

## Bases de données : REQUÊTES de MISE À JOUR => CORRECTION

On considère une base de données contenant trois tables x y et z dont une vue est présentée ci-dessous.

table x	
a	b
1	1
2	2
3	2
4	2
5	1
6	9
7	1

table y	
c	d
9	9
10	10
11	9
12	20
13	30
14	9
15	1
16	10
17	10

table z		
a	c	e
1	11	30
2	14	9
5	15	1
7	17	3
1	10	50
2	9	8
2	15	15
3	17	19
4	16	12
5	10	20
2	11	30
7	14	9
7	9	12

Dans la table x : **a** est une clé primaire

Dans la table y : **c** est une clé primaire

Dans la table z : **a** est une clé étrangère qui fait référence à x(a)  
**c** est une clé étrangère qui fait référence à y(c)

Pour chacune des requêtes ci-dessous, dire si elle réussit ou si elle échoue. Si elle réussit, donner l'état final de la table ; si elle échoue, expliquer pourquoi.

On suppose que la base de donnée est toujours dans l'état représenté ci-dessus au début de chacune des requêtes :

1. UPDATE x SET b = b+a ;

2. UPDATE x SET a = b+a ;

applique à TOUTES les lignes de la table x la transformation de la valeur b en b+a

**essaie** d'appliquer à TOUTES les lignes de la table x la transformation de la valeur a en

table x

a	b
1	2
2	4
3	5
4	6
5	6
6	15
7	8

a+b

table x

a	b
2	1
4	2
5	2
6	2
6	1
15	9
8	1

a est une clé primaire donc il est impossible que deux lignes contiennent la même valeur pour ce champ => la requête échoue et aucune modification n'est effectuée !

3. INSERT INTO x(a,b) VALUES (5,5) ;

**échoue** car une ligne contient déjà la valeur 5 pour le champ **a** qui est une **clé primaire**

=> duplicata **impossible**

4. INSERT INTO x(a,b) VALUES (8,8) ;

Possible : ajoute simplement une nouvelle ligne à la table x

5. INSERT INTO y(c,d) VALUES (8,8) ;

Possible : ajoute simplement une nouvelle ligne à la table y

6. INSERT INTO z(a,c,e) VALUES (1,17,1) ;

possible car :

- il n'y a pas de clé primaire sur z (rien n'empêche une valeur de figurer deux fois dans une même colonne)
- la valeur 1 du champ **a** figure bien dans la table x
- la valeur 17 du champ **c** figure bien dans la table y
- les contraintes de clé étrangère sont satisfaites

=> ajoute une ligne à la table z.

7. INSERT INTO z(a,c,e) VALUES (1,18,1) ;

la valeur 18 ne figure pas dans la colonne **c** de la table y => la contrainte de clé étrangère de z, qui fait référence à y(**c**) empêche l'exécution de la requête.

=> la requête échoue, la table z n'est pas modifiée.

8. DELETE FROM x WHERE a = 5 ;

le champ **a** de la table x est référencé par la clé étrangère de z.

Puisque z contient des lignes pour lesquelles **a** vaut 5, cette contrainte de clé étrangère empêche la suppression de la ligne correspondante dans la table x

=> la requête échoue, x n'est pas modifiée

9. DELETE FROM x WHERE a = 6 ;

le champ **a** de la table x est référencée par la clé étrangère de z.

mais la colonne a de z ne contient pas la valeur 6 : cette contrainte de clé étrangère n'empêche pas de supprimer la ligne où a=6 dans x

=> la requête est exécutée, et la ligne où a = 6 est supprimée de la table x

10.DELETE FROM z WHERE e < 4 ;

=> la requête est exécutée, et deux lignes de la table z sont supprimées.