

Introduction aux Bases de Données

Exercices et Travaux Pratiques
Nicolas Travers

TABLE DES MATIÈRES

1	BDD	3
1.1	Analyse des besoins	3
1.2	Modèle Entité/Association	3
1.3	Schéma relationnel	3
1.4	Contraintes	3
1.5	LDD : Langage de Description de Données	4
2	MySQL Workbench	5
2.1	Modélisation	5
2.2	Création de la base de données	5
3	SQL	6
3.1	Chargement des données	6
3.2	Algèbre et SQL	6
3.3	SQL	6
4	PL/SQL	8
4.1	Procédure simple	8
4.2	Fonction	8
4.3	Curseur et Exception	8
4.4	Procédure complexe	9
5	Trigger	10
5.1	Trigger	10
6	Création d'une application BD	11
6.1	Connexion à la base de données	11
6.2	Rôle étudiant	11
6.3	Rôle enseignant	11
6.4	Bonus : Rôle administratif	12

1 CRÉATION D'UNE BASE DE DONNÉES

1.1 Analyse des besoins

Soit le texte suivant décrivant les besoins pour une base de données gérant les enseignements d'un établissement d'enseignement supérieur.

Nous avons besoin d'une base de données permettant de gérer les enseignements de notre établissement. Il faut prendre en compte les étudiants avec leur nom, prénom, date de naissance et numéro d'identification, qui lors de leurs études dans le supérieur s'inscrivent à différents parcours (L1, L2, L3, M1, M2). Ces parcours ont des dénominations, sont reliés entre eux (représentant les possibilités de passage entre parcours) et sont composés d'un ensemble de cours. A noter que des étudiants de différents parcours peuvent suivre un même cours qui doit correspondre au niveau du parcours associé.

Ces mêmes cours chaque année ont un enseignant responsable, un coefficient et la moyenne des notes des étudiants. Cette moyenne sera calculée en fin d'année par la moyenne des notes obtenues par les étudiants à ce cours. La moyenne des notes d'un étudiant pondérée par les coefficients sera calculée pour donner une mention à son parcours (Refusé/Passable/Assez bien/Bien/Très bien).

Afin de gérer la réservation des salles pour chaque TD d'un parcours, des groupes sont créés dans lesquels seront répartis les étudiants. Les séances de cours forment un emploi du temps qui se verront associer la plage horaire, le type (cours/TD/TP) et une salle attribuée. Les salles sont localisées par un étage et un numéro dans un bâtiment avec une capacité maximale et un type (Amphi, TD, TP).

Les enseignants interviennent dans différents cours, il faut prendre en compte, chaque année, leur service en nombre d'heures de cours, de TD et de TP qu'il effectue. Un enseignant peut être titulaire, moniteur ou vacataire

1.1.1 Donner les acteurs principaux de ce texte ;

1.1.2 Pour chaque acteur, définir leurs caractéristiques ;

1.2 Modèle Entité/Association

1.2.1 Créer le schéma Entité/Association correspondant ;

1.3 Schéma relationnel

1.3.1 Transformer le schéma E/A en schéma relationnel ;

1.3.2 Définir les dépendances fonctionnelles dans chaque Relation ;

1.3.3 Normaliser 3^e Forme Normale ;

1.4 Contraintes

1.4.1 Donner la clé primaire de chaque relation ;

1.4.2 Donner la/les clés étrangères de chaque relation ;

1.4.3 Donner les contraintes de domaines spécifiques ;

1.5 LDD : Langage de Description de Données

1.5.1 Donner les requêtes SQL de création de tables pour : Etudiant, Parcours et Inscription ;

2 MISE EN PRATIQUE AVEC *MySQL Workbench*

Pour la suite des TPs : l'installation de MySQL et MySQL Workbench veuillez suivre l'installation disponible sur [Claroline](#). Idem pour la connexion à la base de données.

2.1 Modélisation

Créer le modèle des tables à partir du schéma Entité/Association réalisé à la question 1.2. Pour cela, il faut ouvrir un nouveau modèle (Menu "File/New Model"), une fois la fenêtre ouverte double-click sur "Add diagram". Vous pouvez maintenant créer les entités et les cardinalités qui les relient.

2.2 Création de la base de données


Générer les scripts de création de la Base de Données ('File/Export/Forward Engineer SQL Script'). Enregistrer le fichier, puis fermer le modèle.

Se connecter à la base MySQL (click sur votre instance), ouvrir le script généré ('File/Open SQL Script'). Exécuter le script (Bouton en forme d'éclair ou 'Query/Execute All').

3 INTERROGATION DE LA BASES DE DONNÉES

3.1 Chargement des données

Afin de travailler sur une base de données homogènes, nous travaillerons sur la base de données dont voici le script de création se trouve le fichier 'BDD_Ecole.sql' de l'espace de téléchargement du cours. Une fois le script exécuté, veuillez exécuter le script d'insertion des données 'BDD_Ecole_insert.sql'.

Pour tester une requête SQL sur la base de données, ouvrir un onglet 'SQL', écrire la requête SQL, puis exécuter avec le bouton 'execute All' .

3.2 Algèbre et SQL

Pour chacune des questions suivantes, produire l'expression algébrique correspondante puis la requête SQL. Tester ensuite la requête SQL sur la base de données.

- 3.2.1 Donner la liste des noms d'enseignants ;
- 3.2.2 Donner la liste des noms et prénoms des étudiants nés avant le '01/01/1990' ;
- 3.2.3 Donner le nom des enseignants responsables d'un cours ;
- 3.2.4 Donner les notes obtenues par l'étudiant 'Kevin Dupont' en 2014 ;
- 3.2.5 Donner le nom et le status des enseignants dont le service (nombre d'heures enseignées) dépasse 200 heures ;


3.3 SQL

Pour chacune des requêtes suivantes, donner la requête SQL correspondante et la tester sur la base de données.


- 3.3.1 Donner l'intitulé des parcours de L1 ;
- 3.3.2 Donner les noms des inscrits en 2014 ;
- 3.3.3 Donner les intitulés des enseignements du parcours de L3 en Informatique ;
- 3.3.4 Donner le nom des enseignants qui n'enseignent pas dans un cours dont ils sont responsables ;
- 3.3.5 Donner les intitulés distincts de cours dans lesquels intervient Mr 'Dupré'. Proposer deux manières de le faire ;
- 3.3.6 Donner les moyennes pour chaque année d'inscription de l'étudiant 'Kevin Dupont'. Ordonner le résultat par date croissante ;
- 3.3.7 Donner le nombre de séances de cours de 'Bases de Données' ayant lieu dans un Amphi ;
- 3.3.8 Donner le nom des étudiants qui ont réussi à toutes les matières (toutes les notes au dessus de la moyenne et pas de valeurs nulles) ;
- 3.3.9 Donner les noms des enseignants qui n'interviennent pas dans un parcours de M2 (2 requêtes SQL différentes) ;

- 3.3.10 Donner les noms des enseignants qui interviennent dans au moins un parcours de M2 et au moins un parcours de M1 ;
- 3.3.11 Donner la dernière année d'inscription de l'étudiant 'Guillard'. Donner deux manières de le faire ;
- 3.3.12 Donner le nombre d'étudiants inscrits en M1 en 2014 ;
- 3.3.13 Donner le nom des enseignants distincts qui sont responsables de plus d'un cours. Donner deux manières de le faire ;
- 3.3.14 Donner la moyenne pondérée (coeff) des notes de l'étudiant 'Durand' en 2014 (sans utiliser la table Inscription) ;
- 3.3.15 Donner le nom des étudiants dont ayant plus de 4 notes supérieures à 15 dans toute leur scolarité ;
- 3.3.16 Donner les intitulés des enseignements se trouvant dans plus de 2 parcours différents ;
- 3.3.17 Donner le nom de l'enseignant qui a fait le plus grand service d'enseignement (sommes des heures) en 2013 ;
- 3.3.18 Donner les intitulés des cours ayant lieu dans le bâtiment A au premier étage le '2014-10-15' ;
- 3.3.19 Donner pour chaque parcours de 2013, le nombre d'étudiants, la moyenne et l'écart-type de leurs notes ;
- 3.3.20 Donner le nom des étudiants ayant suivi tous les parcours d'informatique de l'école ;

4 PROGRAMMATION DANS LA BASE DE DONNÉES

Afin de créer une procédure sur la base de données, il faut sélectionner le schéma (double-click sur la base de données) puis appuyer sur le bouton 'Create Stored Procedure'  (ou bouton droit sur l'item 'Stored Procedure' du menu 'Schemas' - en bas à gauche).

Une fois la procédure créée (sans erreurs), appliquer les modifications avec le bouton 'Apply' (en bas à droite du champ textuel). La syntaxe des procédures stockées sous MySQL peut être trouvée ici : <http://dev.mysql.com/doc/refman/5.0/fr/stored-procedure-syntax.html>.

Pour exécuter une procédure, il suffit de mettre dans une fenêtre 'SQL script' le mot-clé 'CALL' avec le nom de la procédure et ses paramètres. Le bouton 'Execute All'  permet d'exécuter la requête.

Pour chaque procédure créée, tester le résultat.

4.1 Procédure simple

- 4.1.1 Créer une procédure 'affEtudiants' qui affiche la liste des noms et prénoms d'étudiants inscrits d'une année donnée en paramètre ;
- 4.1.2 Créer une procédure 'affNotes' qui affiche les notes avec l'intitulé du cours correspondant pour un étudiant (nom et prénom) et année donnée ;
- 4.1.3 Créer une procédure 'affMatières' qui affiche les intitulés et coefficients des matières qu'un étudiant (nom et prénom) suit durant une année donnée. Trier les matières par coefficients décroissants ;

4.2 Fonction

- 4.2.1 Créer une fonction 'retEtudiant' qui retourne l'identifiant d'un étudiant pour un nom et prénom donné ;
- 4.2.2 Créer une fonction 'moyennePonderee' qui retourne la moyenne pondérée des résultats d'un étudiant (utiliser la fonction 'retEtudiant') et année donnée ;
- 4.2.3 Donner la procédure qui affiche pour le parcours M1 'Informatique' de 2014, le nom des étudiants inscrits et leur moyenne pondérée en utilisant la fonction 'moyennePonderee' ;

4.3 Curseur et Exception

- 4.3.1 Modifier la fonction 'retEtudiant' qui retourne une exception si aucun étudiant n'est trouvé ;
- 4.3.2 Modifier les procédures 'affNotes' et 'affMatières' en utilisant la fonction 'retEtudiant' et afficher un message d'erreur lorsqu'une exception est lancée ;
- 4.3.3 Créer une procédure 'affBulletins' qui pour un étudiant donné, affiche l'intitulé de chaque parcours (avec l'année d'inscription croissante) qu'il a suivi et pour chaque parcours les résultats obtenus pour chaque matière (avec intitulé) ;

- 4.3.4 Créer une procédure 'affInterventions' qui affiche pour chaque niveau de parcours croissant les intitulés de cours et de parcours dans lequel intervient un enseignant donné (nom et prénom);
- 4.3.5 Créer une procédure qui met à jour toutes les moyennes des inscriptions (utiliser la fonction 'moyennePondere') des étudiants d'un parcours donné (id);

4.4 Procédure complexe

- 4.4.1 Créer une procédure 'affCours' qui affiche les informations d'un cours à une date donnée selon le schéma suivant :
HeureDeb, HeureFin, Salle/Batiment, Groupe1/Groupe2/Groupe3...
- 4.4.2 Créer une procédure 'affPlanning' qui affiche le planning d'un parcours donné (niveau et intitulé), organisé par enseignement ;

Afin aller sur la fenêtre de création d'un TRIGGER pour une table, il faut sélectionner la table dans la base de données (menu schéma en bas à gauche), faire click-droit sur la table et '*Alter Table*' et choisir l'onglet '*Triggers*' (en bas de la fenêtre).

Pour créer le Trigger, sélectionner le mode de déclenchement (AFTER DELETE/AFTER UPDATE/AFTER INSERT...) et cliquer sur '*Add Trigger*'. Il faut ensuite préciser le nom du Trigger et la procédure à exécuter. Une fois le Trigger terminé, cliquer sur '*Apply*'.

5.1 Trigger

On souhaite créer ajouter à la base de données les contraintes qui ne sont pas exprimable sous formes de clés primaires, clés étrangères, types, domaines de valeurs ou CHECK. Créer les Trigger correspondant aux contraintes suivantes :

Contraintes particulières non déduisible du schéma :

- Le responsable d'un enseignement est un enseignant titulaire ;
- L'étage d'une salle ne peut être supérieur au nombre d'étages du bâtiment ;
- On ne peut avoir de séances de 'Cours' dans une salle de 'TP', et on ne peut avoir de 'TP' que dans des salles de 'TP' ;
- Deux séances de cours ne peuvent avoir lieu dans la même salle en même temps ;
- Le total cumulé des séances de cours ne peut dépasser le nombre d'heures maximum d'un enseignement ;

Pour chaque Trigger créé, testé le avec une insertion d'un enregistrement provoquant l'erreur.

6 CRÉATION D'UNE APPLICATION BD

La documentation relative à la connexion à votre Base de Données est disponible dans le document “MySQL Connector/J Developer’s Guide” (site du cours).

Nous cherchons ici à développer une application gérant différents besoins du système d’information :

- Rôle étudiant : Consultation du planning ;
- Rôle Enseignant : Consulter son planning, consulter son service ;
- Rôle Administratif : Consulter les parcours, les étudiants, ajouter/modifier une note ;

6.1 Connexion à la base de données

Créer un objet gérant la connexion à la base de données. Il vous servira dans chacun des programmes suivants.

Entraînement : Créer un programme affichant la liste des intitulés de parcours.

6.2 Rôle étudiant

Créer une interface graphique qui affiche l’emploi du temps (tableau) d’un parcours (paramètre d’entrée avec année) à partir des informations suivantes :

- Sélectionner un parcours \Rightarrow liste de tous les cours de ce parcours, trié par date+heure.
Ne pas oublier d’afficher l’intitulé du cours et la salle ;
- Sélectionner en plus un numéro de semaine (liste déroulante) ;

Conseil : (non obligatoire)

- Créer un objet stockant les informations d’un parcours (id, intitulé, niveau, cours) ;
- Créer un objet stockant les informations d’une séance de cours (id, intitulé, groupe(s), salle, type, jour, horaires) ;
- Créer un objet stockant les informations d’une journée de cours ;
- Créer un objet stockant les informations d’une semaine de cours ;
- Créer un objet qui interroge la base de données et récupère l’ensemble des cours du parcours pour une semaine donnée ;
- Afficher d’abord la liste complète de tous les cours ;
- Découper par journée ;

6.3 Rôle enseignant

Créer une interface graphique qui affiche les informations de l’enseignant avec son service (somme des heures enseignées), et un onglet permettant de consulter son planning (même design que pour l’étudiant) ;

Conseil : (non obligatoire)

- Créer un objet stockant les informations d'un enseignant (id, nom, prénom);
- Créer un objet stockant l'ensemble des services d'un enseignant (le relier à l'objet précédent);
- Créer un objet qui interroge la base de données et récupère l'ensemble du service d'un enseignant;
- Créer un objet qui interroge la base de données et récupère l'ensemble des cours dans lequel il intervient;
- Requête pour le service + Requête pour le planning (idem que pour parcours, mais avec idEnseignant)

6.4 Bonus : Rôle administratif

Cette vue n'est pas demandée en TP. Toutefois, vous pouvez la faire pour vous entraîner.

Créer une interface graphique permettant de créer un parcours et de modifier les informations de celui-ci (associer un enseignement, un enseignant, un passage);

Conseil :

- 6.4.1 Créer un objet de gestion d'un étudiant (informations, inscriptions, résultats);
- 6.4.2 Créer une interface graphique permettant de consulter les informations de l'ensemble des étudiants d'un parcours;
- 6.4.3 Créer une interface graphique permettant de modifier/ajouter/supprimer les informations concernant un étudiant (ou créer un étudiant);
- 6.4.4 Créer un objet de gestion d'un parcours (informations, enseignements, passages);