

Compte Rendu

TP1-Administration des Bases de Données sous SQL Server

BOUKROUH Insaf

DAOUDI Wissal

- 1- Création de la base de données « Evaluation » en configurant les groupes de fichiers GF1 (F1, F2) et GF2 (F3, F4) en configurant GF1 un groupe de fichiers par défaut

Dans la fenêtre « Properties » :

On spécifie Les file groups GF1 et GF2

Dans la fenêtre « Explorer » :

Databases -> New Database

Une nouvelle fenêtre apparaît.

Dans l'onglet « General », dans le champ Name on saisit « Evaluation ».

Dans l'onglet « Filegroups », on affecte les fichiers F1, F2, F3 et F4 à leurs groupes respectifs.

- 2- Création des tables « Etudiant » et « Matière » sur le groupe GF1 et la table « Examen » sur le groupe GF2

Dans la fenêtre « Explorer » :

Databases -> Evaluation -> Tables -> New Table

Une nouvelle fenêtre apparaît.

On saisit les noms des colonnes avec leurs types. Au moment de sauvegarde, on nomme la table.

- 3- La table Examen sera stockée sur quels fichiers : **F3 et F4**
- 4- Spécification des contraintes :
 - Le champ Code de la table Etudiant est une clé primaire

Dans la fenêtre « Explorer » :

Databases -> Evaluation -> Tables -> Etudiant -> Code -> Modify

Une nouvelle fenêtre apparaît. Dans la barre en haut à gauche, on clique sur la cellule correspondant à notre clé.

- Le champ Code de la table Matière est une clé primaire

Dans la fenêtre « Explorer » :

Databases -> Evaluation -> Tables -> Matière-> Code -> Modify

Une nouvelle fenêtre apparaît. Dans la barre en haut à gauche, on clique sur la cellule correspondant à notre clé.

- Le champ Code de la table Examen est une clé primaire

Dans la fenêtre « Explorer » :

Databases -> Evaluation -> Tables -> Examen-> Code -> Modify

Une nouvelle fenêtre apparaît. Dans la barre en haut à gauche, on clique sur la cellule correspondant à notre clé.

- Le champ Code_Etud de la table Examen est une clé étrangère qui référence le champ Code de la table Etudiant avec configuration d'une mise à jour en Cascade

Dans la fenêtre « Explorer » :

Databases -> Evaluation -> Tables -> Examen -> Keys -> New Foreign Key

Une nouvelle fenêtre apparaît. Dans le volet :

- « Général », on clique sur « Tables And Columns Specification » et une nouvelle fenêtre s'ouvre dans laquelle on va définir la clé étrangère et la clé primaire puis on clique sur « OK ».
 - « Identity », on entre le nom « FK_Examen_Etudiant »
 - « Table Designer », on clique sur « INSERT and UPDATE Specification ».
- Ensuite, Update Rule -> Cascade et Delete Rule -> Cascade.**

- Le champ Code_Mat de la table Examen est une clé étrangère qui référence le champ Code de la table Matière avec configuration d'une mise à jour en Cascade

Dans la fenêtre « Explorer » :

Databases -> Evaluation -> Tables -> Examen -> Keys -> New Foreign Key

Une nouvelle fenêtre apparaît. Dans le volet :

- « Général », on clique sur « Tables And Columns Specification » et une nouvelle fenêtre s'ouvre dans laquelle on va définir la clé étrangère et la clé primaire puis on clique sur « OK ».
 - « Identity », on entre le nom « FK_Examen_Matière »
 - « Table Designer », on clique sur « INSERT and UPDATE Specification ».
- Ensuite, Update Rule -> Cascade et Delete Rule -> Cascade.**
- Date_Ex doit être inférieure ou égale à la date système

On utilise « Foreign Key Relationships » et on impose la règle

« Date_Ex<getdate() »

- La valeur du champ Note doit être entre 0 et 20

On utilise « Foreign Key Relationships » et on impose la règle « Note>=0 and Note<=20 »

5- Insertion des étudiants

USE Evaluation

**INSERT INTO Etudiant VALUES (1,'ABIDI','AHMED','2GL-G1'),
(2,'BENBRAHIM','SAID','2GL-G3'),(3,'BOUSOUF','HALIMA','2GL-G1') GO**

6- Insertion des matières

USE Evaluation

**INSERT INTO Matière VALUES ('M2.1','Réseaux Info',3),
('M2.2 ','Administration BD',3),('M2.3 ','Communication',2) GO**

7- Insertion d'un examen pour chaque étudiant dans chaque matière

USE Evaluation

INSERT INTO Examen VALUES

```
(1,'M2.1', '2016-04-18', 17),  
(1,'M2.2','2016-04-19', 18),  
(1,'M2.3', '2016-04-20', 14),  
(2,'M2.1', '2016-04-18', 5),  
(2,'M2.2', '2016-04-19', 9),  
(2,'M2.3','2016-04-20', 11),  
(3,'M2.1', '2016-04-18', 20),  
(3,'M2.2', '2016-04-19', 19),  
(3,'M2.3', '2016-04-20', 19);  
GO
```

8- Tests d'insertions

- INSERT INTO Etudiant VALUES (12,'BENALI','SALMA','2GL-G3')
⇒ **Effectuée**
- INSERT INTO Etudiant VALUES (13,'MOURAD','OUSSAMA','2GL-G2')
⇒ **Effectuée**
- INSERT INTO Etudiant VALUES (2,'TOUHAMI','BADR','2GL-G1')
⇒ **Echouée, Violation of Primary Key Constraint**
- INSERT INTO Examen (Code_Etud,Code_Mat,Note) VALUES
(12,'M2.1',15)
⇒ **Effectuée**
- INSERT INTO Examen (Code_Etud,Code_Mat,Note) VALUES
(12,'M2.5',18)
⇒ **Echouée, Violation de la contrainte « FK_Examen_Matière »**
- INSERT INTO Examen (Code_Etud,Code_Mat,Note) VALUES
(12,'M2.2',21)
⇒ **Echouée, Violation de la contrainte « Note»**
- INSERT INTO Examen (Code_Etud,Code_Mat, Date_Ex,Note) VALUES
(3,'M2.2',getdate()-1,16)
⇒ **Effectuée**

- INSERT INTO Examen (Code_Etud,Code_Mat,Date_Ex,Note) VALUES (12,'M2.1',getdate()+1,15)

⇒ **Echouée, Violation de la contrainte « Date_Ex »**

9- Affichage de tous les étudiants

USE Evaluation ;

SELECT * FROM Etudiant ;

10- Affichage du nombre d'étudiants

USE Evaluation ;

SELECT count(*) FROM Etudiant ;

11- Test des mises à jour en Cascade

USE Evaluation .

UPDATE Examen SET Code= Code+100;

DELETE FROM Matière Where Code='M2.1'

⇒ **Nous remarquons que les modifications opèrent également dans la table « Examen ».**

12- Affichage des examens des étudiants de code 1, 2,3 ordonnés par code étudiant croissant et par ordre décroissant sur la note pour chaque étudiant

USE Evaluation;

SELECT dbo.Examen.Code, dbo.Etudiant.Code as Code_Etud , Code_Mat, Note

FROM dbo.Examen,dbo.Etudiant

WHERE dbo.Etudiant.Code = Code_Etud AND dbo.Etudiant.Code IN(101,102,103)

ORDER BY Code_Etud, Note desc

13- Affichage pour chaque étudiant du nombre d'examens, la note minimale et la note maximale obtenues

USE Evaluation;

SELECT Etudiant.Code,count(Examen.Code) as 'Nbr examens', min(Note) as 'Note

Min', max(Note) as 'Note Max'

FROM Etudiant, Examen

Where Examen.Code_Etud = Etudiant.Code
GROUP BY Etudiant.Code

14- Modification de la note de l'étudiant 2 obtenue dans l'examen de la matière de code M1.2, la nouvelle note étant 14

USE Evaluation;
UPDATE Examen
SET Note = 14
WHERE Code_Etud = 102 and Code_Mat = 'M2.1'

15- Suppression des examens dont la note est entre 0 et 8

USE Evaluation;
DELETE FROM Examen
WHERE Note between 0 and 8

16- Affichage de la moyenne générale de chaque étudiant

USE Evaluation;
SELECT Code_Etud, (sum(Note*Coef)/Sum(Coef))
FROM Examen,Matière
WHERE Examen.Code_Mat = Matière.Code
GROUP BY Code_Etud

17- Affichage de la note minimale et de la note maximale

USE Evaluation;
SELECT min(Note), max(Note) FROM Examen

18- Affichage des étudiants ayant obtenu la note minimale ou la note maximale

USE Evaluation;

```
SELECT Code_Etud,Note  
FROM Examen  
WHERE note=(Select min(note) FROM Examen) OR note=(Select max(note)  
FROM Examen)
```

19- Affichage des matières concernées par la note maximale et la note minimale

```
USE Evaluation;  
SELECT Code_Etud,Note, Code_Mat  
FROM Examen  
WHERE note=(Select min(note) FROM Examen) OR note=(Select max(note)  
FROM Examen)
```

20- Suppression des examens dans lesquels la note obtenue est soit la note minimale ou la note maximale

```
USE Evaluation;  
DELETE FROM Examen  
WHERE note=(Select min(note) FROM Examen) OR note=(Select max(note)  
FROM Examen)
```