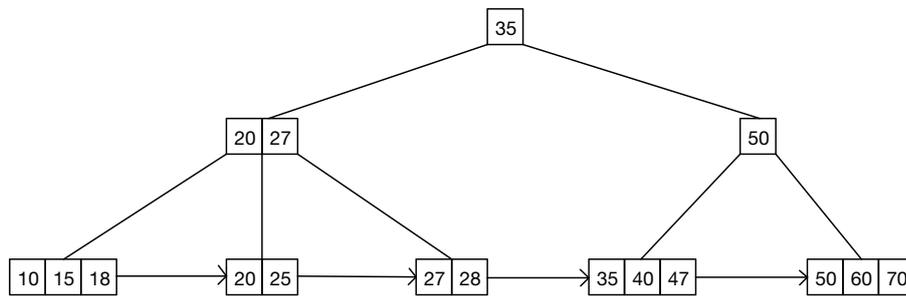
**Question 5** Soit  $R(A, B, C, D, E, F, G)$  un schéma de relation avec dépendances fonctionnelles :

$\mathcal{F} = \{$   
 $ACE \rightarrow D$   
 $BE \rightarrow F$   
 $C \rightarrow G$   
 $FE \rightarrow D$   
 $D \rightarrow E$   
 $AC \rightarrow B$   
 $B \rightarrow AG\}$

- Donner toutes les clefs de  $R$ .
- Trouver une décomposition de  $R$  en *forme normale de Boyce-Codd* sans perte d'information. Est-ce que cette décomposition préserve toutes les dépendances fonctionnelles ?
- Trouver une décomposition de  $R$  en *troisième forme normale* sans perte d'information et sans perte de dépendances fonctionnelles. (Se rappeler que il est nécessaire d'abord de minimiser  $\mathcal{F}$ , c'est-à-dire d'abord minimiser toutes les parties gauches, et ensuite éliminer toutes les dépendances fonctionnelles redondantes.)

## Indexation

**Question 6** Considérer l'arbre  $B^+$  suivant



Supposer que chaque bloc du disque peut contenir au plus 3 clefs et 4 pointeurs.

Supposer que les opérations suivantes sont effectuées dans l'ordre :

- insertion de la clef 48 ;
- insertion de la clef 75 ;
- insertion de la clef 65 ;
- insertion de la clef 68 ;
- suppression de la clef 20 ;
- suppression de la clef 25 ;
- suppression de la clef 70 ;

Combien de noeuds a l'arbre après chacune de ces opérations ? (Inutile de montrer l'arbre, répondre en donnant juste sept nombres, un pour chaque point de a) à g)).