1. Le titre du livre etISBN sont 2 fois, dans la table Livre et Emprunt or on sait qu’une base relationnelle est crée afin d’éviter les doublons. Cela alourdit donc inutilement la base de données
2. A. (Abonne\_ID, Livre\_ID, DateEmprunt)

b. Ce n’est pas possible que la date de retour soit avant la date d’emprunt du livre, il y aura donc une erreur.

c. On ne peut pas créer un identifiant d’emprunt a parti d’un identifiant abonné si lui même n’existe pas dans sa propre table c’est à dire la table abonné et inversement on ne peut pas supprimer un abonné de sa propre table s’il n’est pas dans la table Emprunt.

d. L’ajout de la contrainte suivante est une mauvaise idée car on ne pourra plus rentrer des emprunts sans entrer la date de retour. Quand on emprunte un livre on se sait pas exactement quand on va rendre le livre.

1. a. SELECT Nom, Prenom FROM Abonne WHERE Ville = 'Bayonne'

b. SELECT Nom, Prenom FROM Abonne WHERE CodePostal LIKE '64%'

c. SELECT Abonne\_ID FROM Emprunt WHERE DateRetourReel > DateRetourPrevu

d. SELECT Nom FROM Abonne JOIN Emprunt

ON Abonne.Abonne\_ID = Emprunt.Abonne\_ID WHERE DateRetourReel > DateRetourPrevu

e. SELECT Titre FROM Livre L JOIN Emprunt E ON L.Livre\_ID = E.Livre\_ID

WHERE DateRetourReel IS NULL

f. SELECT Nom FROM Abonne

JOIN Emprunt ON Abonne.Abonne\_ID = Emprunt.Abonne\_ID JOIN Livre ON Livre.Livre\_ID = Emprunt.Livre\_ID

WHERE Livre.Titre = 'Papillon'

g. SELECT \* FROM Abonne

WHERE NOT EXISTS (SELECT \* FROM Emprunt

WHERE Abonne.Abonne\_ID = Emprunt.Abonne\_ID)

h. SELECT Nom FROM Abonne WHERE Abonne\_ID NOT IN

(SELECT Abonne\_ID FROM Emprunt

WHERE DateRetourReel > DateRetourPrevu)

i.SELECT Nom FROM Abonne WHERE Abonne\_ID NOT IN

(SELECT Abonne\_ID FROM Emprunt

WHERE DateRetourReel <= DateRetourPrevu)

j. SELECT Nom, Prenom FROM Abonne

JOIN Emprunt ON Abonne.Abonne\_ID = Emprunt.Abonne\_ID WHERE Ville = 'Bayonne' AND DateRetourPrevu < DateRetourReel

k. SELECT \* FROM Abonne

WHERE DateNaissance = (SELECT Min(DateNaissance) FROM Abonne)

l. SELECT MAX(DateRetourReel-DateRetourPrevu) FROM Emprunt

m. SELECT Abonne\_ID FROM Emprunt

WHERE DateRetourReel-DateRetourPrevu = (SELECT MAX(DateRetourReel-DateRetourPrevu)

FROM Emprunt)

n. SELECT AVG(DateRetourReel-DateRetourPrevu) FROM Emprunt

WHERE DateRetourReel>DateRetourPrevu

NSI Lycée Louis de Foix

o. SELECT nom FROM Abonne WHERE nom BETWEEN 'A' AND 'I'

p. SELECT Count(\*)

FROM Emprunt E

JOIN Livre L ON L.Livre\_ID = E.Livre\_ID WHERE L.Titre = 'Papillon'

1. A. INSERT INTO Emprunt VALUES(803, 4252, 2020-11-02 , 2020-11-23, NULL)

b. UPDATE Abonne SET Adresse = '11 rue Lormand', Ville = 'Bayonne', CodePostal = '64100'

WHERE Nom = 'DUPONT' AND Prenom = 'Jean'

C. SELECT \* FROM Emprunt E JOIN Abonne A ON E.Abonne\_ID = A.Abonne\_ID WHERE Nom = 'ETCHEGARAY' AND Prenom = 'Jon' AND DateRetourReel IS NULL

d. DELETE FROM Emprunt

WHERE Abonne\_ID IN (SELECT Abonne\_ID

FROM Abonne

WHERE Nom = 'ETCHEGARAY' AND Prenom = 'Jon')

e. DELETE FROM Abonne WHERE Nom = 'ETCHEGARAY' AND Prenom = 'Jon'

1. SELECT \* FROM Emprunt WHERE NOT EXISTS

(SELECT 1 FROM Abonne WHERE Abonne.Abonne\_ID = Emprunt.Abonne\_ID)

OR NOT EXISTS (SELECT 1 FROM Livre WHERE Livre.Livre\_ID = Emprunt.Livre\_ID)