

## Master 1 Humanités numériques – Bases de données

### TD 1 : Modèle physique MariaDB

J. Darmont – <https://eric.univ-lyon2.fr/jdarmont/>

#### Préambule

**MariaDB** est un système de gestion de bases de données relationnelles (SGBD-R) qui permet la structuration, le stockage, la maintenance, la mise à jour et la consultation de données réparties dans plusieurs *tables*. Ces données peuvent être mises en lien les unes avec les autres.

Les tables stockent les données. Une table se présente comme un tableau dont les colonnes sont les *attributs* et les lignes les *n-uplets*. Toute table doit avoir une *clé primaire*.

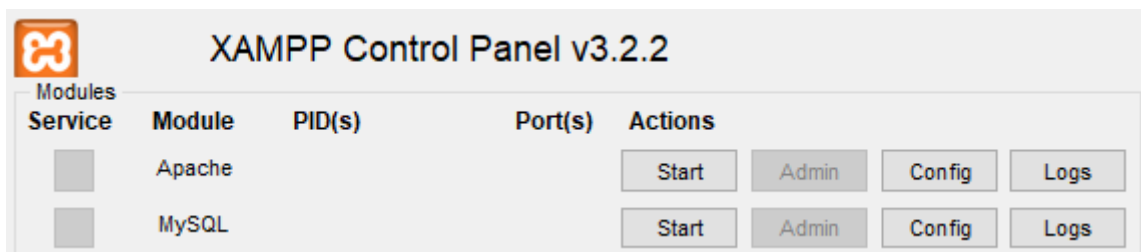
Une clé primaire est l'ensemble des attributs d'une table qui permet d'identifier de façon certaine et unique chaque n-uplet. Une clé primaire peut être composée d'un ou plusieurs attributs. Des valeurs identiques (doublons) ou manquantes (NULLES) ne sont pas autorisées dans les attributs composant une clé primaire.

**phpMyAdmin** est une interface web permettant d'accéder à MariaDB sans passer par des lignes de commandes.

MariaDB et phpMyAdmin sont inclus dans l'outil *open source* de développement web **XAMPP**<sup>1</sup>, qui fonctionne sous Linux, Windows et OS X.

#### Création d'une base de données

1. Lancer le panneau de contrôle de XAMPP, puis démarrer les serveurs Apache (qui permet l'exécution de phpMyAdmin) et MySQL (en fait, MariaDB) en cliquant sur les boutons « Start ».



2. Pour accéder à phpMyAdmin, utiliser un navigateur web et aller à l'adresse <http://127.0.0.1/phpmyadmin/> (qui est locale, tout est sur votre ordinateur).

<sup>1</sup> Téléchargement : <https://www.apachefriends.org/fr/index.html>

3. Pour créer une base de données, cliquer sur l'onglet « Base de données », choisir un nom (bd1, par exemple) et cliquer sur « Créer ». La nouvelle base de données apparaît dans l'explorateur à gauche de l'écran.

## Création de la structure d'une table


On souhaite créer la table ETUDIANT du cours. Cette table doit comprendre les attributs indiqués dans le dictionnaire des données ci-dessous.

Nom de l'attribut	Type de l'attribut	Taille	Description de la donnée
NumEtu	INT <sup>2</sup>		Numéro d'étudiant
Nom	VARCHAR <sup>3</sup>	50	Nom
Prénom	VARCHAR	50	Prénom
DateNaiss	DATE		Date de naissance
Rue	VARCHAR	50	Rue (adresse)
CP	VARCHAR	5	Code postal
Ville	VARCHAR	50	Ville (adresse)
NumCarte	INT		Numéro de carte IZLY
CodeGroupe	VARCHAR	3	Code du groupe de TD

*Dictionnaire des données*

1. Normalement, après la création de la base de données, vous devriez être dans l'onglet « Structure ». Si ce n'est pas/plus le cas, cliquer sur « Structure ». Renseigner le nom de la table (etudiant) et le nombre de colonnes/attributs (9), puis cliquer sur « Exécuter ».

2. À l'aide du dictionnaire des données, renseigner pour chaque attribut son nom, son type et sa taille quand c'est nécessaire. Les étudiants étant identifiés par leur numéro d'étudiant, renseigner également pour NumEtu la rubrique « Index » avec la valeur PRIMARY (pour *primary key*, clé primaire). Dans la boîte de dialogue, cliquer directement sur « Exécuter ». Si vous souhaitez inclure la description d'un attribut dont le nom n'est pas clair, renseigner la rubrique « Commentaires ». Cliquer sur « Enregistrer ». Vous devriez obtenir le schéma suivant et voir apparaître dans l'explorateur à gauche le nom de la table.

Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires
<u>NumEtu</u> 	int(11)			Non	Aucun(e)	Numéro d'étudiant
Nom	varchar(50)	latin1_swedish_ci		Non	Aucun(e)	
Prenom	varchar(50)	latin1_swedish_ci		Non	Aucun(e)	
<u>DateNaiss</u>	date			Non	Aucun(e)	Date de naissance
Rue	varchar(50)	latin1_swedish_ci		Non	Aucun(e)	
<u>CP</u>	varchar(5)	latin1_swedish_ci		Non	Aucun(e)	Code postal
Ville	varchar(50)	latin1_swedish_ci		Non	Aucun(e)	
<u>NumCarte</u>	int(11)			Non	Aucun(e)	Numéro de carte Izly
<u>CodeGroupe</u>	varchar(3)	latin1_swedish_ci		Non	Aucun(e)	Groupe de TD

2 Nombre entier ; utiliser FLOAT pour un nombre réel.

3 Chaîne de caractères

## Modification de la structure d'une table

1. Nous allons maintenant ajouter un attribut à la table ETUDIANT. Sous la structure de la table, renseigner « Ajouter une colonne après NumEtu » et cliquer sur « Exécuter ».

 Ajouter  colonne(s)

2. Ajouter un attribut « Civilité » de type « VARCHAR », de taille 8 et de valeur par défaut « Tel que défini »/« Madame ».

3. Cliquer sur l'onglet « Structure » pour voir le résultat.

## Saisie et modification de données

1. Nous allons maintenant saisir les données ci-dessous. Cliquer sur l'onglet « Insérer ». Il est possible de saisir les n-uplets un par un (cliquer sur le premier bouton « Exécuter ») ou deux par deux (cliquer sur le second bouton « Exécuter »). Attention : les dates sont au format américain. Recommencer l'opération jusqu'à la saisie complète, puis cliquer sur l'onglet « Parcourir » pour visualiser les données.

NumEtu Numéro d'étudiant	Civilité	Nom	Prenom	DateNaiss Date de naissance	Rue	CP Code postal	Ville	NumCarte Numéro de carte Izly	CodeGroupe Groupe de TD
1000	Monsieur	OUEST	Robert	1951-12-10	Grande Rue de Vaise	69009	Lyon	4000	A
1001	Madame	BREL	Madeleine	1980-12-15	Boulevard Mermoz	69008	Lyon	4001	B
1002	Monsieur	PAQUET	Didier	1981-01-01	Avenue du Général de Gaulle	69530	Brignais	4002	B
1003	Monsieur	QUANTI	Tutti	1983-06-01	Avenue Pierre Mendès France	69676	Bron	4003	A
1004	Madame	PERSONNE	Paula	1976-04-10	Parc de la Tête d'Or	69006	Lyon	4004	A
1005	Madame	MATHEI	Mimi	1982-09-07	Quai Claude Bernard	69007	Lyon	4005	B

2. Modifier le nom de Madame Mathei en Mattei à l'aide du lien « Éditer » à gauche de la ligne.

3. De même, modifier le numéro d'étudiant de Monsieur Ouest à 1001. Que se passe-t-il ? Pourquoi ?

## Importation de données

Nous allons importer dans la base de données les tables CARTE\_IZLY, GROUPE\_TD, EPREUVE, MATIERE et PASSER vues en cours.

1. À l'aide d'un navigateur web, télécharger le script SQL « etu-import » à l'adresse suivante.

<https://eric.univ-lyon2.fr/jdarmont/docs/etu-import.sql>

2. Pour effectuer l'importation, utiliser l'onglet « Importer » (conserver les paramètres par défaut) ou copier/coller le contenu du script SQL dans l'onglet « SQL ». Dans les deux cas, cliquer ensuite sur « Exécuter ». Les nouvelles tables doivent apparaître dans l'explorateur.

3. Dans la table EPREUVE, le champ qui permet de savoir de quelle matière une épreuve relève est CodeMat. C'est une *clé étrangère* qui fait référence au champ CodeMat de MATIERE. Ce champ étant NULL (pas de valeur) dans EPREUVE, il faut le compléter en vous aidant du tableau ci-contre pour effectivement créer ce lien au

niveau des **valeurs** contenues dans les tables. Parcourir la table MATIERE peut grandement aider !

Code d'épreuve	Matière correspondante
ECO1	Économie
ECO2	Économie
GEST1	Gestion
INFO1	Informatique
STAT1	Statistique

## Clés étrangères

Saisir dans la table EPREUVE l'enregistrement suivant, pour lequel la matière n'existe pas dans la table MATIERE (il est possible de le vérifier).

```
<CodeEpreuve = 'BIDON', Date = '2013-12-25',  
  Lieu = 'Amphi Say', CodeMat = 'MATH'>
```

Cela fonctionne quand même, bien que la matière de code MATH n'existe pas ! L'intégrité des données de la base n'est pas respectée car la base de données exemple n'inclut pas (encore) la définition explicite des clés étrangères.

Il s'agit donc définir les clés étrangères suivantes, au niveau de la **structure** de la base de données : NumCarte et CodeGroupe (table ETUDIANT), CodeMat (table EPREUVE), NumEtu et CodeEpreuve (table PASSER).

1. Commençons par la clé étrangère NumEtu de la table PASSER, qui référence l'attribut NumEtu dans la table ETUDIANT. Afficher la structure de la table PASSER, puis cliquer sur le bouton « Vue relationnelle » en haut. Renseigner un nom pour la clé étrangère (NumEtu\_FK, par exemple), l'attribut (colonne) concerné (NumEtu), puis la table (ETUDIANT) référencée. La clé primaire d'ETUDIANT devrait être sélectionnée automatiquement. Cliquer sur « Enregistrer » pour valider l'opération.

2. Procéder de même pour les clés étrangères CodeEpreuve (PASSER) → CodeEpreuve (EPREUVE), NumCarte (ETUDIANT) → NumCarte (CARTE\_IZLY) et CodeGroupe (ETUDIANT) → CodeGroupe (GROUPE\_TD).

3. Procéder de même pour établir la relation CodeMat (EPREUVE) → CodeMat (MATIERE). Une erreur survient car les données de la table EPREUVE ne respectent pas l'intégrité référentielle : un code d'épreuve qui n'existe pas dans MATIERE est présent dans EPREUVE (la matière MATH de l'épreuve BIDON).

4. Supprimer l'enregistrement correspondant à l'épreuve avec matière invalide dans la table EPREUVE (épreuve de code BIDON) et répondre à nouveau à la question 3.

5. Essayer de saisir à nouveau dans la table EPREUVE l'enregistrement invalide. Conclusion ?

```
<CodeEpreuve = 'BIDON', Date = '2013-12-25',  
  Lieu = 'Amphi Say', CodeMat = 'MATH'>
```

6. Dans l'explorateur de gauche, cliquer sur le nom de la base de données, puis activer l'onglet « Concepteur ». Le résultat est le *modèle physique* de la base de données présenté sous forme graphique (cf. ci-contre).

