

Sciences de l'informatique - Base de données

Correction Epreuve de contrôle - Session de juin 2011

Exercice 1 : (3 points)

Alias : Un nom attribué à une colonne ou à une table et qui est différent du nom d'origine.

Indépendance entre programmes et données : Les données sont décrites séparément des traitements exécutés sur ces données.

Données structurées : Les données sont réparties en lignes et colonnes selon une structure bien définie.

Projection : L'opération d'extraction d'un sous-ensemble des colonnes d'une table.

Non redondance des données : Une même donnée ne sera pas répétée plusieurs fois dans la base de données.

Sélection : L'opération d'extraction d'un sous-ensemble des lignes d'une seule table.

Exercice 2 : (3 points)

1. Etude des règles de gestion :

- (1) Un groupe est composé de plusieurs élèves et un élève appartient à un seul groupe.
- (2) Un groupe réalise un seul projet et un projet est réalisé par un ou plusieurs groupes.
- (3) Dans chaque groupe, un élève est désigné comme chef de groupe.

a) La contrainte n°1, signifie que :

- Etant donné un élève (**NumElev**), il ne peut appartenir qu'à un seul groupe (**NumGrp**).
- A l'inverse, étant donné un groupe (**NumGrp**), il peut accueillir plusieurs élèves (**NumElev**)

→ Dans la table fille **Eleve**, ajout de **NumGrp**, en référence aux valeurs de la colonne **NumGrp** de la table mère **Groupe**).

b) La contrainte n°2, signifie que :

- Etant donné un groupe (**NumGrp**), il ne peut réaliser qu'un seul projet (**NumProj**).
- A l'inverse, étant donné un projet (**NumProj**), il peut être réalisé par plusieurs groupes (**NumGrp**).

→ Dans la table fille **Groupe**, ajout de **NumProj**, en référence aux valeurs de la colonne **NumProj** de la table mère **Projet**).

c) La contrainte n°3, signifie que :

- Etant donné un groupe (**NumGrp**), il doit être dirigé par un élève qui sera désigné comme chef (**NumElevChefGrp**) de ce groupe.
- A l'inverse, étant donné un élève (**NumElev**), il n'est pas systématiquement nommé chef d'un groupe (**NumElevChefGrp**).

→ Dans la table fille **Groupe**, ajout de **NumElevChefGrp**, en référence aux valeurs de la colonne **NumElev** de la table mère **Eleve**).

ELEVE (NumElev, Nom, Prenom, DateNais, Sexe, **NumGrp#**)
GROUPE (NumGrp, NomGrp, **NumProj#**, **NumElevChefGrp#**)
PROJET (**NumProj#**, NomProj)

2. Si un groupe peut réaliser différents projets, alors :
- Pour un groupe donné (**NumGrp**), il existe plusieurs projets (**NumProj**).
 - Pour un projet donné (**NumProj**), il peut lui correspondre plusieurs groupes (**NumGrp**).

→ Il s'agit d'une relation n – n. Il faut créer une table supplémentaire.

Rappel : La transformation d'une relation n – n en table, se fait en appliquant les règles suivantes :

- Les colonnes de la nouvelle table (**Realiser**) sont les clés primaires (**NumGrp** et **NumProj**) des tables (**Groupe** et **Projet**) de la relation. Si cette relation est porteuse de données, les mettre également sous forme de colonne.
- La clé de la nouvelle table est obtenue par la **concaténation des clés** des tables de la relation.

En appliquant la méthode sur l'exemple, on obtient la représentation textuelle suivante :

ELEVE (NumElev, Nom, Prenom, DateNais, Sexe, **NumGrp#**)
GROUPE (NumGrp, NomGrp, **NumElevChefGrp#**)
PROJET (NumProj, NomProj)
REALISER (NumProj#, NumGrp#)

Exercice 3 : (7 points)

1) ALTER TABLE Connexion MODIFY PrixCnx DECIMAL(6,3) ;

Ou bien

ALTER TABLE CONNEXION DROP COLUMN PrixCnx ;

ALTER TABLE CONNEXION ADD COLUMN PrixCnx DECIMAL(6,3) ;

2) UPDATE Connexion SET PrixCon = 0.9 * PrixCnx

WHERE DebitCon <= 512 ;

3) SELECT NomAg, AdrAg, TelAg FROM Agence

WHERE VilleAg = 'Bizerte' Order by 1 DESC ;

4) SELECT C.NumTelCI, NumCIN, NomCI FROM Client C, Abonnement A

WHERE C.NumTelCI = A.NumTelCI

AND PeriodeAb = 6;

5) SELECT COUNT(*) FROM Abonnement Ab, Agence Ag

WHERE Ab.CodeAg = Ag.CodeAg

AND NomAg = 'ACTEL';

6) DELETE * FROM Connexion

WHERE (DebitCnx = 128) OR (DebitCnx = 256);

On accepte : DebitCnx IN (128,256) ; (DebitCnx between 128 AND 256);

7) ALTER TABLE Abonnement ADD CONSTRAINT PeriodeAn

CHECK (PeriodeAb BETWEEN 1 AND 12);

Exercice 4 : (7 points)

1) Liste des colonnes:

Nom colonne	Description	Type de données	Taille	Sujet
CodeAct	Code de l'activité	Texte	8	Activite
NomAct	Nom de l'activité	Texte	20	Activite
NumSalle	Numéro de la salle	Texte	4	Activite
MatEns	Matricule de l'enseignant	Texte	10	Enseignant
NomEns	Nom de l'enseignant	Texte	20	Enseignant
PrenomEns	Prenom de l'enseignant	Texte	20	Enseignant
CodeSpec	Code de la spécialité de l'enseignant	Texte	5	Specialite
LibSpec	Libellé de la spécialité de l'enseignant	Texte	20	Specialite
NumIns	Numéro de l'inscription de l'élève	Texte	5	Eleve
NomElev	Nom de l'élève	Texte	20	Eleve
PrenomElev	Prénom de l'élève	Texte	20	Eleve
DateNaisElev	Date naissance de l'élève	Date		Eleve
ClassElev	Classe de l'élève	Texte	5	Eleve
NumSalle	Numéro de la salle	Texte	4	Salle
EtagSalle	Étage de la salle	Texte	10	Salle
BlocSalle	Bloc de la salle	Texte	10	Salle
CapSalle	Capacité de la salle	Numérique	2	Salle

2) Liste des tables:

Nom	Description	Sujet
Activite	Regroupe l'ensemble des activités scientifiques et culturelles du lycée	Activite
Enseignant	Regroupe l'ensemble des enseignants qui animent des activités	Enseignant
Eleve	Regroupe l'ensemble des élèves inscrits dans des activités	Eleve
Salle	Regroupe l'ensemble des salles du lycée	Salle
Specialite	Regroupe toutes les spécialités des enseignants	Specialite

3) Liens entre les tables :

Table mère	Table fille	Clé primaire	Clé étrangère
Activite	Enseignant	CodeAct	CodeAct
Specialite	Enseignant	CodeSpec	CodeSpec
Activite	Eleve	CodeAct	CodeAct
Activite	Salle	NumSalle	NumSalle

4) Représentation textuelle.

Activite (**CodeAct**, NomAct, NumSalle#)

Enseignant (**MatEns**, NomEns, PrenomEns, CodeSpec#, CodeAct#)

Specialite (**CodeSpec**, LibSpec)

Eleve (**NumIns**, NomElev, PrenomElev, DateNaisElev, ClassElev, CodAct#)

Salle (**NumSalle**, EtagSalle, BlocSalle, CapSalle)