

SGBD : BASES DE DONNÉES AVANCÉES [M3106C]

TD N°5 - NORMALISATION DE DONNÉES

OBJECTIFS

- Règles de gestions et Dépendances Fonctionnelles
- Erreurs et Normalisation

ENONCÉS

Exercice I :

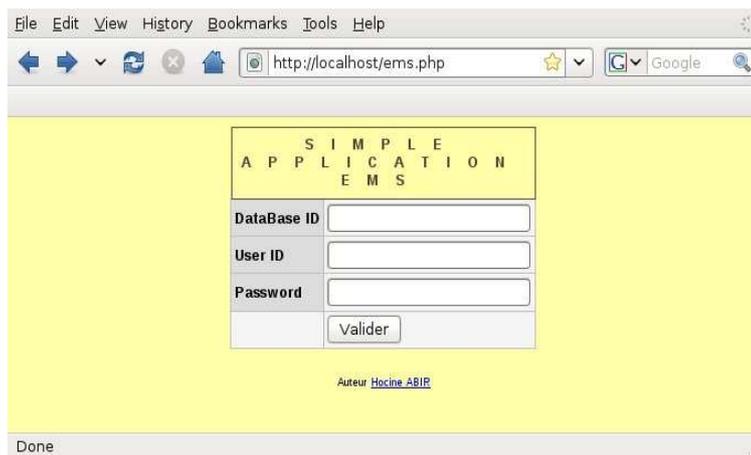
On considère la relation EMS non normalisée suivante :

```
CREATE TABLE EMS
(
  nom varchar(60),
  prenom varchar(60),
  matiere varchar(60),
  salle varchar(60)
);
```

avec les dépendances fonctionnelles suivantes :

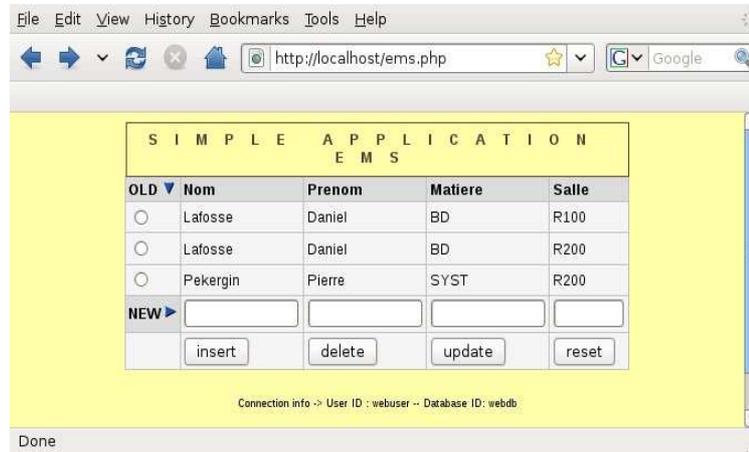
$F = \{ \text{nom} \rightarrow \text{prenom}, \text{nom matiere} \rightarrow \text{salle} \}$ et une simple application `ems.php` développée autour de ce schéma de relation. Cette application utilise deux fenêtres :

- (1) Connexion à la base de données où est définie la relation EMS.



Date: 30 septembre 2014.
Hocine ABIR - IUT Villetaneuse .

- (2) Opérations : insertion, mise à jour et suppression de données de EMS.



Question 1.1. Montrer les différentes erreurs que l'on peut avoir sur la relation EMS du fait que cette relation n'est pas normalisée.

L'application `ems.php` utilise l'identifiant de tuple `ctid` pour distinguer les différentes lignes de la relation.

Question 1.2. Montrer dans quel cas certaines opérations peuvent être perdues.

Pour résoudre les problèmes de la **Question 1.1** sans modifier `ems.php`, on propose de décomposer EMS en deux relations comme suit :

```
CREATE TABLE EMS1
(
    nom varchar(60) primary key,
    prenom varchar(60)
);

CREATE TABLE EMS2
(
    nom varchar(60),
    matiere varchar(60),
    salle varchar(60),
    primary key(nom,matiere)
);

ALTER TABLE EMS2
ADD FOREIGN KEY(nom) REFERENCES EMS1(nom)
DEFERRABLE INITIALLY DEFERRED;
```

Question 1.3. *Instancier EMS1 et EMS2 à partir de EMS.*

Question 1.4. *Remplacer la relation EMS par un schéma externe (vue) EMS décrit comme suit :*

| Column | Type |
|---------|-----------------------|
| nom | character varying(60) |
| prenom | character varying(60) |
| matiere | character varying(60) |
| salle | character varying(60) |
| ctid | tid |

et tester l'application `ems.php`.

Question 1.5. *insert*

Compléter le schéma externe pour permettre l'insertion de nouveaux tuples : seule la contrainte de clé primaire de EMS2 peut entraîner un echec.

Question 1.6. *delete*

Compléter le schéma externe pour permettre la suppression de tuples : un tuple de EMS1 non référencié par EMS2 doit être supprimé de EMS1.

Question 1.7. *update*

Compléter le schéma externe pour permettre la mise à jour de tuples en supprimant l'ancienne image et en insérant la nouvelle image si nécessaire.

Question 1.8. *Définir les droits d'accès nécessaires pour que n'importe quel utilisateur puisse utiliser l'application `ems.php`.*

Question 1.9. *Modifier le schéma externe et/ou les droits d'accès de sorte qu'un utilisateur ne puisse modifier (`update`) ou supprimer (`delete`) que les lignes qu'il a créées (`insert`).*