

## S2.04 Exploitation d'une Base de Données

A R N A U D   A U B L E T  
M A K S E N   G H R O U S

---

R E N D U   D E   L A  
S A E   2 0 4

---

2 0 2 2 - 2 0 2 3

## **S2.04 Exploitation d'une Base de Données**

### **II/ Modélisation des données :**

Cahier des charges - Projet de gestion des notes des étudiants en BUT

#### **Objectifs du projet**

Il faut concevoir un modèle de données adapté à la gestion des notes des étudiants. Le modèle devra prendre en compte les informations relatives aux étudiants, aux cours et aux notes attribuées.

Il faut développer des fonctionnalités permettant de gérer les données dérivées telles que les relevés de notes, les bilans, etc. Ces fonctionnalités devront être capables de générer et de mettre à jour ces données de manière précise et cohérente.

Restrictions d'accès : Il faut mettre en place des restrictions d'accès pour garantir la confidentialité et la sécurité des données. Ces restrictions devront permettre un accès différent en fonction des rôles des utilisateurs (étudiants, enseignants, responsables de matière, etc.).

#### **Exigences fonctionnelles**

- a. Enregistrement des données des étudiants : Permettre l'ajout, la modification et la suppression des informations relatives aux étudiants.
- b. Gestion des données dérivées : Fournir des fonctionnalités pour générer automatiquement les relevés de notes, les bilans et d'autres données dérivées pertinentes.
- c. Restrictions d'accès : Mettre en place des mécanismes de contrôle d'accès pour limiter l'accès aux données sensibles en fonction des rôles des utilisateurs.
- d. Sécurité des données : Assurer la confidentialité et l'intégrité des données en mettant en place des mesures de sécurité appropriées.

#### **Exigences techniques**

Le système devra utiliser une base de données relationnelle pour stocker et gérer les notes des étudiants. En utilisant SQL.

## S2.04 Exploitation d'une Base de Données

### Livrables

Les livrables attendus sont les suivants :

- le modèle de données finalisé,
- le système de gestion des notes fonctionnel,
- le système de restriction d'accès fonctionnel,

### Échéance

2 semaines

#### 1. Modeles

Base de données : Notre base de données possède 4 tables différentes :

- La table Enseignant, qui contient :
  - IdEnseignant, un entier qui représente l'id du Enseignant, qui est aussi une clé primaire.
  - Nom, une chaîne de caractères contenant le nom d'un Enseignant.
  - Prenom, une chaîne de caractères contenant le prénom d'un Enseignant.
  - Numero, qui contient un entier représentant le numéro de téléphone d'un Enseignant.
  - Mail, une chaîne de caractères qui contient l'adresse mail d'un Enseignant.
- La table Etudiants, qui contient :
  - IdEtudiant, un entier qui représente l'id de l'étudiant, qui est aussi une clé primaire.
  - Nom, une chaîne de caractères qui contient le nom de l'étudiant.
  - Prenom, une chaîne de caractères qui contient le prénom de l'étudiant.
  - Groupe, une chaîne de caractère qui contient le groupe de l'étudiant.
  - Numero, un entier qui contient le numéro de téléphone de l'étudiant.
  - Mail, une chaîne de caractères qui contient l'adresse mail de l'étudiant.
- La table Matières, qui contient :
  - IdEnseignant, qui référence la clé IdEnseignant de la table Enseignant, qui fait partie de la clé primaire.
  - IdMatière, une chaîne de caractères qui contient le code de la matière, et est la seconde partie de la clé primaire (ex : R101)
  - NomMatière, une chaîne de caractère qui contient le nom de la matière (ex : SQL avancée).
  - ChefMatiere, un boolean qui indique si le Enseignant a qui appartient l'ID est le chef de la matière.

## S2.04 Exploitation d'une Base de Données

- La table Controles, qui contient :
  - IdEnseignant, qui référence la clé IdEnseignant de la table Enseignants, et qui est la première partie de la clé primaire.
  - IdEtudiant, qui référence la clé IdEtudiant de la table Etudiants, et qui est la deuxième partie de la clé primaire.
  - IdMatiere, qui référence la clé IdMatiere de la table Matieres, et qui est la troisième partie de la clé primaire.
  - IdControle, une chaîne de caractère qui contient le code du contrôle, et qui est la quatrième et dernière partie de la clé primaire.
  - Nom, une chaîne de caractère qui contient le nom du contrôle.
  - Note, un float qui représente la note de l'élève.

3. Définir les règles de gestion de ces données et leurs mises en œuvre par des procédures stockées.

- Le groupe admin, qui possède tous les droits

Le groupe étudiant, qui possède les droits suivants :

- Table Enseignant : Aucun droit, ne voit pas la table
- Table Etudiant : Voir soi-même mais ne peut rien modifier
- Table Matiere : Voir ses matières mais ne peut rien modifier
- Table Contrôle : Voir mais ne peut rien modifier

- Le groupe enseignant, qui possède les droits suivants :

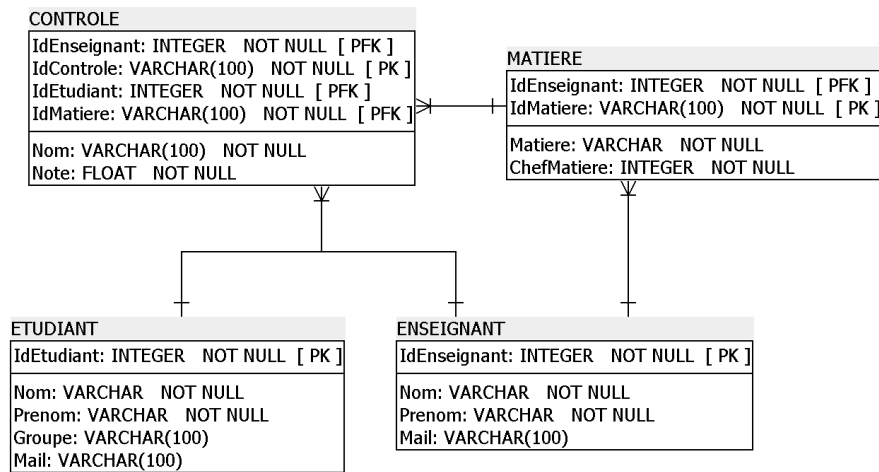
- Table Enseignant : Voir mais ne peut pas modifier
- Table Etudiant : Voir mais ne peut rien modifier
- Table Matiere : Voir mais ne peut rien modifier
- Table Contrôle : À tous les droits

4. Fournir un script de création de la base de données. Vous devez rendre un dossier comportant au moins 2 éléments essentiels :

## S2.04 Exploitation d'une Base de Données

```
CREATE TABLE Enseignant(  
  IdEnseignant integer primary key,  
  Nom varchar not null,  
  Prenom varchar not null,  
  Mail varchar not null);  
  
CREATE TABLE Etudiant(  
  IdEtudiant integer primary key,  
  Nom varchar not null,  
  Prenom varchar not null,  
  Groupe varchar not null,  
  Mail varchar not null);  
  
CREATE TABLE Matiere(  
  IdEnseignant integer,  
  IdMatiere varchar not null,  
  Matiere varchar not null,  
  ChefMatiere integer,  
  primary key (IdEnseignant, IdMatiere),  
  foreign key (IdEnseignant) references Enseignant(IdEnseignant));  
foreign key (ChefMatiere) references Enseignant(IdEnseignant))  
  
CREATE TABLE Controle(  
  IdEnseignant integer,  
  IdEtudiant integer,  
  IdControle varchar not null,  
  IdMatiere varchar not null,  
  Nom varchar not null,  
  Note float,  
  primary key (IdEnseignant, IdEtudiant, IdControle, IdMatiere),  
  foreign key (IdEnseignant) references Enseignant(IdEnseignant),  
  foreign key (IdEtudiant) references Etudiant(IdEtudiant));
```

## S2.04 Exploitation d'une Base de Données



### III/ Visualisation des données :

1/ Ensemble de données dérivées à visualiser :

1. Moyenne d'un étudiant
2. Moyenne d'un groupe
3. Moyenne de la promotion
4. Moyenne d'une matière
5. Relevé des notes d'une Sae
6. Relevé des notes d'une Sae d'un groupe
7. Relevé des notes de Sae d'un étudiant
8. Moyenne d'une matière d'un étudiant
9. Moyenne d'une matière d'un groupe

2/ Visualisation des données :

#### Moyenne d'un étudiant

```
CREATE FUNCTION Moyenne_Etudiant (in id int, out IDetudiant int, out
                                Moyenne float, out idMatiere varchar)
    RETURNS SETOF RECORD
AS
$$

Select idetudiant, avg(note), idmatiere from controle where
controle.idetudiant = $1 group by idetudiant, idmatiere;
```

## S2.04 Exploitation d'une Base de Données

```
$$ language sql;

select * from Moyenne_Etudiant('12345678');
```

Cela donne la moyenne de chaque matière pour un étudiant grâce au numéro étudiant.

### Moyenne d'un groupe

```
CREATE FUNCTION Moyenne_Groupe (in groupe varchar, out Groupe varchar,
                                out Moyenne float, out idMatiere varchar)
    RETURNS SETOF RECORD
AS
$$

Select groupe, avg(note), idmatiere from controle JOIN etudiant on
Controle.idEtudiant = Etudiant.idetudiant where etudiant.groupe = $1 group by
idetudiant, groupe;

$$ language sql;

Select * from Moyenne_Groupe('Tlaloc');
```

Cela donne la moyenne de chaque matière pour un groupe grâce à son nom.

### Moyenne de la promotion

```
CREATE FUNCTION Moyenne_Promotion (out Moyenne float, out IdMatiere varchar)
    RETURNS SETOF RECORD
AS
$$

Select avg(note), iIMatiere from controle group by IdMatiere;

$$ language sql;

Select * from Moyenne_General();
```

Cela donne la moyenne de chaque matière pour la promotion.

### Moyenne d'une matière

## S2.04 Exploitation d'une Base de Données

```
CREATE FUNCTION Moyenne_Matiere (in id varchar, out Moyenne float,  
                                out idMatiere varchar)  
RETURNS SETOF RECORD  
AS  
$$  
  
Select avg(note), IdMatiere from controle where controle.IdMatiere=$1 group  
by IdMatiere;  
  
$$ language sql;  
  
Select * from Moyenne_Matiere('R1-04');
```

Cela donne la moyenne de la matière grâce à son identifiant.

### Relevé des notes des Sae

```
CREATE FUNCTION Releve_Sae(out Nom varchar, out Prenom varchar,  
                           out Groupe varchar, out IdControle varchar,  
                           out IdMatiere varchar, out Nom_Controlle varchar,  
                           out Note float)  
RETURNS SETOF RECORD  
AS  
$$  
  
Select Etudiant.Nom, Prenom, IdControle, IdMatiere, Note From Controle JOIN  
Etudiant on Controle.IdEtudiant = Etudiant.IdEtudiant WHERE Controle.Nom='SAE'  
Order by Etudiant.Nom, Prenom, IdControle  
  
$$language SQL;  
  
select * from Releve_Sae();
```

Cela donne le relevé de toutes les notes des Sae.

### Relevé des notes des Sae d'un groupe

```
CREATE FUNCTION Releve_Groupe(in Groupe varchar, out Nom varchar,  
                              out Prenom varchar, out Groupe varchar,  
                              out IdControle varchar, out IdMatiere varchar,  
                              out Nom_Controlle varchar, out Note float)  
RETURNS SETOF RECORD
```



## S2.04 Exploitation d'une Base de Données

```
AS
$$

Select Etudiant.Nom, Prenom, IdControle,IdMatiere,Note From Controle JOIN
Etudiant on Controle.IdEtudiant = Etudiant.IdEtudiant WHERE Etudiant.Groupe=$1
AND Controle.Nom='SAE' Order by Etudiant.Nom,Prenom,IdControle

$$language SQL;

select * from Releve_Groupe('Tlalloc');
```

Cela nous donne le relevé de toutes les notes des Sae d'un groupe entier grâce à son nom.

### IV/ Restrictions d'accès aux Données

- un étudiant ne peut consulter que ses propres notes,
- un étudiant ne peut consulter que ses propres matières
- un enseignant doit pouvoir saisir les notes de ses contrôles,
- un enseignant doit pouvoir ajouter des contrôles,
- un enseignant doit pouvoir consulter les notes des étudiants de ses contrôles,
- un enseignant ne peut pas modifier les enseignant,
- un enseignant ne peut pas modifier les matières,
- un enseignant ne peut pas modifier les étudiants.
- l'admin, possède tous les droits

**Enlever puis mettre les bonnes permissions :**

```
REVOKE INSERT,UPDATE,DELETE TO etudiant;

CREATE user '01234567' WITH PASSWORD '1234';
CREATE user '23456789' WITH PASSWORD '0000';
```

## S2.04 Exploitation d'une Base de Données

```
CREATE user '12345678' WITH PASSWORD '1111';
CREATE user ADMIN WITH PASSWORD '9999';

CREATE GROUP role_enseignant WITH USER '01234567';
CREATE GROUP role_etudiant WITH USER '23456789', '12345678';

GRANT select ON enseignant to role_enseignant;
GRANT select ON etudiant to role_enseignant;
GRANT select, update, insert ON matiere TO role_enseignant;
GRANT select, update, insert ON controle TO role_enseignant;
GRANT select ON matiere to role_etudiant;
GRANT select ON controle to role_etudiant;

GRANT all privileges on enseignant TO ADMIN;
GRANT all privileges on etudiant TO ADMIN;
GRANT all privileges on matiere TO ADMIN;
GRANT all privileges on controle TO ADMIN;
```

Fonction pour qu'un enseignant change une note :

```
CREATE FUNCTION Changer_Note(in Note float, in idEt int, in IdMat varchar,
                             in NomControle varchar)
    RETURNS float
AS
$$

UPDATE controle SET Note=$1 WHERE $2 = Controle.IdEtudiant AND
Controle.IdEnseignant = session_user AND Controle.IdControle = $3 AND
Controle.Nom = $4 RETURNING Note

$$language SQL;
```

Cela permet à un enseignant de pouvoir modifier n'importe quelle note qu'il a déjà rentré et que les siennes.

Fonction pour qu'un élève accède à son relevé de note :

## S2.04 Exploitation d'une Base de Données

```
CREATE FUNCTION MesResultats(out Matiere varchar, out Controle varchar,  
                             out Nom_Controlle varchar, out Note float)  
    RETURNS SETOF RECORD  
AS  
$$  
  
Select Matiere, IdControle, Nom, Note FROM Matiere JOIN Controle WHERE  
Controle.IdEtudiant = session_user  
  
$$language SQL;  
  
Select * from MesResultats();
```

Cela permet à un étudiant d'accéder à ses notes et seulement aux siennes.