BDD (fiche 2) – Langage SQL

**Exercice 1**

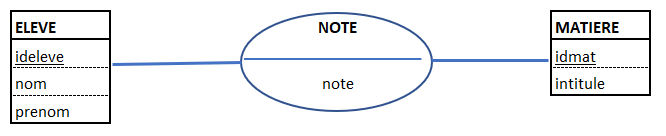
Créer une table annuaire avec nom et prenom string de longueur <=100 et avec pour clef primaire le numéro (string <=20). Les attributs nom et prenom ne doivent pas être null.

**Exercice 2**

Donner le code SQL créant ces tables

Les clefs primaires sont des string <=20.

**Exercice 3**

****Donner les commandes SQL de création des 3 tables sachant que :  
- nom, prenom et intitule sont des string <=100  
- idmat est un entier  
- ideleve est une string <=20  
- la note est un décimal du type ab,cd  
- la note doit être comprise entre 0 et 20

Donner les commandes détruisant cette BDD.

**Exercice 4**

On reprend le MLD suivant :

**ARTICLE(idarticle,nom,poids,idcatégorie)**

**CATEGORIE(idcategorie,libellé)**

**MAGASIN(idmagasin , nom , adresse )**

**STOCK( idarticle, idmagasin,quantité)**

Ecrire les commandes SQL créant cette BDD. Les clefs primaires sont des entiers.  
quantité doit être un entier positif ou nul

**Exercice 5**

Dans chacun des **quatre** cas suivants, trouver et expliquer l'erreur.

DROP TABLE client;

CREATE TABLE client(cid INT PRIMARY KEY, nom VARCHAR(100),points INT not null, CHECK(points >= 0));

CREATE TABLE client(cid INT PRIMARY KEY, nom VARCHAR(100),points INT not null, CHECK(points >= 0));

CREATE TABLE commande(cid INT REFERENCES client(cid), pid INT REFERENCES produit(pid), date DATE not null);

CREATE TABLE produit(pid INT PRIMARY KEY, nom VARCHAR(100), prix DECIMAL(10,2));

CREATE TABLE client(cid INT PRIMARY KEY, nom VARCHAR(100),points INT not null, CHECK(points >= 0));

CREATE TABLE produit(pid INT PRIMARY KEY, nom VARCHAR(100), prix DECIMAL(10,2));

CREATE TABLE commande(cid INT REFERENCES client(cid), nomp VARCHAR(100) REFERENCES produit(nom), date DATE not null);

CREATE TABLE client(cid INT PRIMARY KEY, nom VARCHAR(100),points INT not null, CHECK(points >= 0));

CREATE TABLE produit(pid INT PRIMARY KEY, nom VARCHAR(100), prix DECIMAL(10,2));

CREATE TABLE commande(cid INT REFERENCES client(cid), pid INT REFERENCES produit(pid), date DATE not null);

INSERT INTO commande(0,0,'2020-03-02')

**Exercice 6**

On considère les deux tables suivantes

CREATE TABLE joueur (jid INT PRIMARY KEY, nom VARCHAR(100) NOT NULL);

CREATE TABLE partie (j1 INT REFERENCES joueur(jid), j2 INT REFERENCES joueur(jid), score1 INT NOT NULL,   
 score2 INT NOT NULL, CHECK( (j1<>j2) );

Décrire cette BDD

Rappeler ce qu'est une contrainte d'entité. Donner des ordres SQL violant cette contrainte. Rappeler ce qu'est une contrainte de domaine. Donner un ordre SQL violant cette contrainte. Rappeler ce qu'est une contrainte de référence. Donner un ordre SQL violant cette contrainte. Rappeler ce qu'est une contrainte utilisateur. Donner un ordre SQL violant cette contrainte.

**Exercice 7** Modifier la déclaration précédente (exercice 6) sachant que :  
- la table partie contient un attribut (date) non null.  
- les scores ne peuvent pas être négatifs  
- deux joueurs ne peuvent pas jouer plusieurs fois le même jour.

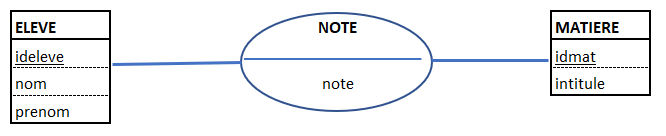
**Exercice 8**

Proposer un MCD d'un bulletin scolaire. On devra y trouver des élèves (nom et prénom de type str) possédant un numéro d'identifiant unique (str) , un ensemble de matières (str) identifiées par un entier, au plus une note (float) sur 20 par élève et par matière.

Ecrire les requêtes SQL créant cette BDD

**Exercice 9**

Une image contenant table

Description générée automatiquementA l'aide de Python, créer la BDD   
bulletin.db suivante :

Et les enregistrements suivants :

Une image contenant table

Description générée automatiquement**Exercice 10**

Créer la table suivante :

La colonne valeur1 est égale à 

La colonne valeur2 est égale à 

La colonne valeur contient successivement "pierre", "paul","jacques"

Il y a 100 enregistrements.