

## TP1 : Modèle Entité-association. Introduction au langage SQL.

Ce TP s'articule en trois phases : une première phase de conception des modèles EA et une phase de manipulation de votre première base de données. Les questions, au nombre de 4, sont indiquées en *bleu*.

### Partie 1. Modèle EA

Vous utiliserez un logiciel de traitement de texte (openoffice ou autre) pour rédiger votre rapport. Le fichier devra porter vos noms et prénoms au format suivant : "NON\_PRENOM.extension". L'extension pourra être "doc", "docx", "odt" ou autre.

#### 1. Université.

Le système informatique d'une université utilise les données suivantes :

- Pour chaque étudiant : son numéro de matricule, son nom, son prénom et son adresse (rue, numéro, cp, ville).
- Pour chaque cours : le code administratif, l'intitulé et un petit résumé.
- Pour chaque professeur : son numéro de matricule, son titre, son nom, son prénom, et son adresse.

De plus le système doit savoir quels sont les cours que chaque étudiant suit, le professeur titulaire de chaque cours, la filière choisie par chaque étudiant et pour chaque filière (dont on retient le nom et le code) le professeur la dirigeant.

**QUESTION 1:** Donner un modèle entité-association de ce système et préciser les contraintes d'intégrité (cardinalité), ainsi que les rôles de chaque entité dans les relations.

#### 2. Pré-requis.

**QUESTION 2 :** Ajouter à votre modélisation précédente le fait que chaque cours nécessite de zéro à plusieurs pré-requis, et préciser les nouvelles contraintes d'intégrité si nécessaire.

Indication : un pré-requis représente un cours antérieur.

Effectuez les modifications sur un nouveau schéma.

# Partie 2. Introduction à l'environnement phpMyAdmin

## 1. SGBD.

Qu'est ce qu'un Système de Gestion de Bases de Données ?

Un SGBD est un logiciel qui prend en charge la structuration, le stockage, la mise à jour et la maintenance d'une base de données. Il est l'unique interface entre les informaticiens et les données (définition des schémas, programmation des applications), ainsi qu'entre les utilisateurs et les données (consultation et mises à jour).

Parmi les logiciels les plus connus, il est possible de citer : MySQL, PostgreSQL, Oracle Database, SQLite, Microsoft SQL Server, ...

Ces systèmes peuvent être catégorisés selon leur fonctionnement :

- **Système propriétaire** : Oracle Database, Microsoft SQL Server, DB2, MaxDB, 4D, dBase, Informix, Sybase
- **Système libre** MySQL, PostgreSQL, MariaDB, Firebird, Ingres, HSQLDB, Derby
- **Orienté objet** : ZODB, db4o
- **Embarqué** : SQLite, Berkeley DB
- **NoSQL** : Cassandra, Redis, MongoDB, SimpleDB, BigTable, CouchDB, HBase, LevelDB, RethinkDB, Memcached
- **Autre système** : Access, OpenOffice.org Base, FileMaker, HyperFileSQL, Paradox, Neo4j

## 2. phpMyAdmin

Les exercices de TP seront basés sur *phpMyAdmin*. *PhpMyAdmin* (PMA) est une [application Web](#) de gestion pour les [systèmes de gestion de base de données MySQL](#) réalisée en [PHP](#) et distribuée sous licence [GNU GPL](#).

# PARTIE 3. Introduction au langage SQL.

## 1. SQL

Le **SQL** (Structured Query Language) est un langage permettant de communiquer avec une base de données. Ce langage informatique est notamment très utilisé par les développeurs web pour communiquer avec les données d'un site web. Le SQL présente trois caractéristiques :

- le SQL est un langage de définition de données. Il permet de créer des tables dans une base de données relationnelle, ainsi que de d'en modifier ou en supprimer
- le SQL est un langage de manipulation de données. Cela signifie qu'il permet de sélectionner, insérer, modifier ou supprimer des données dans une table d'une base de données relationnelle.
- le SQL est un langage de protections d'accès. Il est possible de définir des permissions au niveau des utilisateurs d'une base de données. On parle alors de DCL (Data Control Language).

Le langage SQL se compose en une série d'instructions. Chaque instruction est composée d'un verbe d'action et d'une série d'option. Comme dans la plupart des langages de programmation, chaque instruction se termine par une ponctuation. En SQL, la ponctuation de fin d'instruction (ou fin de phrase) est le point-virgule.

## 2. Modèle EA Université

Reprenons le problème de la PARTIE 1.

**QUESTION 3** : A l'aide de la commande "create", créer chacune tables correspondant à chacune des entités du problème. Chacun des attributs sera stocké soit sous forme d'entier (`int`), soit sous formes d'une chaîne de caractère (`varchar(dim)`). Nous considérerons qu'il n'y a pas de clé primaire pour l'instant.

Indication1 : La syntaxe de création de table est du type

```
CREATE TABLE TAB(VAR1 TYPE1, VAR2, TYPE, .... );
```

A tout moment, pour visualiser le nombre de groupe d'entités(table) dans la base de données, nous pouvons utiliser la commande :

```
show tables;
```

**QUESTION 4** : A l'aide de la commande d'insertion "insert", ajouter au moins deux enregistrements à chacune des tables.

*Indication3* : la syntaxe de l'insertion est du type "INSERT INTO TAB(ATT2, ATT1) VALUES (V1, V2); ". On suppose ici que la table relationnelle TAB comprend deux colonnes ATT1 et ATT2. On veut insérer deux valeurs V1 et V2.

*Indication4* : A tout moment, vous pouvez interroger le serveur sur le contenu d'une table.

```
SELECT * FROM NOM_TABLE;
```

Pendant le TP :

- ✓ Créer un rapport "NOM\_PRENOM.extension" dans un répertoire TP1\_NOM
- ✓ Mettre toutes les réponses dans le rapport
- ✓ Compresser le répertoire TP1\_NOM (zip)
- ✓ envoyer le répertoire compresser par mail à l'adresse *nargesse.merad@univ-amu.fr*