



Báo cáo Cơ sở dữ liệu hệ thống bán sách online

Nhóm thực hiện: Nhóm 9

Thành viên:

- Bùi Hải Đăng - 23020356 (Nhóm trưởng)
- Nguyễn Duy Hoàng - 23020368

▼ I: Giới thiệu

▼ 1: Hệ thống bán sách online và ứng dụng của hệ thống trong thực tế

Hệ thống bán sách online là một hệ thống quản lý và cung cấp dịch vụ bán sách trực tuyến. Hiện nay, các hệ thống bán hàng online nói chung và bán sách online nói riêng đã trở thành xu hướng phổ biến nhờ tính tiện lợi, nhanh chóng và khả năng tiếp cận đối tượng khách hàng rộng khắp mọi nơi. Thông qua hệ thống này, chủ cửa hàng có thể dễ dàng quản lý nhân viên, cùng với những sản phẩm được bán ra cùng với đó là nhiều tiện ích giúp người tiêu dùng dễ dàng và tiếp cận với sản phẩm.

Những tiện ích của hệ thống bán sách online bao gồm:

- **Hiệu quả hoá quy trình kinh doanh:** Giúp nhà cung cấp quản lý tốt hơn các danh mục sách, đơn hàng và kho hàng.
- **Tăng tính cạnh tranh:** Hỗ trợ tiếp cận nhiều khách hàng hơn nhờ vào sự tiêu dùng internet ngày càng phổ biến.
- **Tiết kiệm chi phí:** Giảm thiểu chi phí vận hành so với các hình thức kinh doanh truyền thống.

Nhận thấy được tính thực tiễn và mức độ phổ biến của hệ thống bán sách online, nhóm chúng tôi quyết định chọn đề tài này để nghiên cứu và triển khai. Trong báo cáo này, chúng tôi sẽ chi tiết từng bước thực hiện từ khâu nghiên cứu hệ thống, thiết kế mô hình ER, đến việc triển khai cơ sở dữ liệu trong DBMS.

▼ 2: Các yêu cầu về lưu trữ

- **Nhân viên:** Thông tin về nhân viên trong hệ thống bao gồm: họ tên, ngày sinh, số điện thoại, địa chỉ, mã nhân viên, vai trò và văn phòng làm việc.
- **Hàng hóa:** Thông tin về sách bao gồm: tên sách, tác giả, thể loại, giá bán, số lượng tồn kho, và nhà xuất bản.
- **Khách hàng:** Bao gồm thông tin như: mã khách hàng, họ tên, địa chỉ, số điện thoại, số dư tài khoản, danh sách sách đã mua.
- **Đơn hàng:** Bao gồm mã đơn hàng, tên đơn hàng, ngày đặt hàng, ngày giao hàng, trạng thái đơn hàng, chi tiết đơn hàng.
- **Nhà cung cấp:** Bao gồm thông tin nhà cung cấp sách như: tên nhà cung cấp, địa chỉ, số điện thoại, danh mục hàng cung cấp.

- **Thanh toán:** Bao gồm thông tin thanh toán như: mã khách hàng, số tiền thanh toán, ngày thanh toán.

▼ II: Lược đồ E - R

▼ 1: Các thực thể trong hệ thống

- **Văn phòng (offices)**
 - Quản lý thông tin về các chi nhánh văn phòng.
 - **Thuộc tính:** Tên văn phòng, mã văn phòng, địa chỉ.
- **Nhân viên (employees)**
 - Lưu trữ thông tin nhân viên làm việc tại văn phòng.
 - **Thuộc tính:** Mã nhân viên, tên, số điện thoại, vai trò, mã văn phòng.
- **Khách hàng (customers)**
 - Lưu trữ thông tin khách hàng và giao dịch của họ.
 - **Thuộc tính:** Mã khách hàng, tên, số điện thoại, địa chỉ, số dư tài khoản, nhân viên quản lý.
- **Sách (books)**
 - Quản lý thông tin về các đầu sách.
 - **Thuộc tính:** Tên sách, giá bán, thể loại, tác giả, số lượng tồn kho.
- **Đơn hàng (orders)**
 - Lưu trữ thông tin về đơn hàng của khách hàng.
 - **Thuộc tính:** Mã đơn, mã khách hàng, giá trị đơn, ngày đặt và ngày giao hàng.
- **Chi tiết đơn hàng (orderdetails)**
 - Liên kết sách với đơn hàng.
 - **Thuộc tính:** Mã đơn hàng, mã sách, số lượng sản phẩm.
- **Thanh toán (payments)**
 - Theo dõi các giao dịch thanh toán từ khách hàng.
 - **Thuộc tính:** Mã khách hàng, số tiền thanh toán, ngày thanh toán.

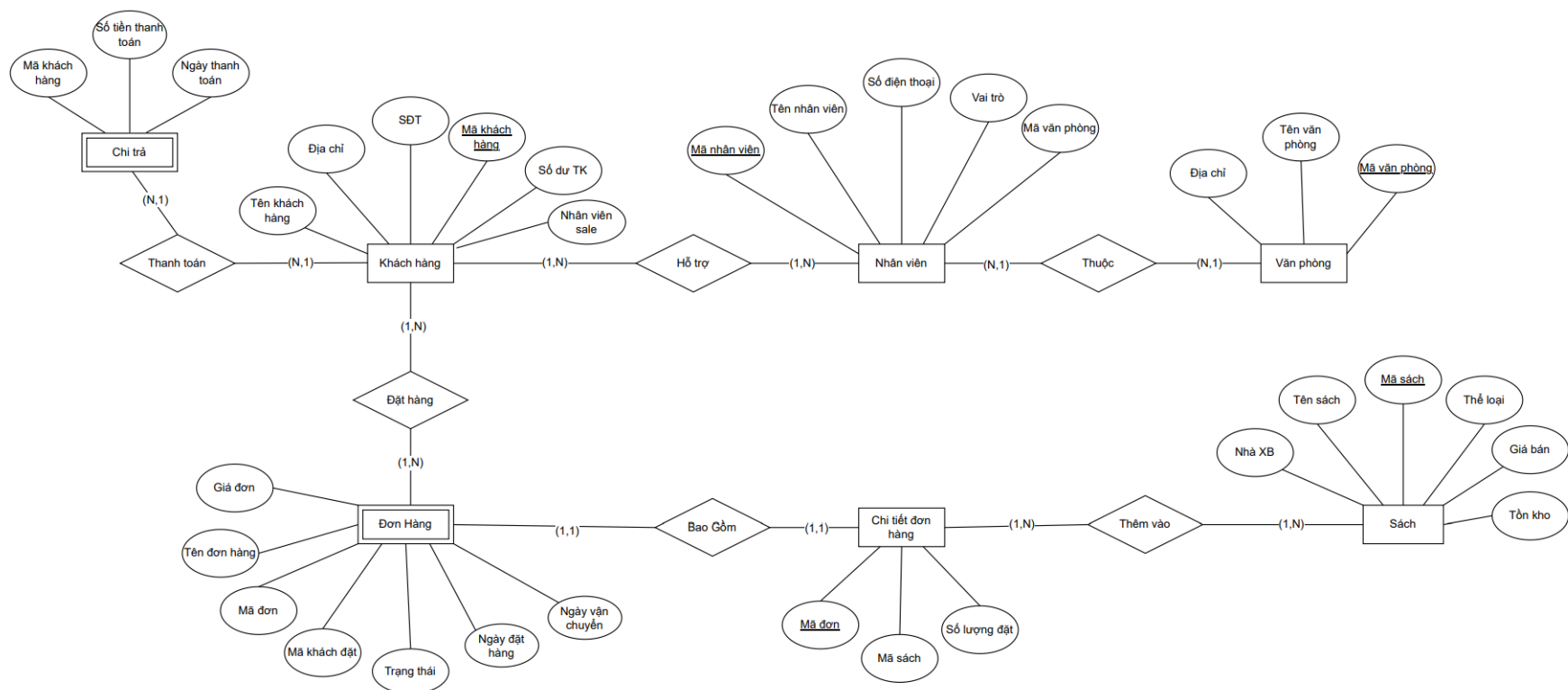
▼ 2: Các liên kết trong hệ thống

- **Thanh toán:** Liên kết giữa khách hàng và thanh toán, với mối quan hệ (N,1), nghĩa là mỗi giao dịch thanh toán liên kết với một khách hàng.
- **Hỗ trợ:** Liên kết giữa nhân viên và khách hàng, với mối quan hệ (1,N), nghĩa là một nhân viên có thể hỗ trợ nhiều khách hàng.
- **Thuộc:** Liên kết giữa nhân viên và văn phòng, với mối quan hệ (N,1), nghĩa là nhiều nhân viên thuộc một văn phòng.
- **Đặt hàng:** Liên kết giữa khách hàng và đơn hàng, với mối quan hệ (1,N), nghĩa là một khách hàng có thể đặt nhiều đơn hàng.
- **Bao gồm:** Liên kết giữa đơn hàng và chi tiết đơn hàng, với mối quan hệ (1,1), nghĩa là một đơn hàng sẽ chỉ có một chi tiết đơn hàng.
- **Thêm vào:** Liên kết giữa chi tiết đơn hàng và sách, với mối quan hệ (1,N), nghĩa là một chi tiết đơn hàng có thể tham chiếu đến nhiều cuốn sách, nhưng một cuốn sách chỉ có thể nằm trong một chi tiết đơn hàng.

▼ 3: Luồng hoạt động của hệ thống

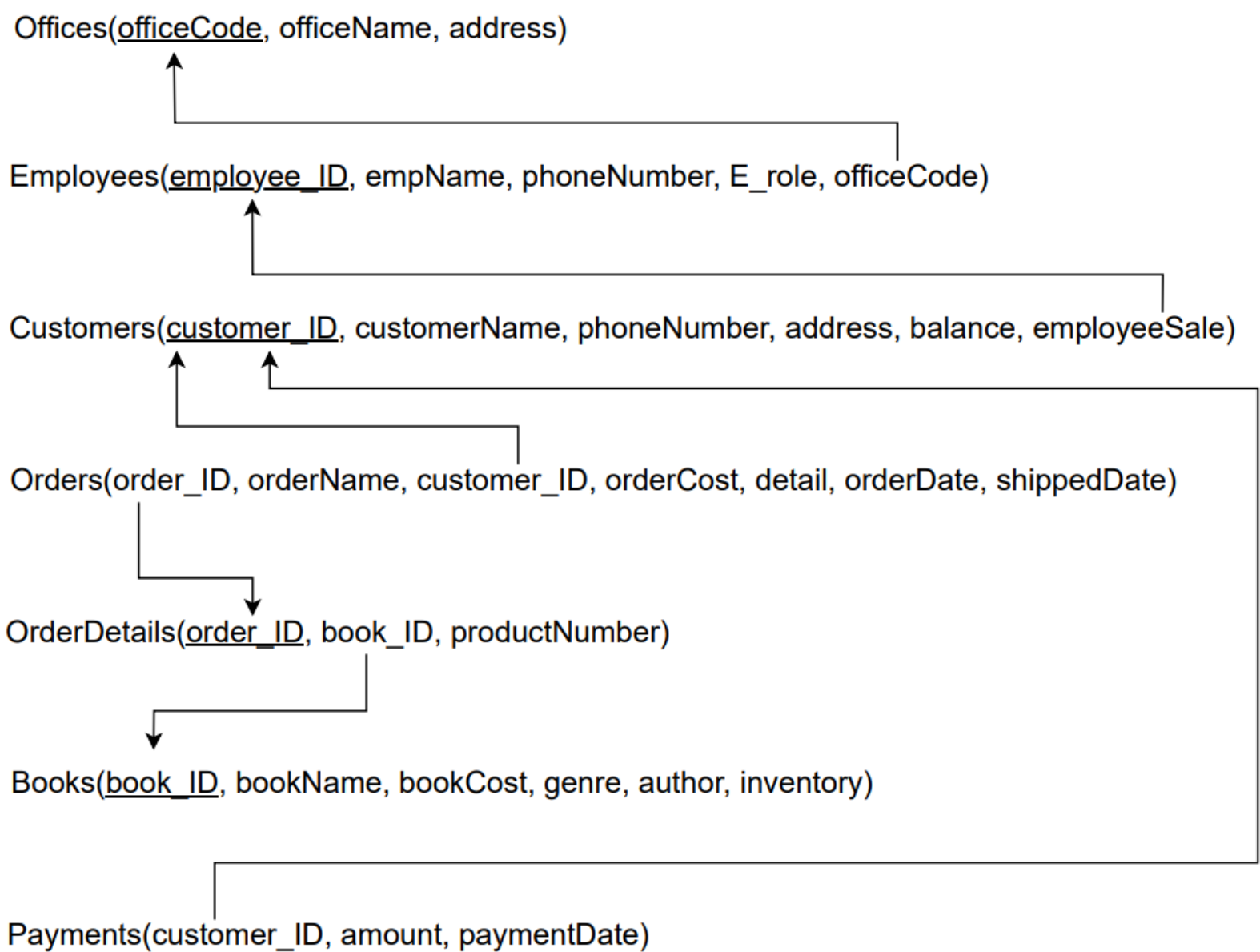
- Nhân viên quản lý các thông tin văn phòng và khách hàng.
- Khách hàng tạo đơn hàng và thanh toán trực tuyến.
- Hệ thống tự động cập nhật tồn kho và thông tin giao dịch sau mỗi đơn hàng.

▼ 4: Lược đồ E - R



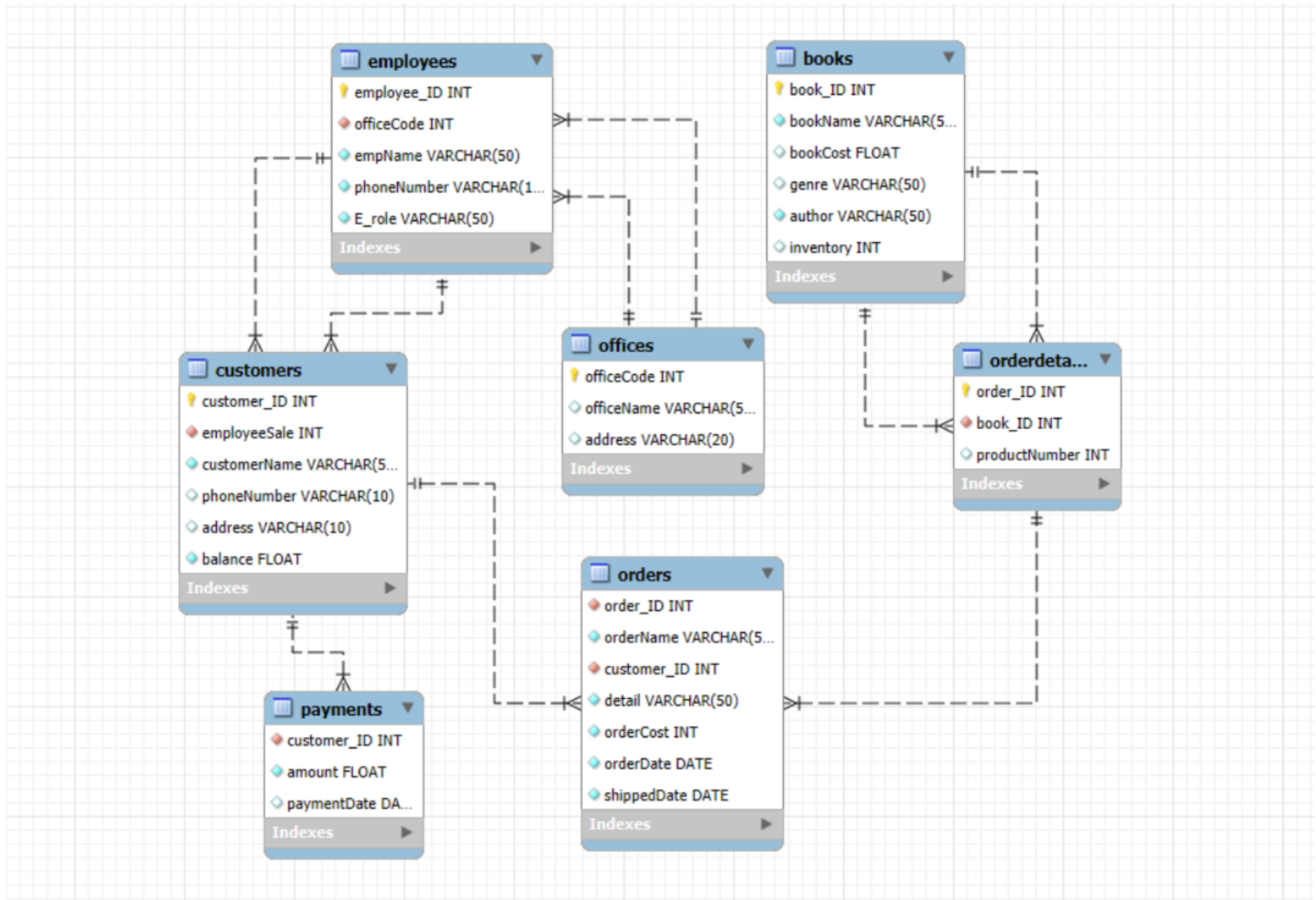
▼ III: Lược đồ quan hệ

Chuyển từ lược đồ E - R sang lược đồ quan hệ



▼ IV: Cài đặt hệ thống

▼ 1: Cài đặt DBMS



▼ Bảng 1: Bảng Offices

▼ Câu lệnh SQL:

```
-- khởi tạo bảng offices
create table Offices (
    officeCode int not null,
    officeName varchar(50),
    address varchar(20),
    primary key (officeCode)
);
INSERT INTO offices (officeCode, officeName, address) VALUES
(1, 'Head Office', '123 Main St'),
(2, 'Branch Office', '456 Elm St'),
(3, 'Central Office', '789 Oak Ave'),
(4, 'North Office', '321 Maple Rd'),
(5, 'South Office', '654 Pine Ln'),
(6, 'West Office', '987 Cedar Ct'),
(7, 'East Office', '246 Birch Blvd'),
(8, 'Downtown Office', '135 Elm St'),
(9, 'Uptown Office', '864 Spruce St'),
(10, 'Suburban Office', '753 Willow Way');
```

▼ Kết quả:

	officeCode	officeName	address
1	1	Head Office	123 Main St
2	2	Branch Office	456 Elm St
3	3	Central Office	789 Oak Ave
4	4	North Office	321 Maple Rd
5	5	South Office	654 Pine Ln
6	6	West Office	987 Cedar Ct
7	7	East Office	246 Birch Blvd
8	8	Downtown Office	135 Elm St
9	9	Uptown Office	864 Spruce St
10	10	Suburban Office	753 Willow Way

▼ **Bảng 2: Bảng Employees**

▼ **Câu lệnh SQL:**

```
-- khởi tạo bảng Employees
create table Employees (
    employee_ID int not null,
    officeCode int not null,
    empName varchar(50) not null,
    phoneNumber varchar(10) not null,
    E_role varchar(50) not null,
    primary key(employee_ID),
    foreign key( officeCode) references Offices(officeCode)
);
-- Thêm dữ liệu vào bảng Employees
INSERT INTO Employees (employee_ID, officeCode, empName, phoneNumber, E_role) VALUES
(101, 1, 'Nguyen Van A', '0901234567', 'Manager'),
(102, 1, 'Tran Thi B', '0909876543', 'Sales'),
(103, 2, 'Le Van C', '0912345678', 'Staff'),
(104, 3, 'Pham Hong D', '0923456789', 'Sales'),
(105, 4, 'Hoang Minh E', '0934567890', 'Manager'),
(106, 5, 'Nguyen Thi F', '0945678901', 'Staff'),
(107, 6, 'Tran Van G', '0956789012', 'Sales'),
(108, 7, 'Le Thi H', '0967890123', 'Staff'),
(109, 8, 'Nguyen Van I', '0978901234', 'Manager'),
(110, 9, 'Pham Thi J', '0989012345', 'Sales');
```

▼ **Kết quả:**

	employee_ID	officeCode	empName	phoneNumber	E_role
1	101	1	Nguyen Van A	0901234567	Manager
2	102	1	Tran Thi B	0909876543	Sales
3	103	2	Le Van C	0912345678	Staff
4	104	3	Pham Hong D	0923456789	Sales
5	105	4	Hoang Minh E	0934567890	Manager
6	106	5	Nguyen Thi F	0945678901	Staff
7	107	6	Tran Van G	0956789012	Sales
8	108	7	Le Thi H	0967890123	Staff
9	109	8	Nguyen Van I	0978901234	Manager
10	110	9	Pham Thi J	0989012345	Sales

▼ **Bảng 3: Bảng Customers**

▼ **Câu lệnh SQL:**

```
-- khởi tạo bảng Customers
create table Customers(
```



```

customer_ID int not null,
employeeSale int not null,
customerName varchar(50) not null,
phoneNumber varchar(10),
address varchar(10),
balance float not null,
primary key(customer_ID),
foreign key(employeeSale) references Employees(employee_ID)
);
-- Thêm dữ liệu vào bảng Customers
INSERT INTO Customers (customer_ID, employeeSale, customerName, phoneNumber, address,
(201, 102, 'Pham Hong D', '0933333333', '1A', 100.0),
(202, 103, 'Hoang Minh E', '0944444444', '2B', 200.0),
(203, 104, 'Nguyen Thi F', '0955555555', '3C', 150.0),
(204, 105, 'Le Van K', '0966666666', '4D', 80.0),
(205, 106, 'Tran Thi L', '0977777777', '5E', 120.0),
(206, 107, 'Hoang Van M', '0988888888', '6F', 300.0),
(207, 108, 'Nguyen Thi N', '0999999999', '7G', 50.0),
(208, 109, 'Pham Van O', '0911111111', '8H', 40.0),
(209, 110, 'Le Thi P', '0922222222', '9I', 500.0),
(210, 101, 'Tran Van Q', '0933333334', '10J', 70.0);

```

▼ Kết quả:

	customer_ID	employeeSale	customerName	phoneNumber	address	balance
1	201	102	Pham Hong D	0933333333	1A	0
2	202	103	Hoang Minh E	0944444444	2B	200
3	203	104	Nguyen Thi F	0955555555	3C	150
4	204	105	Le Van K	0966666666	4D	80
5	205	106	Tran Thi L	0977777777	5E	120
6	206	107	Hoang Van M	0988888888	6F	300
7	207	108	Nguyen Thi N	0999999999	7G	50
8	208	109	Pham Van O	0911111111	8H	40
9	209	110	Le Thi P	0922222222	9I	500
10	210	101	Tran Van Q	0933333334	10J	70

▼ Bảng 4: Bảng Orders

▼ Câu lệnh SQL:

```

-- khởi tạo bảng orders
create table Orders (
order_ID int not null primary key,
orderName varchar(50) not null,
customer_ID int not null,
detail varchar(50) not null,
orderCost int not null,
orderDate date not null,
shippedDate date not null,
foreign key(order_ID) references orderdetails(order_ID),
foreign key(customer_ID) references Customers(customer_ID)
);
-- Thêm dữ liệu vào bảng Orders
INSERT INTO Orders (order_ID, orderName, customer_ID, detail, orderCost, orderDate, s
(401, 'Order 1', 201, '2 copies of Book A', 100, '2023-12-01', '2023-12-03'),
(402, 'Order 2', 202, '1 copy of Book B', 60, '2023-12-02', '2023-12-04'),
(403, 'Order 3', 203, '3 copies of Book C', 210, '2023-12-03', '2023-12-05'),
(404, 'Order 4', 204, '5 copies of Book D', 200, '2023-12-04', '2023-12-06'),
(405, 'Order 5', 205, '4 copies of Book E', 120, '2023-12-05', '2023-12-07'),
(406, 'Order 6', 206, '2 copies of Book F', 180, '2023-12-06', '2023-12-08'),
(407, 'Order 7', 207, '3 copies of Book G', 300, '2023-12-07', '2023-12-09'),

```

```
(408, 'Order 8', 208, '1 copy of Book H', 45, '2023-12-08', '2023-12-10'),
(409, 'Order 9', 209, '2 copies of Book I', 160, '2023-12-09', '2023-12-11'),
(410, 'Order 10', 210, '4 copies of Book J', 220, '2023-12-10', '2023-12-12');
```

▼ Kết quả:

order_ID	orderName	customer_ID	detail	orderCost	orderDate	shippedDate
401	Order 1	201	2 copies of Book A	100	2023-12-01	2023-12-03
402	Order 2	202	1 copy of Book B	60	2023-12-02	2023-12-04
403	Order 3	203	3 copies of Book C	210	2023-12-03	2023-12-05
404	Order 4	204	5 copies of Book D	200	2023-12-04	2023-12-06
405	Order 5	205	4 copies of Book E	120	2023-12-05	2023-12-07
406	Order 6	206	2 copies of Book F	180	2023-12-06	2023-12-08
407	Order 7	207	3 copies of Book G	300	2023-12-07	2023-12-09
408	Order 8	208	1 copy of Book H	45	2023-12-08	2023-12-10
409	Order 9	209	2 copies of Book I	160	2023-12-09	2023-12-11
410	Order 10	210	4 copies of Book J	220	2023-12-10	2023-12-12

▼ **Bảng 5: Bảng OrderDetails**

▼ Câu lệnh SQL:

```
-- khởi tạo bảng orderdetails
create table Orderdetails (
    order_ID int not null ,
    book_ID int not null,
    productNumber int,
    primary key(Order_ID),
    foreign key (book_ID) references Books(book_ID)
);
-- Thêm dữ liệu vào bảng orderdetails
INSERT INTO Orderdetails (order_ID, book_ID, productNumber) VALUES
(401, 301, 2),
(402, 302, 1),
(403, 303, 3),
(404, 304, 5),
(405, 305, 4),
(406, 306, 2),
(407, 307, 3),
(408, 308, 1),
(409, 309, 2),
(410, 310, 4);
```

▼ Kết quả:

order_ID	book_ID	productNumber
401	301	2
402	302	1
403	303	3
404	304	5
405	305	4
406	306	2
407	307	3
408	308	1
409	309	2
410	310	4

▼ **Bảng 6: Bảng Books**

▼ Câu lệnh SQL:

```
-- khởi tạo bảng books
create table Books (
    book_ID int not null primary key,
    bookName varchar(50) not null,
    bookCost float,
    genre varchar(50),
    author varchar(50) not null,
    inventory int
);
-- Thêm dữ liệu vào bảng books
INSERT INTO Books (book_ID, bookName, bookCost, genre, author, inventory) VALUES
(301, 'Book A', 50.0, 'Fiction', 'Author 1', 20),
(302, 'Book B', 60.0, 'Science', 'Author 2', 15),
(303, 'Book C', 70.0, 'History', 'Author 3', 10),
(304, 'Book D', 40.0, 'Fantasy', 'Author 4', 25),
(305, 'Book E', 30.0, 'Fiction', 'Author 1', 50),
(306, 'Book F', 90.0, 'Science', 'Author 5', 8),
(307, 'Book G', 100.0, 'Biography', 'Author 6', 12),
(308, 'Book H', 45.0, 'Mystery', 'Author 7', 18),
(309, 'Book I', 80.0, 'Adventure', 'Author 8', 14),
(310, 'Book J', 55.0, 'Horror', 'Author 9', 30);
```

▼ Kết quả:

book_ID	bookName	bookCost	genre	author	inventory
301	Book A	50	Fiction	Author 1	20
302	Book B	60	Science	Author 2	15
303	Book C	70	History	Author 3	10
304	Book D	40	Fantasy	Author 4	25
305	Book E	30	Fiction	Author 1	50
306	Book F	90	Science	Author 5	8
307	Book G	100	Biography	Author 6	12
308	Book H	45	Mystery	Author 7	18
309	Book I	80	Adventure	Author 8	14
310	Book J	55	Horror	Author 9	30

▼ Bảng 7: Bảng Payments

▼ Câu lệnh SQL:

```
-- khởi tạo bảng payments
create table Payments(
    customer_ID int not null,
    amount float not null,
    paymentDate date,
    foreign key(customer_ID) references Customers(customer_ID)
);
-- Thêm dữ liệu vào bảng Payments
INSERT INTO Payments (customer_ID, amount, paymentDate) VALUES
(201, 100.0, '2023-12-05'),
(202, 60.0, '2023-12-06'),
(203, 210.0, '2023-12-07'),
(204, 200.0, '2023-12-08'),
(205, 120.0, '2023-12-09'),
(206, 180.0, '2023-12-10'),
(207, 300.0, '2023-12-11'),
(208, 45.0, '2023-12-12'),
```



```
(209, 160.0, '2023-12-13'),  
(210, 220.0, '2023-12-14');
```

▼ Kết quả:

customer_ID	amount	paymentDate
201	100	2023-12-05
202	60	2023-12-06
203	210	2023-12-07
204	200	2023-12-08
205	120	2023-12-09
206	180	2023-12-10
207	300	2023-12-11
208	45	2023-12-12
209	160	2023-12-13
210	220	2023-12-14

▼ 2: Các yêu cầu của Assignment

▼ 2.1: Create at least 3 constraints by using « alter table » statements

```
1  -- Adding constraints to ensure data integrity  
2  ✓ ALTER TABLE Orderdetails  
3      ADD CONSTRAINT fk_book FOREIGN KEY (book_ID) REFERENCES Books(book_ID);  
4  
5  ✓ ALTER TABLE Employees  
6      ⚡ ADD CONSTRAINT fk_office FOREIGN KEY (officeCode) REFERENCES offices(officeCode);  
7  
8  ✓ ALTER TABLE Customers  
9      ADD CONSTRAINT fk_employee FOREIGN KEY (employeeSale) REFERENCES Employees(employee_ID);  
10
```

▼ 2.2: Insert data in your database

▼ Câu lệnh SQL:

```
INSERT INTO offices (officeCode, officeName, address) VALUES  
(11, 'Xuan Thuy Office', '123 Xuan Thuy'),  
(12, 'Hoan Kiem Office', '60 Hang Trong'),  
(13, 'Ha Dong Office', '165 Luong The Vinh'),  
(14, 'Cau Dien Office', '20 Cau Dien'),  
(15, 'Tay Ho Office', '28 Xuan La');
```

```
INSERT INTO employees (employee_ID, officeCode, empName, phoneNumber, E_role) VALUES  
(111, 2, 'Nguyen Duy Hoang', '0353558161', 'Manager'),  
(112, 4, 'Nguyen Van Duy', '0901231122', 'Sales'),  
(113, 7, 'Phan Hoang Dung', '0901233344', 'Staff'),  
(114, 8, 'Ngo Quang Dung', '0901238866', 'Staff'),  
(115, 9, 'Bui Thanh Dan', '0909873434', 'Sales');
```

```
INSERT INTO customers (customer_ID, employeeSale, customerName, phoneNumber, address,  
(211, 102, 'Nguyen Dang Duong', '0933331234', '65A', 123.0),  
(212, 111, 'Hoang Tien Dat', '0933331122', '12A', 189.0),  
(213, 112, 'Ton Thanh Dat', '0933335566', '19C', 213.0),  
(214, 101, 'Phan Quang Truong', '0933331265', '21B', 3423.0),  
(215, 109, 'Hoang Minh Quang', '0944442308', '29B', 2010.0);
```

```
INSERT INTO book (book_ID, bookName, bookCost, genre, author, inventory) VALUE
(311, 'Harry Poster',200, 'Long Story', 'Author 11', 8),
(312, 'Alien',99, 'Horror', 'Author 18', 20),
(313, 'Predator',119, 'Horror', 'Author 12', 5),
(314, 'Jujutsu Kaisen',20, 'Comic', 'Author 21', 2),
(315, 'No love no life',30, 'Drama', 'Author 16', 18);

INSERT INTO orderdetails (order_ID, book_ID, productNumber) VALUE
(411,312,1),
(412,313,2),
(413,311,5),
(414,314,3),
(415,315,2);

INSERT INTO orders (order_ID, orderName, customer_ID, detail, orderCost, orderDate, s
(411, 'Order 11', 211, '1 copy Alien', 99, '2024-09-08', '2024-09-12'),
(412, 'Order 12', 212, '2 copy Predator', 238, '2024-10-18', '2024-10-22'),
(413, 'Order 13', 213, '5 copies Harry Poster',1000 , '2024-10-08', '2024-10-12'),
(414, 'Order 14', 214, '3 copy Jujutsu Kaisen', 60, '2024-11-28', '2024-12-02'),
(415, 'Order 15', 215, '2 copies No love no life', 60, '2024-11-30', '2024-12-06');

INSERT INTO payments (customer_ID, amount, paymentDate) VALUE
(211,99, '2024-09-12'),
(212,240, '2024-10-22'),
(213,1000, '2024-12-19'),
(214,60, '2024-12-02'),
(215,65, '2024-12-8');
```

▼ Kết quả:

▼ Thêm dữ liệu thành công bảng Offices

officeCode	officeName	address
1	Head Office	123 Main St
2	Branch Office	456 Elm St
3	Central Office	789 Oak Ave
4	North Office	321 Maple Rd
5	South Office	654 Pine Ln
6	West Office	987 Cedar Ct
7	East Office	246 Birch Blvd
8	Downtown Office	135 Elm St
9	Uptown Office	864 Spruce St
10	Suburban Office	753 Willow Way
11	Xuan Thuy Office	123 Xuan Thuy
12	Hoan Kiem Office	60 Hang Trong
13	Ha Dong Office	165 Luong The Vinh
14	Cau Dien Office	20 Cau Dien
15	Tay Ho Office	28 Xuan La

▼ Thêm dữ liệu thành công bảng Employees

employee_ID	officeCode	empName	phoneNumber	E_role
101	1	Nguyen Van A	0901234567	Manager
102	1	Tran Thi B	0909876543	Sales
103	2	Le Van C	0912345678	Staff
104	3	Pham Hong D	0923456789	Sales
105	4	Hoang Minh E	0934567890	Manager
106	5	Nguyen Thi F	0945678901	Staff
107	6	Tran Van G	0956789012	Sales
108	7	Le Thi H	0967890123	Staff
109	8	Nguyen Van I	0978901234	Manager
110	9	Pham Thi J	0989012345	Sales
111	2	Nguyen Duy Hoang	0353558161	Manager
112	4	Nguyen Van Duy	0901231122	Sales
113	7	Phan Hoang Dung	0901233344	Staff
114	8	Ngo Quang Dung	0901238866	Staff
115	9	Bui Thanh Dan	0909873434	Sales

▼ Thêm dữ liệu thành công bảng Customers

customer_ID	employeeSale	customerName	phoneNumber	address	balance
201	102	Pham Hong D	0933333333	1A	0
202	103	Hoang Minh E	0944444444	2B	200
203	104	Nguyen Thi F	0955555555	3C	150
204	105	Le Van K	0966666666	4D	80
205	106	Tran Thi L	0977777777	5E	120
206	107	Hoang Van M	0988888888	6F	300
207	108	Nguyen Thi N	0999999999	7G	50
208	109	Pham Van O	0911111111	8H	40
209	110	Le Thi P	0922222222	9I	500
210	101	Tran Van Q	0933333334	10J	70
211	102	Nguyen Dang Duong	0933331234	65A	123
212	111	Hoang Tien Dat	0933331122	12A	189
213	112	Ton Thanh Dat	0933335566	19C	213
214	101	Phan Quang Truong	0933331265	21B	3423
215	109	Hoang Minh Quang	0944442308	29B	2010

▼ Thêm dữ liệu thành công bảng Orders

order_ID	orderName	customer_ID	detail	orderCost	orderDate	shippedDate
401	Order 1	201	2 copies of Book A	100	2023-12-01	2023-12-03
402	Order 2	202	1 copy of Book B	60	2023-12-02	2023-12-04
403	Order 3	203	3 copies of Book C	210	2023-12-03	2023-12-05
404	Order 4	204	5 copies of Book D	200	2023-12-04	2023-12-06
405	Order 5	205	4 copies of Book E	120	2023-12-05	2023-12-07
406	Order 6	206	2 copies of Book F	180	2023-12-06	2023-12-08
407	Order 7	207	3 copies of Book G	300	2023-12-07	2023-12-09
408	Order 8	208	1 copy of Book H	45	2023-12-08	2023-12-10
409	Order 9	209	2 copies of Book I	160	2023-12-09	2023-12-11
410	Order 10	210	4 copies of Book J	220	2023-12-10	2023-12-12
411	Order 11	211	1 copy Alien	99	2024-09-08	2024-09-12
412	Order 12	212	2 copy Predator	238	2024-10-18	2024-10-22
413	Order 13	213	5 copies Harry Poster	1000	2024-10-08	2024-10-12
414	Order 14	214	3 copy Jujutsu Kaisen	60	2024-11-28	2024-12-02
415	Order 15	215	2 copies No love no Life	60	2024-11-30	2024-12-06

▼ Thêm dữ liệu thành công bảng Orderdetails

order_ID	book_ID	productNumber
401	301	2
402	302	1
403	303	3
404	304	5
405	305	4
406	306	2
407	307	3
408	308	1
409	309	2
410	310	4
411	312	1
412	313	2
413	311	5
414	314	3
415	315	2

▼ Thêm dữ liệu thành công bảng Books

book_ID	bookName	bookCost	genre	author	inventory
301	Book A	50	Fiction	Author 1	20
302	Book B	60	Science	Author 2	15
303	Book C	70	History	Author 3	10
304	Book D	40	Fantasy	Author 4	25
305	Book E	30	Fiction	Author 1	50
306	Book F	90	Science	Author 5	8
307	Book G	100	Biography	Author 6	12
308	Book H	45	Mystery	Author 7	18
309	Book I	80	Adventure	Author 8	14
310	Book J	55	Horror	Author 9	30
311	Harry Poster	200	Long Story	Author 11	8
312	Alien	99	Horror	Author 18	20
313	Predator	119	Horror	Author 12	5
314	Jujutsu Kaisen	20	Comic	Author 21	2
315	No love no life	30	Drama	Author 16	18

▼ Thêm dữ liệu thành công bảng Payments

customer_ID	amount	paymentDate
201	100	2023-12-05
202	60	2023-12-06
203	210	2023-12-07
204	200	2023-12-08
205	120	2023-12-09
206	180	2023-12-10
207	300	2023-12-11
208	45	2023-12-12
209	160	2023-12-13
210	220	2023-12-14
211	99	2024-09-12
212	240	2024-10-22
213	1000	2024-12-19
214	60	2024-12-02
215	65	2024-12-08
201	100	2023-12-15

▼ 2.3: Write some query for each of the following requirements

▼ **Câu lệnh SQL:**

```
-- Query using inner join
SELECT e.empName, o.officeName
FROM Employees e
      INNER JOIN offices o ON e.officeCode = o.officeCode;

-- Query using outer join
SELECT c.customerName, e.empName
FROM Customers c
      LEFT OUTER JOIN employees e ON c.employeeSale = e.employee_ID;

-- Using subquery in WHERE clause
SELECT * FROM Books
WHERE bookCost > (SELECT AVG(bookCost) FROM Books);

-- Using subquery in FROM clause
SELECT *
FROM employees
WHERE E_role = 'Staff';

-- Query using GROUP BY and aggregate functions
SELECT genre, COUNT(*) AS book_count, AVG(bookCost) AS avg_cost
FROM Books
GROUP BY genre;
```

▼ **Kết quả:**

▼ Query using inner join

empName	officeName
Nguyen Van A	Head Office
Tran Thi B	Head Office
Le Van C	Branch Office
Pham Hong D	Central Office
Hoang Minh E	North Office
Nguyen Thi F	South Office
Tran Van G	West Office
Le Thi H	East Office
Nguyen Van I	Downtown Office
Pham Thi J	Uptown Office
Nguyen Duy Hoang	Branch Office
Nguyen Van Duy	North Office
Phan Hoang Dung	East Office
Ngo Quang Dung	Downtown Office
Bui Thanh Dan	Uptown Office

▼ Query using outer join

genre	book_count	avg_cost
Fiction	2	40
Science	2	75
History	1	70
Fantasy	1	40
Biography	1	100
Mystery	1	45
Adventure	1	80
Horror	3	91
Long Story	1	200
Comic	1	20
Drama	1	30

▼ Using subquery in where

book_ID	bookName	bookCost	genre	author	inventory
306	Book F	90	Science	Author 5	8
307	Book G	100	Biography	Author 6	12
309	Book I	80	Adventure	Author 8	14
311	Harry Poster	200	Long Story	Author 11	8
312	Alien	99	Horror	Author 18	20
313	Predator	119	Horror	Author 12	5

▼ Using subquery in from

employee_ID	officeCode	empName	phoneNumber	E_role
103	2	Le Van C	0912345678	Staff
106	5	Nguyen Thi F	0945678901	Staff
108	7	Le Thi H	0967890123	Staff
113	7	Phan Hoang Dung	0901233344	Staff
114	8	Ngo Quang Dung	0901238866	Staff

▼ Query using group by and aggregate functions

genre	book_count	avg_cost
Fiction	2	40
Science	2	75
History	1	70
Fantasy	1	40
Biography	1	100
Mystery	1	45
Adventure	1	80
Horror	3	91
Long Story	1	200
Comic	1	20
Drama	1	30

▼ 2.4: Write at least one transaction using rollback

▼ Câu lệnh SQL:

Đã chạy thành công.

```
1  -- 1 : Cập nhật Balance với rollback
2  ✓ UPDATE Customers
3    SET balance = balance - 100
4    WHERE customer_ID = 201;
5
6  ✓ INSERT INTO Payments (customer_ID, amount, paymentDate)
7    VALUES ( customer_ID 201, amount 100.0, paymentDate '2023-12-15');
8
9  ✓ ROLLBACK;
10
11 -- 2: Thêm dữ liệu order với rollback
12 ✓ START TRANSACTION;
13 ✓ INSERT INTO Orderdetails (order_ID, book_ID, productNumber) VALUE
14   ( order_ID 601, book_ID 312, productNumber 3);
15 ✓ INSERT INTO orders (order_ID, orderName, customer_ID, detail, orderCost, orderDate, shippedDate)
16   VALUES ( order_ID 601, orderName 'Test', customer_ID 202, detail 'Detail', orderCost 150, orderDate '2024-01-01', shippedDate '2024-01-03');
17
18 ✓ ROLLBACK;
19
20 -- 3: cập nhật inventory với rollback
21 ✓ START TRANSACTION;
22 ✓ UPDATE Books
23   SET inventory = inventory - 2
24   WHERE book_ID = 301;
25
26 ✓ ROLLBACK;
```

▼ 2.5: Write at least one trigger

▼ Câu lệnh SQL:

Đã chạy thành công.

```
1  ✓ -- 1 : cập nhật hàng tồn kho mới
2    CREATE TRIGGER update_inventory
3      AFTER INSERT ON Orderdetails
4      FOR EACH ROW
5      BEGIN
6        UPDATE Books
7        SET inventory = inventory - NEW.productNumber
8        WHERE book_ID = NEW.book_ID;
9      END;
10
11 ✓ -- 2: Cập nhật tồn kho sau khi chèn đơn hàng
12 CREATE TRIGGER update_inventory_after_order
13   AFTER INSERT ON Orderdetails
14   FOR EACH ROW
15   BEGIN
16     UPDATE Books
17     SET inventory = inventory - NEW.productNumber
18     WHERE book_ID = NEW.book_ID;
19   END;
20
21 ✓ -- 3: Không cho số dư là số âm
22 CREATE TRIGGER prevent_negative_balance
23   BEFORE UPDATE ON Customers
24   FOR EACH ROW
25   BEGIN
26     IF NEW.balance < 0 THEN
27       SIGNAL SQLSTATE '45000'
28       SET MESSAGE_TEXT = 'Balance cannot be negative';
29     END IF;
30   END;
```

▼ 2.6: Write at least one procedure

▼ Câu lệnh SQL:

Đã chạy thành công.

```

1 ✓ -- 1: Truy xuất đơn hàng của khách hàng
2 CREATE PROCEDURE GetCustomerOrders(IN customerID INT)
3 BEGIN
4     SELECT o.orderName, o.orderCost, o.orderDate
5     FROM Orders o
6     WHERE o.customer_ID = customerID;
7 END;
8
9 ✓ -- 2: Tính tổng doanh số theo nhân viên
10 CREATE PROCEDURE GetEmployeeSales(IN employeeID INT)
11 BEGIN
12     SELECT SUM(o.orderCost) AS total_sales
13     FROM Orders o
14     JOIN Customers c 1..n<->1: ON o.customer_ID = c.customer_ID
15     WHERE c.employeeSale = employeeID;
16 END;
17
18 ✓ -- 3: Truy xuất sách theo thể loại
19 CREATE PROCEDURE GetBooksByGenre(IN genreName VARCHAR(50))
20 BEGIN
21     SELECT bookName, bookCost, inventory
22     FROM Books
23     WHERE genre = genreName;
24 END;

```

▼ V: Tổng kết

▼ 1: Báo cáo về mô hình Cơ sở dữ liệu

- Xây dựng sơ đồ ER mô tả các thực thể và mối quan hệ quan trọng trong hệ thống.
- Chuẩn hóa cơ sở dữ liệu đến dạng 3NF để đảm bảo tính toàn vẹn dữ liệu và giảm thiểu dư thừa.
- Triển khai và quản lý cơ sở dữ liệu trên hệ quản trị cơ sở dữ liệu (DBMS), tối ưu hóa việc truy vấn và cập nhật thông tin.

▼ 2: Những tiện ích mà hệ thống mang lại

- Dễ dàng lưu trữ thông tin và tìm kiếm sách thông qua tên sách, loại sách, tác giả,
- Quản lý và cập nhật đơn hàng dễ dàng: cho phép chỉnh sửa thông tin khi có thay đổi, ví dụ như cập nhật giá sách, thay đổi số lượng tồn kho,...
- Dễ dàng quản lý khách hàng, đơn mua, thông qua hệ thống thanh toán tiện ích.

▼ 3: Những thiếu sót cần chỉnh sửa trong tương lai

- **Khả năng mở rộng:** Cần thiết kế thêm các module để hỗ trợ phân tích dữ liệu lớn hoặc tích hợp với các hệ thống bên ngoài như hệ thống vận chuyển, hệ thống quản lý nguồn sách.
- **Bảo mật dữ liệu:** Cần cải thiện cơ chế phân quyền truy cập và bảo vệ dữ liệu nhạy cảm, đảm bảo hệ thống có thể hoạt động an toàn trong môi trường thực tế.
- **Hiệu suất truy vấn:** Một số truy vấn phức tạp có thể cần tối ưu hóa thêm bằng cách áp dụng chỉ mục hoặc các kỹ thuật tối ưu hóa khác.

▼ 4: Lời kết

Hệ thống bán sách của chúng em có thể còn nhiều thiếu sót rất mong được thầy cô cùng anh chị góp ý để phát triển! Chúng em xin gửi lời cảm ơn đến thầy vì đã giành thời gian đọc bài!