

TP – Bases de Données : Gestion d'un Centre de Formation

Objectif

Créer une base de données relationnelle pour gérer un centre de formation : étudiants, formateurs, formations, modules, salles et inscriptions.

1. Modèle de données

Table 1 : Etudiant

Champ	Type	Contraintes
id_etudiant	INT	PK, AUTO_INCREMENT
nom	VARCHAR(50)	NOT NULL
prenom	VARCHAR(50)	NOT NULL
email	VARCHAR(100)	UNIQUE

Table 2 : Formateur

Champ	Type	Contraintes
id_formateur	INT	PK, AUTO_INCREMENT
nom	VARCHAR(50)	NOT NULL
prenom	VARCHAR(50)	NOT NULL
specialite	VARCHAR(100)	

Table 3 : Formation

Champ	Type	Contraintes
id_formation	INT	PK, AUTO_INCREMENT
intitule	VARCHAR(100)	NOT NULL
duree_heures	INT	NOT NULL
id_formateur	INT	FK vers Formateur(id_formateur)

Table 4 : Module

Champ	Type	Contraintes

<code>id_module</code>	INT	PK, AUTO_INCREMENT
<code>nom_module</code>	VARCHAR(100)	NOT NULL
<code>id_formation</code>	INT	FK vers Formation(id_formation)

Table 5 : Salle

Champ	Type	Contraintes
<code>id_salle</code>	INT	PK, AUTO_INCREMENT
<code>nom_salle</code>	VARCHAR(50)	NOT NULL
<code>capacite</code>	INT	NOT NULL

Table 6 : Inscription

Champ	Type	Contraintes
<code>id_inscription</code>	INT	PK, AUTO_INCREMENT
<code>id_etudiant</code>	INT	FK vers Etudiant(id_etudiant)
<code>id_formation</code>	INT	FK vers Formation(id_formation)
<code>date_inscription</code>	DATE	NOT NULL

2. Mettre les relations entre les tables

Ajouter 3 enregistrements dans chaque table

Étudiant

- Dupont, Marc, marc.dupont@mail.com
- Martin, Lea, lea.martin@mail.com
- Durand, Karim, karim.durand@mail.com

Formateur

- Bernard, Sophie, Bases de données
- Lopez, Julien, Réseaux
- Renaud, Clara, Développement Web

Formation

- BTS SIO SLAM, 1200 h, formateur 3
- BTS SIO SISR, 1200 h, formateur 2
- SQL Niveau 1, 40 h, formateur 1

Module

- SQL – Initiation, formation 3
- HTML/CSS, formation 1
- Administration réseaux, formation 2

Salle

- Salle A01, 30 places
- Salle B12, 20 places
- Salle Info, 16 places

Inscription

- Étudiant 1 → Formation 3
- Étudiant 2 → Formation 1
- Étudiant 3 → Formation 1

3. Requêtes SQL à réaliser

Requêtes simples sans jointure

①Afficher tous les étudiants

```
SELECT * FROM Etudiant;
```

②Afficher les formations dont la durée est supérieure à 100 heures

```
SELECT * FROM Formation
```

```
WHERE duree_heures > 100;
```

③Afficher les formateurs spécialisés en “Réseaux”

```
SELECT * FROM Formateur
```

```
WHERE specialite = 'Réseaux';
```

④Afficher les modules de la formation n°1

(On utilise simplement le champ id_formation, pas de jointure)

```
SELECT * FROM Module
```

```
WHERE id_formation = 1;
```

5]Afficher les inscriptions d'un étudiant donné (id = 1)

```
SELECT * FROM Inscription
```

```
WHERE id_etudiant = 1;
```

6]Afficher les salles dont la capacité est ≥ 20

```
SELECT * FROM Salle
```

```
WHERE capacite >= 20;
```

7]Afficher les noms des étudiants triés par ordre alphabétique

```
SELECT nom, prenom
```

```
FROM Etudiant
```

```
ORDER BY nom ASC;
```

8]Afficher les formations dont l'intitulé contient "SIO"

```
SELECT * FROM Formation
```

```
WHERE intitule LIKE '%SIO%';
```
