

TD2 BDR

Définition de schémas

Exo 1 - Université

ETUDIANT

Nom	NumeroEtudiant	Classe	Dominante
-----	----------------	--------	-----------

COURS

NomCours	NumeroCours	HeuresCredit	Departement
----------	-------------	--------------	-------------

PREREQUIS

NumeroCours	NumeroPrerequis
-------------	-----------------

UV

IdUV	NumeroCours	Trimestre	Annee	Enseignant
------	-------------	-----------	-------	------------

NOTE

NumeroEtudiant	IdUV	Note
----------------	------	------

© Pearson Education France

Q0: Écrivez les requêtes suivantes en SQL

Q0-1: Extraire les cours dispensés par le professeur Crémant en 1998 et 1999, avec une sous- requête (sans jointure!)

Q0-2: Afficher le nom et la dominante de tous les étudiants qui ont obtenu un A à tous les cours.

Q0-3: Afficher le nom et la dominante de tous les étudiants qui ont n'ont obtenu de A à aucun cours.

Q1: Examinez la BDD dont le schéma est présenté à gauche.

Q1-1: Soulignez les clés primaires.

Q1-2: Quelles sont les contraintes d'intégrité référentielles qui devraient porter sur le schéma?

Q1-3: Complétez les instructions SQL pour définir cette BDD: contraintes NOT NULL, clés primaires, clés étrangères

```
CREATE TABLE ETUDIANT (  
  Nom VARCHAR(30) NOT NULL,  
  NumeroEtudiant INTEGER  
  Niveau INTEGER  
  Dominante CHAR(4)
```

);

```
CREATE TABLE COURS (  
  NomCours VARCHAR(30)  
  NumeroCours CHAR(8)  
  HeuresCredit INTEGER  
  Departement CHAR(4)
```

```
CREATE TABLE NOTE (  
  NumeroEtudiant INTEGER  
  IdUV INTEGER  
  Note CHAR
```

```
CREATE TABLE PREREQUIS (  
  NumeroCours CHAR(8)  
  NumeroPrerequis CHAR(8)
```

);

Q2: Écrivez des instructions de modification

```
INSERT INTO table_name [ AS alias ] [ ( column_name [, ...] ) ]  
{ DEFAULT VALUES | VALUES ( { expression | DEFAULT } [, ...]  
) [, ...] | query }
```

```
DELETE FROM [ ONLY ] table_name [ * ] [ [ AS ] alias ]  
[ USING using_list ]  
[ WHERE condition | WHERE CURRENT OF cursor_name ]
```

```
UPDATE [ ONLY ] table_name [ * ] [ [ AS ] alias ]  
SET { column_name = { expression | DEFAULT } |  
      ( column_name [, ...] ) = ( { expression | DEFAULT } [, ...] ) |  
      ( column_name [, ...] ) = ( sub-SELECT )  
      } [, ...]  
[ FROM from_list ]  
[ WHERE condition | WHERE CURRENT OF cursor_name ]
```

Q2-1: Insérer un nouvel étudiant, <'Jeannin', 25, 1, 'MATH'>.

Q2-2: Insérer une ligne violant une contrainte de clé primaire

Q2-3: Insérer un nouveau cours, <'Ingénierie cognitive', 'CS4390', 3, 'INFO'>.

Q2-4: Supprimer l'enregistrement de l'étudiant dont le nom est « Schmidt » et le numéro 17

Q2-5: Insérer une ligne violant une contrainte de référentielle

Suite de l'exo 1

Enregistrements présents dans la base de données

Q3: Vérification de la cohérence de la base

Pour chacune des questions, écrivez des requêtes SQL pour détecter le(s) conflit(s) existant(s).

ETUDIANT	Nom	NumeroEtudiant	Niveau	Dominante
	Schmidt	17	1	INFO
	Brun	8	2	INFO

Q3-1: Est-ce que la table Cours respecte les contraintes d'intégrités?

Q3-2: Est-ce que la table Notes respecte les contraintes d'intégrités?

Q3-3: Est-ce que la table Prerequis respecte les contraintes d'intégrités?

COURS	NomCours	NumeroCours	HeuresCredit	Departement
	Introduction à l'informatique	CS1310	4	INFO
	Structures de données	CS3320	4	INFO
	Mathématiques discrètes	CS1310	3	MATH
	Bases de données	CS3380	3	INFO

UV	IdUV	NumeroCours	Trimestre	Annee	Enseignant
	85	MATH2410	4	98	Crémant
	92	CS1310	4	98	André
	102	CS3320	3	99	Queneau
	112	MATH2410	3	99	Chenet
	119	CS1310	3	99	André
	135	CS3380	3	99	Segonzac

Q4: Garder la cohérence de la base

Pour chacune des questions ci-dessous, écrivez les requêtes SQL nécessaires pour garder la cohérence de la base.

Q4-1: Remplacez l'identifiant de l'étudiant Schmidt avec la valeur "8". Est-ce possible? Quelle requête devons nous écrire pour nous assurer de la validité de cette opération?

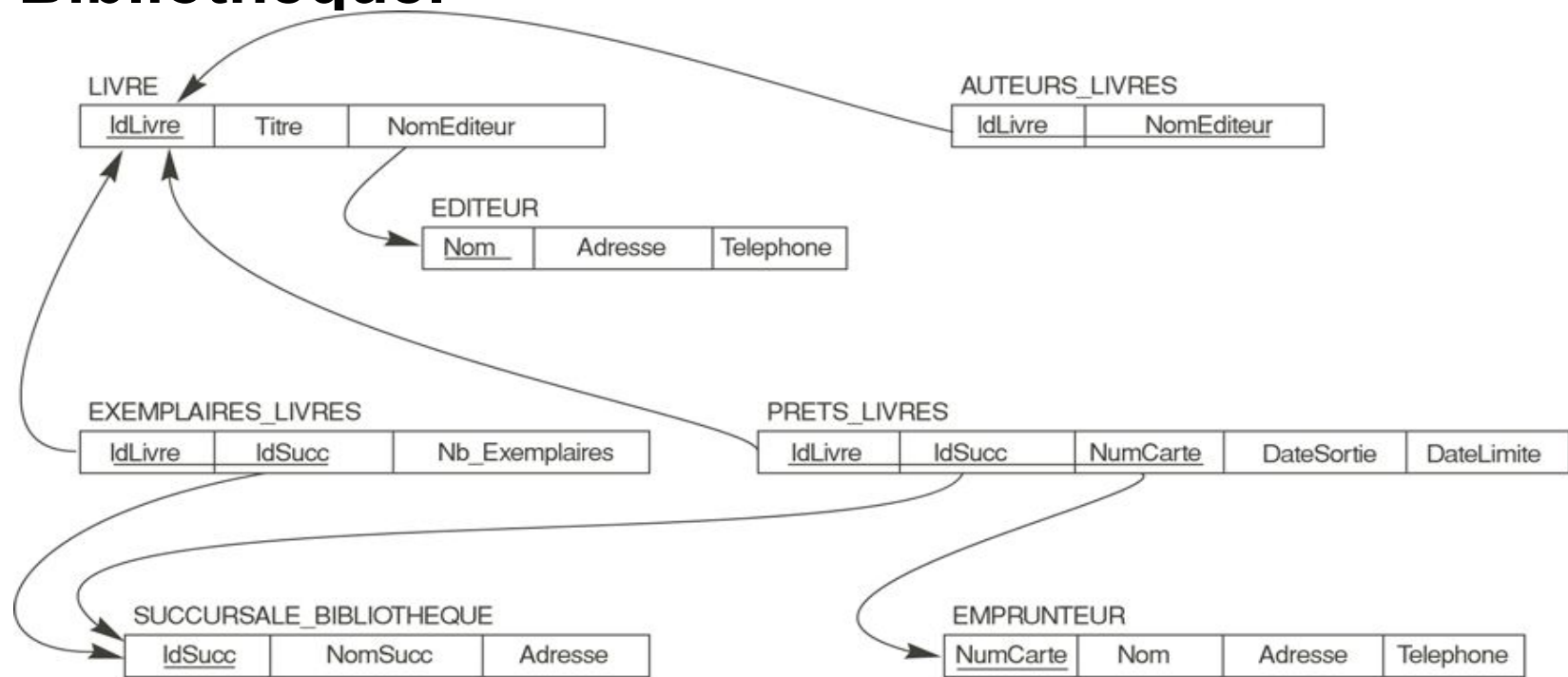
Q4-2: Remplacez l'identifiant de l'étudiant Schmidt avec la valeur "9".

Q4-3: Le cours "Structures de données" ne sera plus enseigné l'année prochaine. Supprimez-le de la base.

NOTE	NumeroEtudiant	IdUV	Note
	17	112	B
	17	119	C
	8	85	A
	8	92	A
	71	102	B
	71	135	A

PREREQUIS	Numero Cours	Numero Prerequis
	CSS3380	INFO3320
	CSS3380	MATH2410
	CSS3380	INFO1310

Exo 2 - Bibliothèque.



© Pearson Education France

Q1: Examinez le schéma relationnel BIBLIOTHEQUE.

Q1-1: Comment assurer que la base reste cohérente après des insertions, modifications et suppressions?

Q1-2: Choisissez l'action appropriée (rejet, cascade, définir à null, définir à la valeur par défaut) pour chacune des contraintes d'intégrité référentielle, aussi bien dans le cas de la suppression d'un tuple référencé que dans celui de la mise à jour de la valeur d'un attribut d'une clé primaire dans un tuple référencé. Justifiez vos choix.

Q2: Compétez les instructions LDD SQL (au verso) pour la déclaration du schéma relationnel de la base de données BIBLIOTHEQUE au dos de cette feuille.
Spécifiez les clés et les actions automatiques adéquates.

CREATE TABLE LIVRE (*IdLivre* CHAR(20) NOT NULL,
Titre VARCHAR(30) NOT NULL,
NomEditeur VARCHAR(20),
PRIMARY KEY (_____),
FOREIGN KEY (_____) REFERENCES EDITEUR (_____) ON _____
);

CREATE TABLE AUTEURS_LIVRES (
IdLivre CHAR(20) NOT NULL,
NomAuteur VARCHAR(30) NOT NULL,
PRIMARY KEY (IdLivre, NomAuteur),
FOREIGN KEY (_____) REFERENCES LIVRE (_____)
ON _____);

CREATE TABLE EDITEUR (
Nom VARCHAR(20) NOT NULL,
Adresse VARCHAR(40) NOT NULL,
Phone CHAR(12),
PRIMARY KEY (_____));

CREATE TABLE EXEMPLAIRES_LIVRES (
IdLivre CHAR(20) NOT NULL,
IdSucc INTEGER NOT NULL,
Nb_Exemplaires INTEGER NOT NULL,
PRIMARY KEY (_____, _____),
FOREIGN KEY (_____) REFERENCES LIVRE (_____)
ON _____,
FOREIGN KEY (_____) REFERENCES _____ (_____)
ON _____);