

# Corrigé – Bases de données

## Exercice 1

Manipulations logicielles ne donnant pas lieu à une correction.

## Exercice 2

Le numéro de sécurité social est suffisant pour identifier de manière unique un élève car il est unique en France pour chaque personne en recevant un.

## Exercice 3

L'unicité des informations n'est pas garantie car plusieurs vélos peuvent faire le même trajet aux mêmes heures de départ et d'arrivée.

Une clé primaire pourrait être définie à l'aide d'un identifiant du vélo utilisé ou d'un identifiant pour l'utilisateur du vélo ou encore de la combinaison d'un identifiant pour le vélo et d'un identifiant pour la personne l'utilisant.

## Exercice 4

On considère une base de données qui décrit des informations sur des articles et des clients qui achètent ces articles. Le schéma relationnel de cette base est le suivant, où la clé primaire de chaque table est soulignée et les attributs qui forment une clé étrangère sont suivis de \*.

- Achats (aid\*, cid\*, dateachat, quantité)
- Articles (aid, rayon, prix)
- Clients (cid, nom, datenaiss)

1. La réponse dépend du format de la date d'achat. Si on ne stocke que le jour, alors le client ne pourra pas acheter plusieurs fois le même article dans la même journée. En revanche, si on stocke aussi l'heure exacte d'achat, il sera possible d'acheter le même article plusieurs fois dans la même journée. L'essentiel est que le trio d'attributs formant la clef primaire soit unique.
2. Attention, il faut bien créer les tables Clients et Articles avant Achats car cette dernière fait référence aux deux premières (à cause des clefs étrangères).

```
CREATE TABLE "Clients" (  
    "cid"          INTEGER NOT NULL UNIQUE,  
    "nom"          TEXT,  
    "datenaiss"    TEXT,  
    PRIMARY KEY("cid" AUTOINCREMENT)  
);
```

```
CREATE TABLE "Articles" (  
    "aid"          INTEGER NOT NULL UNIQUE,  
    "rayon"        TEXT,  
    "prix"         TEXT,  
    PRIMARY KEY("aid" AUTOINCREMENT)  
);
```

```
CREATE TABLE "Achats" (
    "aid"          INTEGER,
    "cid"          INTEGER,
    "dateachat"    TEXT,
    "quantité"     INTEGER,
    PRIMARY KEY("aid","cid","dateachat"),
    FOREIGN KEY("aid") REFERENCES "Articles"("aid"),
    FOREIGN KEY("cid") REFERENCES "Clients"("cid")
);
```

- De même qu'on a créé les tables dans un certain ordre à la question précédente, il faut faire attention à l'ordre d'insertion des données. On ne pourra insérer des achats que si on a des clients et des articles. Il est donc nécessaire d'insérer les données de la table Achats en dernier, peu importe l'ordre d'insertion des données des deux autres.

### Exercice 5

- Une autre clef primaire pourrait être l'attribut email car une adresse de courrier électronique est personnelle et identifie donc de manière unique un client.
- On peut supprimer un enregistrement de la table Véhicule si le véhicule concerné n'est pas en cours de location, c'est-à-dire s'il n'est pas référencé en clef étrangère dans la table Location (attribut idVehi).
- `UPDATE Location SET dateFin = '2022-02-03 16:00:00' WHERE idL = 9268;`
- `SELECT immatr FROM Véhicule WHERE nbKm > 50000;`
- `SELECT COUNT(*) FROM Location WHERE dateFin IS NULL;`
- `SELECT COUNT(*) FROM Véhicule
 JOIN Modèle ON Modèle.idM = Véhicule.modele
 WHERE Modèle.moteur = 'électrique';`
- `SELECT DISTINCT email FROM Client
 JOIN Location ON idC = idCli
 JOIN Vehicule ON idV = idVéhi
 JOIN Modele ON idM = Modele
 WHERE moteur = 'électrique'
 AND dateDeb >= '2022-01-01 00:00:00'
 AND dateDeb < '2022-02-01 00:00:00';`