

Rapport : Gestion d'un Centre de Formation avec Procédures et Fonctions Stockées

Module : Base de Données MySQL

Nom de l'étudiante : Aya ouazzani akkar hammoudi

SOMMAIRE :

- 1) Le contexte et les objectifs pédagogiques
- 2) Le code SQL complet
- 3) Des captures d'écran prouvant l'exécution des requêtes
- 4) Une analyse des résultats
- 5) conclusion personnelle

I. contexte :

Ce projet pratique a pour objectif de concevoir et manipuler une base de données MySQL complète permettant de gérer les formateurs, stagiaires et sessions d'un centre de formation. À travers cet atelier, nous avons appliqué les notions avancées de SQL : procédures stockées, fonctions stockées, boucles, structures conditionnelles et paramètres IN, OUT, INOUT.

Objectifs pédagogiques

- Créer et manipuler une base de données MySQL complète.
- Utiliser des procédures stockées avec différents types de paramètres.
- Mettre en place des fonctions retournant des valeurs calculées.
- Appliquer les structures de contrôle : IF, WHILE, CASE.

2. Conception de la base de données

La base de données utilisée est nommée 'techlearn_db' voici le code SQL de création :

- **Creation:**

CRÉATION DE LA BASE:

```
1 CREATE DATABASE techlearn_db;  
2 USE techlearn_db; |
```

CRÉATION DES TABLES:

```
2 id INT AUTO_INCREMENT PRIMARY KEY,  
3 nom VARCHAR(100),  
4 specialite VARCHAR(100)  
5 );  
6 CREATE TABLE stagiaires (  
7 id INT AUTO_INCREMENT PRIMARY KEY,  
8 nom VARCHAR(100),  
9 ville VARCHAR(100)  
10 );  
11 CREATE TABLE sessions (  
12 id INT AUTO_INCREMENT PRIMARY KEY,  
13 intitule VARCHAR(100),  
14 formateur_id INT,  
15 capacite INT,  
16 inscrits INT DEFAULT 0,
```

insertion :

```
1 INSERT INTO formateurs (nom, specialite) VALUES
2 ('Mme Nadia', 'Bases de données'),
3 ('M. Karim', 'Développement Web'),
4 ('Mme Salma', 'Analyse de données');
5 INSERT INTO stagiaires (nom, ville) VALUES
6 ('Ali', 'Casablanca'),
7 ('Sara', 'Rabat'),
8 ('Mouna', 'Marrakech');
9 INSERT INTO sessions (intitule, formateur_id, capacite, inscrits)
10 VALUES
11 ('MySQL avancé', 1, 20, 5),
12 ('HTML/CSS', 2, 25, 10),
13 ('Python Data', 3, 15, 8);
```

V. Procédures et Fonctions Stockées

PROCÉDURE AVEC PARAMÈTRE IN:

```
1 DELIMITER /
2 CREATE PROCEDURE afficher_stagiaires_par_ville(IN ville_nom
3 VARCHAR(100))
4 BEGIN
5     SELECT * FROM stagiaires WHERE ville = ville_nom;
6 END /
7 DELIMITER ;
8 CALL afficher_stagiaires_par_ville('Casablanca');
```

PROCÉDURE AVEC PARAMÈTRE OUT:

```
1 DELIMITER /
2 CREATE PROCEDURE compter_sessions(OUT total INT)
3 BEGIN
4     SELECT COUNT(*) INTO total FROM sessions;
5 END /
6 DELIMITER ;
7 CALL compter_sessions(@nb);
8 SELECT @nb AS nombre_total_sessions;
```

PROCÉDURE AVEC PARAMÈTRE INOUT:

```

1 DELIMITER /
2 CREATE PROCEDURE inscrire_stagiaire(INOUT nb_inscrits INT, IN
3 session_id INT)
4 BEGIN
5     UPDATE sessions SET inscrits = inscrits + 1 WHERE id =
6 session_id;
7     SELECT inscrits INTO nb_inscrits FROM sessions WHERE id =
8 session_id;
9 END /
10 DELIMITER ;
11 SET @nouveau = 0;
12 CALL inscrire_stagiaire(@nouveau, 1);
13 SELECT @nouveau AS nouveau_nombre_inscrits;

```

FONCTION : TAUX D'OCCUPATION D'UNE SESSION:

```

1 DELIMITER //
2 DROP PROCEDURE IF EXISTS inscrire_stagiaire //
3 DROP FUNCTION IF EXISTS taux_occupation //
4 CREATE PROCEDURE inscrire_stagiaire(INOUT nb_inscrits INT, IN session_id INT)
5 BEGIN
6     UPDATE sessions
7     SET inscrits = inscrits + 1
8     WHERE id = session_id;
9
10    SELECT inscrits INTO nb_inscrits
11    FROM sessions
12    WHERE id = session_id;
13 END //
14 CREATE FUNCTION taux_occupation(id_session INT)
15 RETURNS DECIMAL(5,2)

```

```

18    DECLARE taux DECIMAL(5,2);
19    SELECT (inscrits / capacite) * 100 INTO taux
20    FROM sessions
21    WHERE id = id_session;
22    RETURN taux;
23 END //
24 DELIMITER ;
25 SET @nouveau = 0;
26 CALL inscrire_stagiaire(@nouveau, 1);
27
28 SELECT @nouveau AS nouveau_nombre_inscrits;
29 SELECT intitule, taux_occupation(id) AS taux
30 FROM sessions;

```

BOUCLE ET AUTOMATISATION

```

1 DELIMITER /
2 CREATE PROCEDURE generer_sessions(IN formateur INT)
3 BEGIN
4 DECLARE i INT DEFAULT 1;
5 WHILE i <= 10 DO
6 INSERT INTO sessions (intitule, formateur_id, capacite,
7 inscrits)
8 VALUES (CONCAT('Session ', i, ' - Formateur ', formateur),
9 formateur, 20, 0);
10 SET i = i + 1;
11 END WHILE;
12 END /
13 DELIMITER ;
14 CALL generer_sessions(2);

```

DÉFIS FINAUX :

FONCTION SESSIONS_FORMATEUR (NOM_FORMATEUR):

```

1 DELIMITER /
2 CREATE FUNCTION sessions_formateur(nom_formateur VARCHAR(100))
3 RETURNS INT
4 BEGIN
5 DECLARE nb_sessions INT;
6 SELECT COUNT(s.id) INTO nb_sessions
7 FROM sessions s
8 JOIN formateurs f ON s.formateur_id = f.id
9 WHERE f.nom = nom_formateur;
10 RETURN nb_sessions;
11 END /
12 DELIMITER ;
13
14 SELECT sessions_formateur('H. Karim') AS nb_sessions_karim

```

DÉFIS FINAUX :

PROCÉDURE SUPPRIMER_SESSIONS_VIDES ():

```

1 DELIMITER /
2 CREATE PROCEDURE supprimer_sessions_vides()
3 BEGIN
4 DELETE FROM sessions WHERE inscrits = 0;
5 END /
6 DELIMITER ;
7
8 CALL supprimer_sessions_vides();

```

DÉFIS FINAUX: PROCÉDURE VALIDER_INSCRIPTION (IN SESSION_ID INT):

```

1 DELIMITER /
2 CREATE PROCEDURE valider_inscription(IN session_id INT)
3 BEGIN
4 DECLARE nb_inscrits INT;
5 DECLARE capacite_max INT;
6 SELECT inscrits, capacite INTO nb_inscrits, capacite_max
7 FROM sessions WHERE id = session_id;
8 IF nb_inscrits >= capacite_max THEN
9 SELECT 'Session complète' AS message;
10 ELSE
11 UPDATE sessions SET inscrits = inscrits + 1 WHERE id = session_id;
12 SELECT 'Inscription validée' AS message;
13 END IF;
14 END /
15 DELIMITER ;
16 CALL valider_inscription(1);

```

3. Analyse des résultats :

CRÉATION DE LA BASE :

```
✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0,0034 seconde(s))

CREATE DATABASE techlearn_db;

[ Éditer en ligne ] [ Éditer ] [ Créer le code source PHP ]

Error: #1046 Aucune base n'a été sélectionnée

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0,0003 seconde(s))

USE techlearn_db;

[ Éditer en ligne ] [ Éditer ] [ Créer le code source PHP ]

Error: #1046 Aucune base n'a été sélectionnée
```

CRÉATION DE TABLES :

```
✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0,0223 seconde(s))

CREATE TABLE formateurs ( id INT AUTO_INCREMENT PRIMARY KEY, nom VARCHAR(100), specialite VARCHAR(100) );

[ Éditer en ligne ] [ Éditer ] [ Créer le code source PHP ]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0,0120 seconde(s))

CREATE TABLE stagiaires ( id INT AUTO_INCREMENT PRIMARY KEY, nom VARCHAR(100), ville VARCHAR(100) );

[ Éditer en ligne ] [ Éditer ] [ Créer le code source PHP ]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0,0138 seconde(s))

CREATE TABLE sessions ( id INT AUTO_INCREMENT PRIMARY KEY, intitule VARCHAR(100), formateur_id INT, capacite INT, inscrits INT DEFAULT 0, FOREIGN KEY (formateur_id) REFERENCES formateurs(id) );

[ Éditer en ligne ] [ Éditer ] [ Créer le code source PHP ]
```

INSERTION DE DONNÉES :

```
Afficher la zone SQL

✓ 3 lignes insérées.
Identifiant de la ligne insérée : 3 (traitement en 0,0045 seconde(s))

INSERT INTO formateurs (nom, specialite) VALUES ('Hme Nadia', 'Bases de données'), ('M. Karim', 'Développement Web'), ('Hme Salma', 'Analyse de données');

[ Éditer en ligne ] [ Éditer ] [ Créer le code source PHP ]

✓ 3 lignes insérées.
Identifiant de la ligne insérée : 3 (traitement en 0,0005 seconde(s))

INSERT INTO stagiaires (nom, ville) VALUES ('Ali', 'Casablanca'), ('Sara', 'Rabat'), ('Houna', 'Marrakech');

[ Éditer en ligne ] [ Éditer ] [ Créer le code source PHP ]

✓ 3 lignes insérées.
Identifiant de la ligne insérée : 3 (traitement en 0,0005 seconde(s))

INSERT INTO sessions (intitule, formateur_id, capacite, inscrits) VALUES ('MySQL avancé', 1, 20, 5), ('HTML/CSS', 2, 25, 10), ('Python Data', 3, 15, 8);

[ Éditer en ligne ] [ Éditer ] [ Créer le code source PHP ]
```

PROCÉDURE AVEC PARAMÈTRE IN :

Show query box

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0086 seconds.)

CREATE PROCEDURE generer_sessions(IN formateur INT) BEGIN DECLARE i INT DEFAULT 1; WHILE i <= 10 DO INSERT INTO sessions (intitule, formateur_id, capacite, inscrites) VALUES (CONCAT('Session ', i, ' - Formateur ', formateur), formateur, 20, 0); SET i = i + 1; END WHILE; END;
```

[Edit inline] [Edit] [Create PHP code]

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0273 seconds.)

CALL generer_sessions(2);
```

[Edit inline] [Edit] [Create PHP code]

PROCÉDURE AVEC PARAMÈTRE OUT :

Structure SQL Search Query Export Import Operations Privileges Routines More

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0079 seconds.)

CREATE PROCEDURE compter_sessions(OUT total INT) BEGIN SELECT COUNT(*) INTO total FROM sessions; END;
```

[Edit inline] [Edit] [Create PHP code]

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0002 seconds.)

CALL compter_sessions(@nb);
```

[Edit inline] [Edit] [Create PHP code]

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available. ⓘ

```
Showing rows 0 - 0 (1 total, Query took 0.0001 seconds.)

SELECT @nb AS nombre_total_sessions;
```

☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 | Filter rows: Search this table

Extra options

PROCÉDURE AVEC PARAMÈTRE IN OUT :

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0078 seconds.)

CREATE PROCEDURE inscrire_stagiaire(INOUT nb_inscrits INT, IN session_id INT) BEGIN UPDATE sessions SET inscrites = inscrites + WHERE id = session_id; SELECT inscrites INTO nb_inscrits FROM sessions WHERE id = session_id; END;
```

[Edit inline] [Edit] [Create PHP code]

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0001 seconds.)

SET @nouveau = 0;
```

[Edit inline] [Edit] [Create PHP code]

```
MySQL returned an empty result set (i.e. zero rows). (Query took 0.0025 seconds.)

CALL inscrire_stagiaire(@nouveau, 1);
```

[Edit inline] [Edit] [Create PHP code]

⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available. ⓘ

```
Showing rows 0 - 0 (1 total, Query took 0.0001 seconds.)

SELECT @nouveau AS nouveau_nombre_inscrits;
```

Console

FONCTION : TAUX D'OCCUPATION D'UNE SESSION :

```
✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0255 seconde(s))
DROP PROCEDURE IF EXISTS inscrire_stagiaire;
[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.1119 seconde(s))
DROP FUNCTION IF EXISTS taux_occupation;
[Éditer en ligne] [Éditer] [Créer le code source PHP]

Note: #1305 FUNCTION techlaam_db.taux_occupation does not exist

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0072 seconde(s))
CREATE PROCEDURE inscrire_stagiaire(INOUT nb_inscrits INT, IN session_id INT) BEGIN UPDATE sessions SET inscrits = inscrits + 1 WHERE id = session_id; SELECT inscrits INTO nb_inscrits FROM sessions WHERE id = session_id; END;
[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0066 seconde(s))
CREATE FUNCTION taux_occupation(id_session INT) RETURNS DECIMAL(9,2) DETERMINISTIC BEGIN DECLARE taux DECIMAL(9,2); SELECT (inscrits / capacite) * 100 INTO taux FROM sessions WHERE id = id_session; RETURN taux; END;
[Éditer en ligne] [Éditer] [Créer le code source PHP]
```

```
✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0001 seconde(s))
SET @nouveau = 0;
[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0014 seconde(s))
CALL inscrire_stagiaire(@nouveau, 1);
[Éditer en ligne] [Éditer] [Créer le code source PHP]

La sélection courante ne contient pas de colonne unique. Les grilles d'édition, les cases à cocher ainsi que les liens Éditer, Copier et Supprimer ne sont pas disponibles.

✓ Affichage des lignes 0 - 0 (total de 1, traitement en 0.0001 seconde(s))
SELECT @nouveau AS nouveau_nombre_inscrits;
[Prévisualiser] [Éditer en ligne] [Éditer] [Expliquer SQL] [Créer le code source PHP] [Actualiser]

☐ Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: | Chercher dans cette table
```

```
Parcourir Structures SQL Rechercher Insérer Exporter Importer Privilèges Opérations Déclencheurs
Options supplémentaires
nouveau_nombre_inscrits
0
☐ Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: | Chercher dans cette table

Opérations sur les résultats de la requête
Imprimer Copier dans le presse-papiers Exporter Afficher le graphique Créer une vue

✓ Affichage des lignes 0 - 2 (total de 3, traitement en 0.0026 seconde(s))
SELECT intitule, taux_occupation AS taux FROM sessions;
[Prévisualiser] [Éditer en ligne] [Éditer] [Expliquer SQL] [Créer le code source PHP] [Actualiser]

☐ Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: | Chercher dans cette table | Trier par col: Aucun(x)

Options supplémentaires
+ T +
intitule taux
☐ Éditer ☐ Copier ☐ Supprimer MySQL avancé 40.00
☐ Éditer ☐ Copier ☐ Supprimer HTML/CSS 40.00
☐ Éditer ☐ Copier ☐ Supprimer Python Data 53.33
```

BOUCLE ET AUTOMATISATION:

```
Structure SQL Rechercher Requête Exporter Importer Opérations Privilèges Procédures stockées Plus
Afficher la zone SQL

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0082 seconde(s))
CREATE PROCEDURE generer_sessions(IN formateur INT) BEGIN DECLARE i INT DEFAULT 1; WHILE i <= 10 DO INSERT INTO sessions (intitule, formateur_id, capacite, inscrits) VALUES (CONCAT('Session ', i, ' - Formateur ', formateur), formateur, 20, 0); SET i = i + 1; END WHILE; END;
[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0015 seconde(s))
CALL generer_sessions(2);
[Éditer en ligne] [Éditer] [Créer le code source PHP]
```


DÉFIS FINAUX: FONCTION SESSIONS_FORMATEUR(NOM_FORMATEUR):

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0083 seconde(s))

```
CREATE FUNCTION sessions_formateur(nom_formateur VARCHAR(100)) RETURNS INT BEGIN DECLARE nb_sessions INT; SELECT COUNT(s.id) INTO nb_sessions FROM sessions s JOIN formateurs f ON s.formateur_id = f.id WHERE f.nom = nom_formateur; RETURN nb_sessions; END;
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

⚠ La sélection courante ne contient pas de colonne unique. Les grilles d'édition, les cases à cocher ainsi que les liens Éditer, Copier et Supprimer ne sont pas disponibles.

✓ Affichage des lignes 0 - 0 (total de 1, traitement en 0.0032 seconde(s))

```
SELECT sessions_formateur('H. Karim') AS nb_sessions_karim;
```

☐ Profilage [Éditer en ligne] [Éditer] [Expliquer SQL] [Créer le code source PHP] [Actualiser]

☐ Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

Options supplémentaires

nb_sessions_karim

DÉFIS FINAUX :

PROCÉDURE SUPPRIMER_SESSIONS_VIDES()

Afficher la zone SQL

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0125 seconde(s))

```
CREATE PROCEDURE supprimer_sessions_vides() BEGIN DELETE FROM sessions WHERE inscrits = 0; END;
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0.0016 seconde(s))

```
CALL supprimer_sessions_vides();
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

DÉFIS FINAUX :

PROCÉDURE VALIDER_INSCRIPTION(IN SESSION_ID INT):

✓ MySQL a retourné un résultat vide (c'est à dire aucune ligne). (traitement en 0,0181 seconde(s).)

```
CREATE PROCEDURE valider_inscription(IN session_id INT) BEGIN DECLARE nb_inscrits INT; DECLARE capacite_max INT; SELECT inscrits, capacite INTO nb_inscrits, capacite_max FROM sessions WHERE id = session_id; IF nb_inscrits >= capacite_max THEN SELECT 'Session complète' AS message; ELSE UPDATE sessions SET inscrits = inscrits + 1 WHERE id = session_id; SELECT 'Inscription validée' AS message; END IF; END;
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

⚠ La sélection courante ne contient pas de colonne unique. Les grilles d'édition, les cases à cocher ainsi que les liens Éditer, Copier et Supprimer ne sont pas disponibles.

✓ Affichage des lignes 0 - 0 (total de 1, traitement en 0,0010 seconde(s).)

```
CALL valider_inscription(1);
```

[Éditer en ligne] [Éditer] [Créer le code source PHP]

☐ Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

Options supplémentaires

message

Inscription validée

☐ Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table

Opérations sur les résultats de la requête

4. Analyse des résultats:

Nom	Type	Retourne
<input type="checkbox"/> afficher_stagiaires_par_ville	PROCEDURE	Éditer Exécuter Exporter Supprimer
<input type="checkbox"/> compter_sessions	PROCEDURE	Éditer Exécuter Exporter Supprimer
<input type="checkbox"/> generer_sessions	PROCEDURE	Éditer Exécuter Exporter Supprimer
<input type="checkbox"/> inscrire_stagiaire	PROCEDURE	Éditer Exécuter Exporter Supprimer
<input type="checkbox"/> sessions_formatteur	FUNCTION int(11)	Éditer Exécuter Exporter Supprimer
<input type="checkbox"/> supprimer_sessions_vides	PROCEDURE	Éditer Exécuter Exporter Supprimer
<input type="checkbox"/> valider_inscription	PROCEDURE	Éditer Exécuter Exporter Supprimer

5. Conclusion personnelle

Cet atelier m'a permis de consolider ma maîtrise des concepts avancés de MySQL. J'ai appris à Créer et gérer une base de données complète pour un centre de formation et Utiliser des procédures et fonctions stockées pour automatiser des tâches et Manipuler les paramètres IN, OUT, INOUT et Mettre en pratique les boucles et conditions SQL.