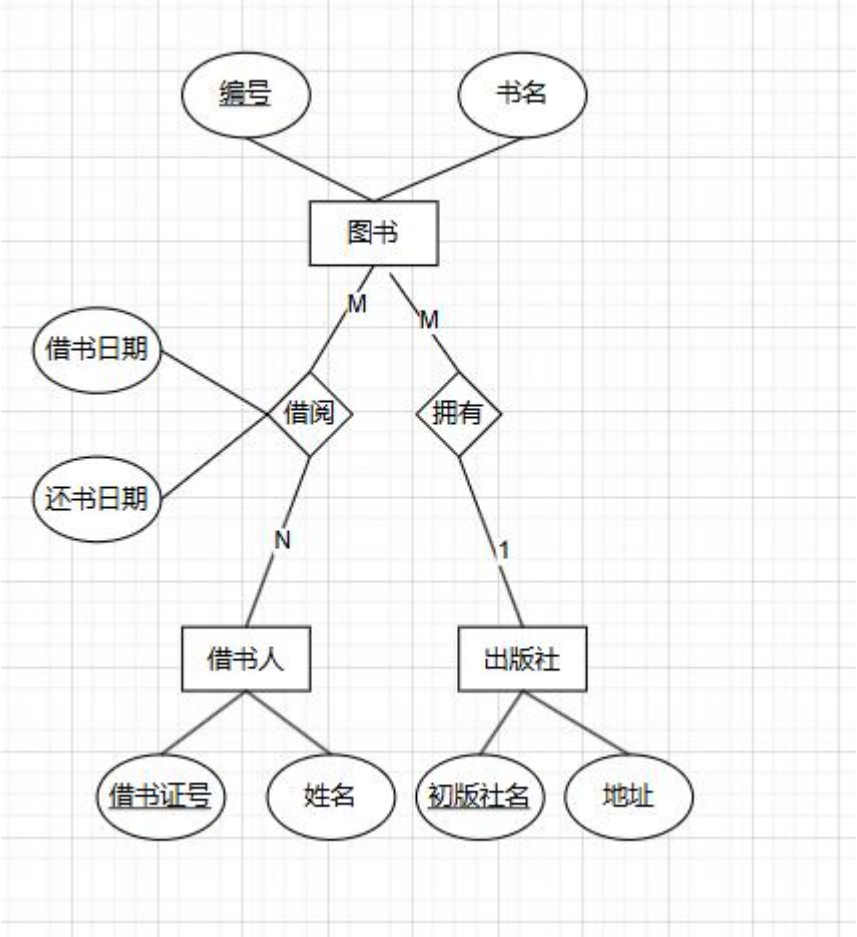
(1)（30 分）根据需求画出概念模型（E-R 图），并在图中注明实体的属性、关系类型以及约束。



(2)（20 分）将概念模型转换为逻辑模型和物理模型

借书人表 (Borrowers)

BorrowerID INT PRIMARY KEY

Name VARCHAR(100)

图书表（Books）

BookID INT PRIMARY KEY

Title VARCHAR(255)

PublisherName VARCHAR(255) FOREIGN KEY REFERENCES Publishers(PublisherName)

出版社表（Publishers）

PublisherName VARCHAR(255) PRIMARY KEY

Address VARCHAR(255)

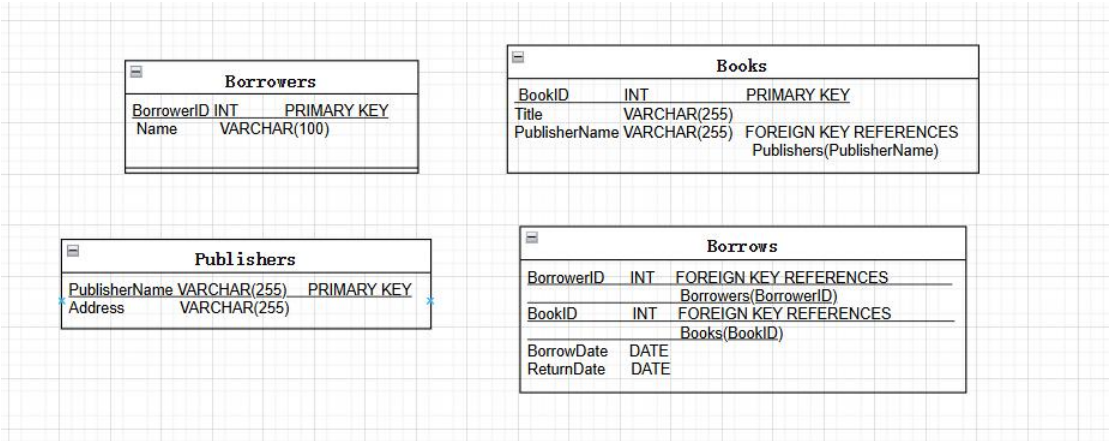
借阅表（Borrows）

BorrowerID INT FOREIGN KEY REFERENCES Borrowers(BorrowerID)

BookID INT FOREIGN KEY REFERENCES Books(BookID)

BorrowDate DATE

ReturnDate DATE



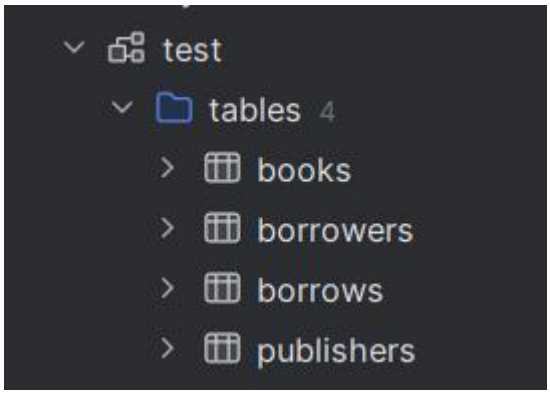
(3)（30 分）生成 DDL 的 SQL 脚本，并在数据库中创建数据库表。展示 SQL 脚本并截图展示创建成功的表结构。

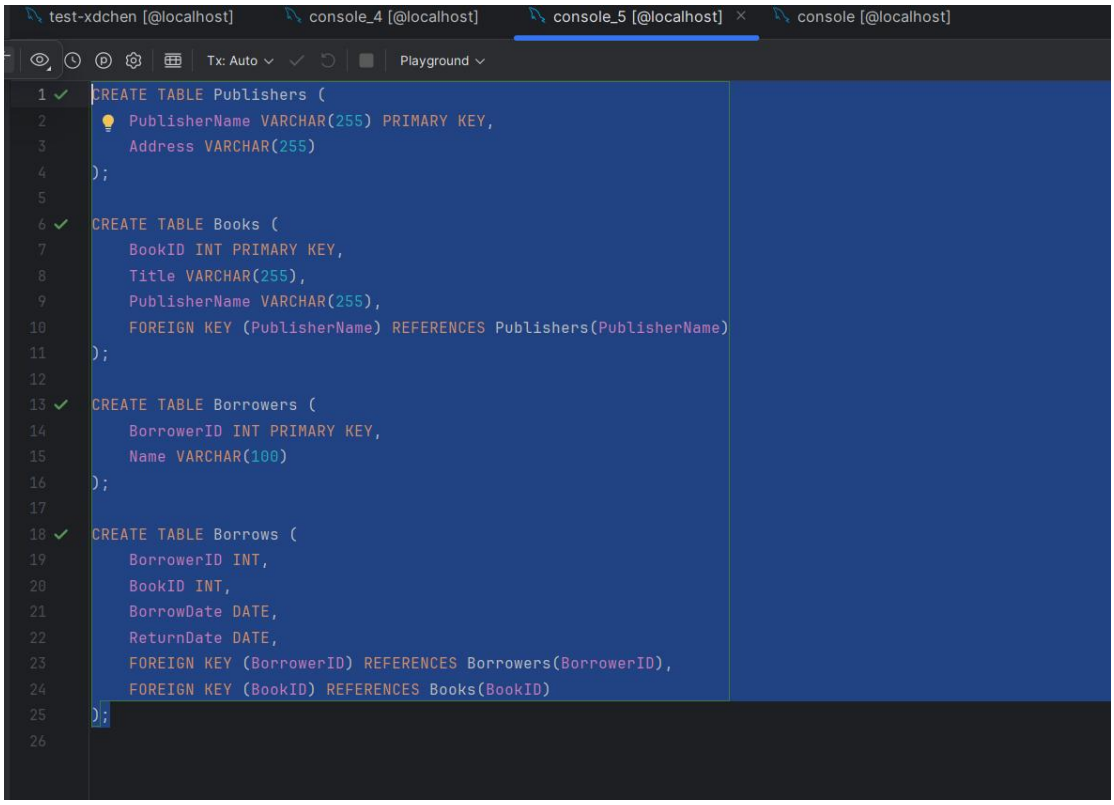
```
CREATE TABLE Publishers (
    PublisherName VARCHAR(255) PRIMARY KEY,
    Address VARCHAR(255)
);

CREATE TABLE Books (
    BookID INT PRIMARY KEY,
    Title VARCHAR(255),
    PublisherName VARCHAR(255),
    FOREIGN KEY (PublisherName) REFERENCES Publishers(PublisherName)
);

CREATE TABLE Borrowers (
    BorrowerID INT PRIMARY KEY,
    Name VARCHAR(100)
);

CREATE TABLE Borrows (
    BorrowerID INT,
    BookID INT,
    BorrowDate DATE,
    ReturnDate DATE,
    FOREIGN KEY (BorrowerID) REFERENCES Borrowers(BorrowerID),
    FOREIGN KEY (BookID) REFERENCES Books(BookID)
);
```




test-xdchen [@localhost]
console_4 [@localhost]
console_5 [@localhost]
console [@localhost]
Tx: Auto
Playground
CREATE TABLE Publishers (
PublisherName VARCHAR(255) PRIMARY KEY,
Address VARCHAR(255)
);
CREATE TABLE Books (
BookID INT PRIMARY KEY,
Title VARCHAR(255),
PublisherName VARCHAR(255),
FOREIGN KEY (PublisherName) REFERENCES Publishers(PublisherName)
);
CREATE TABLE Borrowers (
BorrowerID INT PRIMARY KEY,
Name VARCHAR(100)
);
CREATE TABLE Borrows (
BorrowerID INT,
BookID INT,
BorrowDate DATE,
ReturnDate DATE,
FOREIGN KEY (BorrowerID) REFERENCES Borrowers(BorrowerID),
FOREIGN KEY (BookID) REFERENCES Books(BookID)
);