Assignment 1
Design an Online Bookstore

Database statement

PASSELEGUE Anne – ROELANDT Nicolas

This document defines the SQL schema for the e-commerce bookstore system.
It includes the following tables :

- `users`
- `sellers`
- `books`
- `settlements`
- `orders`
- `reviews`
- `favorites`
- `carts`
- `discounts`
- `coupons`

Users

This table stores information about all customers and administrators. It includes login details, personal data, and contact information.

```
CREATE TABLE users (
  id           INT NOT NULL AUTO_INCREMENT,
  is_admin     BOOLEAN NOT NULL,
  email        VARCHAR(255) NOT NULL,
  password     VARCHAR(255) NOT NULL,
  name         VARCHAR(100) NOT NULL,
  birth_date   DATE,
  gender       VARCHAR(20),
  address      VARCHAR(255),
  phone_number VARCHAR(30),
  created_at   DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (id),
  UNIQUE KEY uk_users_email (email)
);
```

Sellers

This table contains details about book sellers or publishing partners. It stores their business, contact, and payout information.

```
CREATE TABLE sellers (
  id              INT NOT NULL AUTO_INCREMENT,
  business_number VARCHAR(50) UNIQUE,
  business_name   VARCHAR(255) NOT NULL,
  email           VARCHAR(255) NOT NULL UNIQUE,
  phone_number    VARCHAR(50) NOT NULL,
  address         VARCHAR(255) NOT NULL,
  payout_bank     VARCHAR(100) NOT NULL,
  payout_account  VARCHAR(100) NOT NULL,
  payout_holder   VARCHAR(100) NOT NULL,
  PRIMARY KEY (id)
);
```

Books

This table keeps records of all books available in the store. It includes details such as title, author, price, and publication date.

```
CREATE TABLE books (
  id                 INT NOT NULL AUTO_INCREMENT,
  title              VARCHAR(255) NOT NULL,
  author             VARCHAR(200) NOT NULL,
  publisher          VARCHAR(200),
  summary            TEXT,
  isbn               VARCHAR(20) NOT NULL,
  price              DECIMAL(10,2) NOT NULL,
  publication_date   DATE,
  created_at         DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
  updated_at         DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,
  PRIMARY KEY (id),
  UNIQUE KEY uk_books_isbn (isbn),
  CHECK (price >= 0)
);
```

Settlements

This table manages financial settlements for each seller. It records total sales, commissions, and final payouts over specific periods.

```
CREATE TABLE settlements (
  id              INT NOT NULL AUTO_INCREMENT,
  seller_id       INT NOT NULL,
  total_sales     DECIMAL(12,2) NOT NULL,
  commission      DECIMAL(12,2) NOT NULL,
  final_payout    DECIMAL(12,2) NOT NULL,
  period_start    DATE NOT NULL,
  period_end      DATE NOT NULL,
  settlement_date DATETIME NOT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_settlements_seller
    FOREIGN KEY (seller_id) REFERENCES sellers(id)
    ON UPDATE NO ACTION ON DELETE CASCADE,
  CHECK (total_sales >= 0),
  CHECK (commission >= 0),
  CHECK (final_payout >= 0)
);
```

Orders

This table tracks all book purchase transactions made by users. It connects users, books, and settlements with order status and quantity.

```
CREATE TABLE orders (
  id               INT NOT NULL AUTO_INCREMENT,
  settlement_id    INT,
  book_id          INT NOT NULL,
  user_id          INT NOT NULL,
  status           VARCHAR(50) NOT NULL,
  quantity         INT NOT NULL,
  created_at       DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
  updated_at       DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,
  PRIMARY KEY (id),
```

```
  CONSTRAINT fk_orders_user
    FOREIGN KEY (user_id) REFERENCES users(id)
    ON UPDATE NO ACTION ON DELETE CASCADE,
  CONSTRAINT fk_orders_book
    FOREIGN KEY (book_id) REFERENCES books(id)
    ON UPDATE NO ACTION ON DELETE CASCADE,
  CONSTRAINT fk_orders_settlement
    FOREIGN KEY (settlement_id) REFERENCES settlements(id)
    ON UPDATE NO ACTION ON DELETE SET NULL,
  CHECK (quantity > 0)
);
```

Reviews

This table stores user feedback and ratings for books. Each review is linked to a specific user and book.

```
CREATE TABLE reviews (
  id         INT NOT NULL AUTO_INCREMENT,
  book_id    INT NOT NULL,
  user_id    INT NOT NULL,
  rating     INT NOT NULL,
  comment    TEXT,
  created_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,
  updated_at DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,
  PRIMARY KEY (id),
  CONSTRAINT fk_reviews_book
    FOREIGN KEY (book_id) REFERENCES books(id)
    ON UPDATE NO ACTION ON DELETE CASCADE,
  CONSTRAINT fk_reviews_user
    FOREIGN KEY (user_id) REFERENCES users(id)
    ON UPDATE NO ACTION ON DELETE CASCADE,
  CHECK (rating BETWEEN 1 AND 5)
);
```

Favorites

This table keeps track of books that users mark as favorites. It helps users easily access their preferred books later.

```
CREATE TABLE favorites (
  id       INT NOT NULL AUTO_INCREMENT,
  user_id  INT NOT NULL,
  book_id  INT NOT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_favorites_user
    FOREIGN KEY (user_id) REFERENCES users(id)
    ON UPDATE NO ACTION ON DELETE CASCADE,
  CONSTRAINT fk_favorites_book
    FOREIGN KEY (book_id) REFERENCES books(id)
    ON UPDATE NO ACTION ON DELETE CASCADE
);
```

Carts

This table manages items users add to their shopping cart before checkout. It records the selected books and quantities for each user.

```
CREATE TABLE carts (
  id        INT NOT NULL AUTO_INCREMENT,
```

```
  user_id   INT NOT NULL,
  book_id   INT NOT NULL,
  quantity  INT NOT NULL,
  PRIMARY KEY (id),
  CONSTRAINT fk_carts_user
    FOREIGN KEY (user_id) REFERENCES users(id)
    ON UPDATE NO ACTION ON DELETE CASCADE,
  CONSTRAINT fk_carts_book
    FOREIGN KEY (book_id) REFERENCES books(id)
    ON UPDATE NO ACTION ON DELETE CASCADE,
  CHECK (quantity > 0)
);
```

Discounts

This table defines temporary discounts applied to certain books. It includes the discount rate, duration, and validity status.

```
CREATE TABLE discounts (
  id              INT NOT NULL AUTO_INCREMENT,
  book_id         INT NOT NULL,
  discount_rate   DECIMAL(5,2) NOT NULL CHECK (discount_rate >= 0 AND
discount_rate <= 100),
  start           DATETIME NOT NULL,
  end             DATETIME NOT NULL,
  is_valid        BOOLEAN NOT NULL DEFAULT TRUE,
  PRIMARY KEY (id),
  CONSTRAINT fk_discounts_book
    FOREIGN KEY (book_id) REFERENCES books(id)
    ON UPDATE NO ACTION ON DELETE CASCADE
);
```

Coupons

This table stores promotional codes that users can apply for price reductions. Each coupon has a value, validity period, and unique code.

```
CREATE TABLE coupons (
  id              INT NOT NULL AUTO_INCREMENT,
  code            VARCHAR(50) NOT NULL UNIQUE,
  coupon_amount   DECIMAL(10,2) NOT NULL CHECK (coupon_amount >= 0),
  start           DATETIME NOT NULL,
  end             DATETIME NOT NULL,
  is_valid        BOOLEAN NOT NULL DEFAULT TRUE,
  PRIMARY KEY (id)
);
```