Correction exercices base de données

Exercice 1 :

1. proposer une modélisation relationnelle de cet annuaire

ANNUAIRE (numTel : text, nom : text, prenom : text)

On définit le champ numTel comme text pour gérer les numéros de téléphone commençant par 0.

1. Est ce une relation valide ?
   1. Valide. L’ensemble vide est valide pour la relation ANNUAIRE
   2. Pas valide. NumTel est la clé primaire, les valeurs pour cet attribut doivent être unique et non nul. Ici la valeur ‘0123456789’ est dupliquée sur les 2 occurrences.
   3. Valide. Il n’y a pas contraintes utilisateur sur la longueur du champ numTel.
   4. Non valide. Le champ numTel ne doit pas être vide.
   5. Non valide. La contrainte de domaine sur le champ numTel est violé. Le champ doit être de type text, or ici on a un entier.

3) écrire l’instruction de création et d’insertion dans la relation ANNUAIRE

CREATE TABLE ANNUAIRE (

numTel TEXT PRIMARY KEY,

nomTEXT,

prenom TEXT);

INSERT INTO ANNUAIRE

(numTel, nom,prenom)

VALUES

(‘0123456789’, ‘titi’, ‘toto’),

(‘987456120325588’,’titi’,’toto’);

Exercice 2 :

1. dessiner le schéma relationnel décrit précédemment

2) écrire les requêtes de création des relations LIVRES et ECRITS

CREATE TABLE LIVRES (

Id INTEGER PRIMARY KEY,

Titre TEXT,

PrixHT FLOAT,

Annee INTEGER,

Idgenre INTEGER,

Idediteur INTEGER,

FOREIGN KEY idgenre REFERENCES GENRE(id),

FOREIGN KEY idediteur REFERENCES EDITEUR(id)

);

CREATE TABLE ECRITS (

IdAuteur INTEGER,

IdLivre INTEGER,

PRIMARY KEY (idAuteur, idediteur),

FOREIGN KEY idAuteur REFERENCES AUTEURS(id),

FOREIGN KEY idLivre REFERENCES LIVRES(id)

);

3) écrire les requêtes en langage SQL

* 1. SELECT titre FROM LIVRES ;
  2. SELECT Titre FROM LIVRES WHERE Titre LIKE ‘A%’;
  3. SELECT MAX(prixHT) FROM LIVRES ;
  4. SELECT COUNT(\*) FROM AUTEURS ; **ou** SELECT COUNT(id) FROM AUTEURS ;
  5. SELECT COUNT(\*), AVG(prixHT) FROM LIVRES ;
  6. SELECT nom, prenom FROM AUTEURS ORDER BY nom ASC ;
  7. SELECT titre, prixHT \* 1,055 FROM LIVRES WHERE prixHT < 15 ;
  8. SELECT titre, nom FROM LIVRES INNER JOIN EDITEURS ON LIVRES.idEditeur = [ÉDITEURS.id](http://xn--diteurs-9xa.id);
  9. SELECT titre FROM LIVRES INNER JOIN GENRE ON LIVRES.idgenre=[GENRE.id](http://GENRE.id) WHERE genre = ‘science’;
  10. SELECT SUM(prixHT) FROM LIVRES INNER JOIN GENRE ON LIVRES.idgenre=[GENRE.id](http://GENRE.id) WHERE genre = ‘policier’ AND annee = 2019;
  11. SELECT Id FROM LIVRES INNER JOIN AUTEUR ECRITS ON [LIVRES.id](http://LIVRES.id) = ECRITS.idLivre INNER JOIN AUTEURS ON ECRITS.idLivre =AUTEURS.Id WHERE prenom=‘Marie’;