bài 1

| Tên lớp | Vai trò / Mô tả |
| --- | --- |
| Sach (Book) | Đại diện cho thông tin của mỗi cuốn sách trong thư viện. Mỗi sách có các thuộc tính như mã sách, tên, tác giả, năm xuất bản, tình trạng (còn hay đã mượn). |
| DocGia (Reader) | Đại diện cho người đọc hoặc khách hàng của thư viện. Lưu trữ thông tin cá nhân, danh sách sách đã mượn, và lịch sử mượn trả. |
| NhanVien (Librarian) | Đại diện cho nhân viên thư viện – người quản lý, xử lý các nghiệp vụ như thêm sách, xóa sách, quản lý mượn trả. |
| PhieuMuon (BorrowTransaction) | Thể hiện mối quan hệ giữa độc giả và sách khi mượn. Ghi lại thông tin mượn như mã phiếu, ngày mượn, hạn trả, trạng thái (đang mượn, đã trả). |
| PhieuTra (ReturnTransaction) *(tùy chọn – có thể gộp chung với PhieuMuon)* | Ghi nhận thông tin khi độc giả trả sách, như ngày trả và tình trạng sách. |
| ThuVien (Library) | Quản lý toàn bộ hệ thống – chứa danh sách sách, danh sách độc giả, danh sách nhân viên, và các chức năng mượn/trả. |

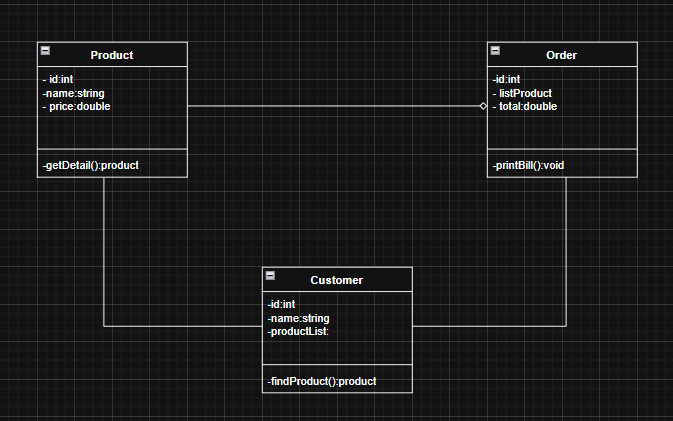
bài 2

| Mối quan hệ | Loại quan hệ | Giải thích |
| --- | --- | --- |
| Một giáo viên giảng dạy nhiều lớp học | Association (Kết hợp) | Là mối quan hệ *liên kết lỏng lẻo* giữa hai lớp. Giáo viên và lớp học có thể tồn tại độc lập — giáo viên nghỉ dạy thì lớp học vẫn có thể có giáo viên khác dạy. |
| Một đơn hàng gồm nhiều sản phẩm | Aggregation (Tập hợp) | Là mối quan hệ “*có – chứa*” nhưng *không phụ thuộc hoàn toàn về vòng đời*. Đơn hàng chứa sản phẩm, nhưng sản phẩm vẫn có thể tồn tại độc lập hoặc thuộc đơn hàng khác. |
| Một cơ thể gồm nhiều bộ phận không thể tách rời | Composition (Thành phần) | Là mối quan hệ “*chứa chặt chẽ*” — khi cơ thể không còn thì các bộ phận cũng không thể tồn tại riêng. Các phần phụ thuộc hoàn toàn vào toàn thể. |

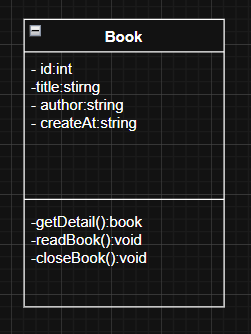
bài 3

| Thành phần | Modifier (Phạm vi truy cập) | Giải thích |
| --- | --- | --- |
| username | private | Thông tin cá nhân của người dùng, chỉ nên được truy cập bên trong class để đảm bảo an toàn dữ liệu. Các class khác muốn xem/chỉnh phải thông qua gette r/setter. |
| password | private | Dữ liệu nhạy cảm, cần được bảo vệ tuyệt đối. Không cho phép truy cập trực tiếp từ bên ngoài class. Chỉ được thao tác gián tiếp thông qua phương thức login() hoặc resetPassword(). |
| lastLoginTime | protected | Có thể được truy cập bởi class User và các lớp kế thừa (ví dụ: AdminUser, CustomerUser). Không nên để public vì đây là thông tin nội bộ hệ thống. |
| login() | public | Là chức năng chính mà bên ngoài (ví dụ: giao diện người dùng hoặc hệ thống khác) cần gọi để đăng nhập. Nên để công khai. |
| resetPassword() | public | Cho phép người dùng hoặc hệ thống quản trị gọi để đổi mật khẩu khi cần. Phải công khai để thực hiện chức năng này từ bên ngoài. |

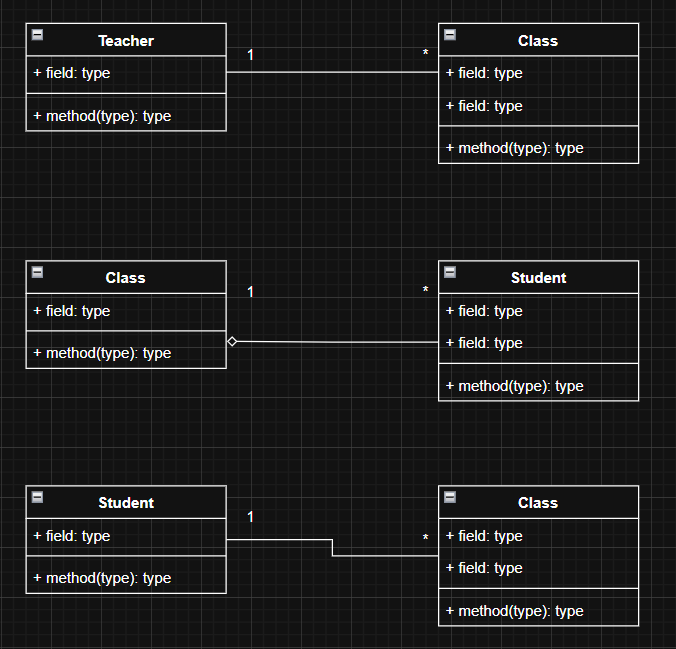
bài 4



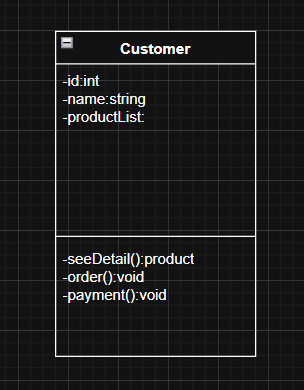
bài 5



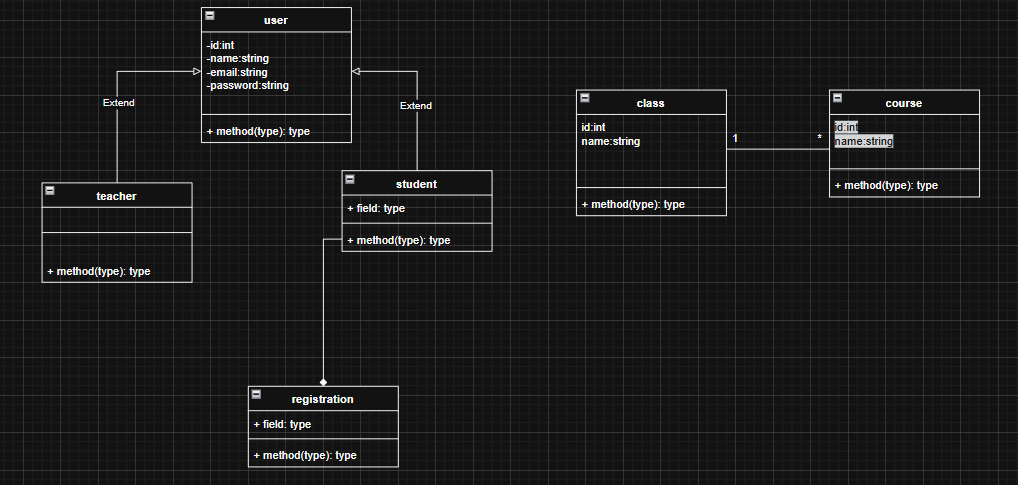
bài 6



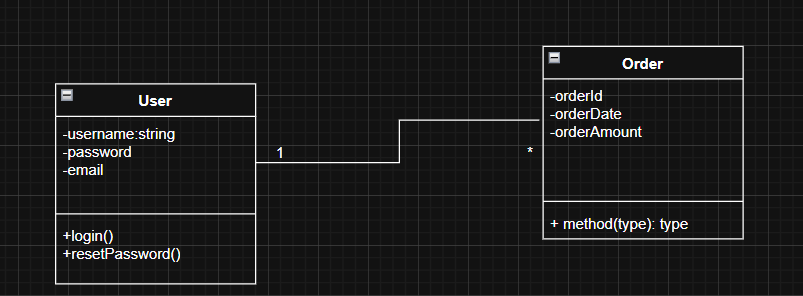
bài 7



bài 8



bài 9

lỗi:

-các thuộc tính trong class nên để private nhằm bảo mật tài khoản

-một user có thể có nhiều order nên phải là 1-N

-mối quan hệ phải là association với 2 class độc lập chỉ liên quan đến nhau

bài 10

