Ex1:

|  |  |  |
| --- | --- | --- |
| **Lớp (Class)** | **Vai trò** | **Thuộc tính chính** |
| **Book** | Đại diện thông tin cuốn sách | maSach, ten, tacGia, namXuatBan, available |
| **BookCopy** | Biểu diễn mỗi bản vật lý | copyId, maSach, trangThai |
| **Reader, Member (Độc giả)** | Người mượn, trả sách | maDocGia, ten, contact, danhSachMuonHienTai |
| **Librarian (Nhân viên)** | Thực thi nghiệp vụ (thêm, xử lý mượn, trả) | maNV, ten |
| **Loan / BorrowRecord** | Lưu một lần mượn cụ thể | maPhieu, maDocGia, maSach/maCopy, ngayMuon, ngayHenTra, ngayTra, trangThai |
| **Library / LibraryManager** | Điều phối, chứa danh sách sách, độc giả, phiếu; cung cấp API nghiệp vụ | books, copies, readers, loans |

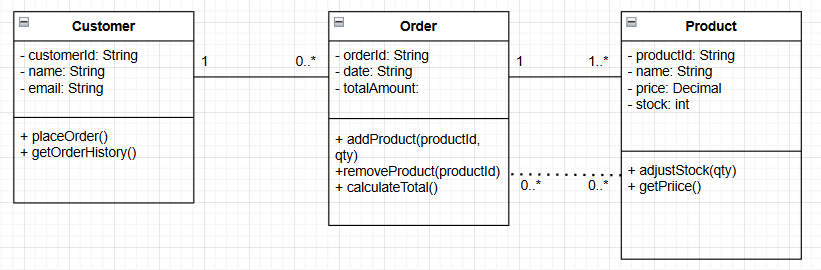
Ex2:

|  |  |  |  |
| --- | --- | --- | --- |
| **Mối quan hệ** | **Loại quan hệ** | **Giải thích** | **Ký hiệu UML** |
| **Một giáo viên giảng dạy nhiều lớp học** | **Association (Kết hợp)** | Quan hệ “giảng dạy” giữa hai lớp Giáo viên và Lớp học. Mỗi giáo viên có thể dạy nhiều lớp, và mỗi lớp có thể được dạy bởi một hoặc nhiều giáo viên tùy mô hình. Đây là mối quan hệ logic, không phụ thuộc sinh tồn. | Đường nối giữa 2 lớp, có thể kèm nhãn “giảng dạy”, và ký hiệu bội số: GiáoVien 1..\* ── 0..\* LopHoc |
| **Một đơn hàng gồm nhiều sản phẩm** | **Aggregation (Kết tập)** | Quan hệ “gồm có” giữa Đơn hàng và Sản phẩm. Đơn hàng chứa nhiều sản phẩm, nhưng nếu đơn hàng bị xóa thì sản phẩm vẫn tồn tại độc lập. (Quan hệ chứa **lỏng lẻo**). | Đường có **hình thoi rỗng** ở phía Đơn hàng: DonHang ◇── SảnPham |
| **Một cơ thể gồm nhiều bộ phận không thể tách rời** | **Composition (Cấu thành)** | Quan hệ “phần–toàn” giữa Cơ thể và Bộ phận. Các bộ phận là **một phần không thể tách rời** của cơ thể — nếu cơ thể bị hủy, các bộ phận cũng mất theo. (Quan hệ chứa **chặt chẽ**). | Đường có **hình thoi đặc** ở phía Cơ thể: CoThe ◆── BoPhan |

Ex3:

|  |  |  |
| --- | --- | --- |
| **Thành phần** | **Modifier** | **Giải thích** |
| username | private | Thông tin tài khoản nên được **đóng gói (encapsulation)** để tránh truy cập trực tiếp từ bên ngoài. Chỉ nên truy cập thông qua getter/setter. |
| password | private | Là thông tin **bảo mật**, tuyệt đối không được truy cập hoặc thay đổi trực tiếp. Chỉ nên được xử lý qua các phương thức an toàn (như login() hoặc resetPassword()). |
| lastLoginTime | private | Là thông tin nội bộ của hệ thống, chỉ nên được cập nhật bởi phương thức login() chứ không thể thay đổi từ bên ngoài. |
| login() | public | Là hành động người dùng thực hiện để đăng nhập, nên **công khai** để có thể gọi từ bên ngoài lớp. |
| resetPassword() | public | Dùng để người dùng (hoặc hệ thống) đặt lại mật khẩu, cần **truy cập được bên ngoài**, nên để public. |

Ex4:



**Mô tả chức năng:**

**Customer:** quản lý thông tin khách, tạo order, xem lịch sử.

**Order:** quản lý danh sách sản phẩm, tính tổng, trạng thái đơn.

**Product:** thông tin sản phẩm, giá, tồn kho, cập nhật kho.

Ex5:

Lớp: Order (Đơn hàng)

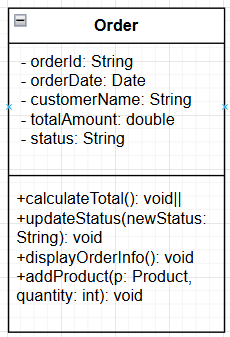
1. **Thuộc tính (Attributes)**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Mô tả** |
| orderId | String | Mã định danh duy nhất của đơn hàng. |
| orderDate | Date | Ngày đặt hàng. |
| customerName | String | Tên khách hàng đã đặt đơn hàng. |
| totalAmount | double | Tổng số tiền của đơn hàng. |
| status | String | Trạng thái đơn hàng (VD: “Đang xử lý”, “Hoàn thành”, “Đã hủy”). |

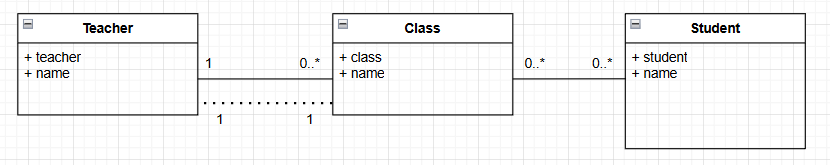
1. **Phương thức (Methods)**

|  |  |
| --- | --- |
| **Tên phương thức** | **Mô tả chức năng** |
| calculateTotal() | Tính tổng số tiền của tất cả sản phẩm trong đơn hàng. |
| updateStatus(String newStatus) | Cập nhật trạng thái của đơn hàng. |
| displayOrderInfo() | Hiển thị thông tin chi tiết đơn hàng. |
| addProduct(Product p, int quantity) *(tùy chọn)* | Thêm sản phẩm vào đơn hàng. |

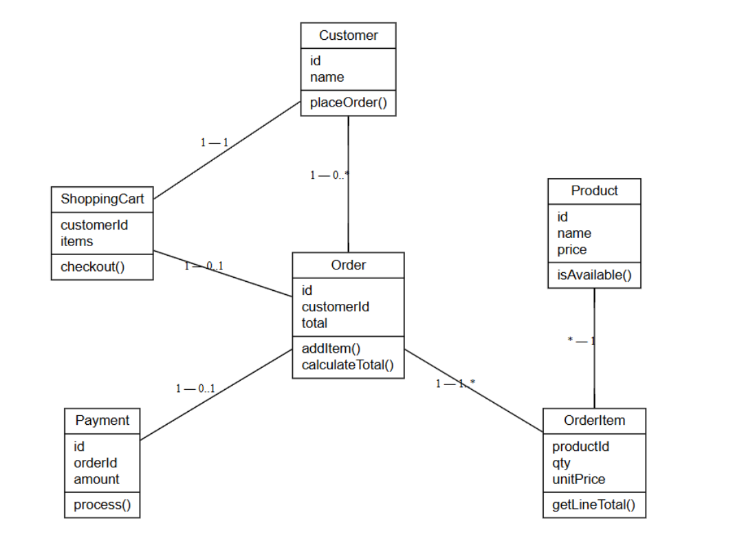
**3. Mô hình UML dạng text**



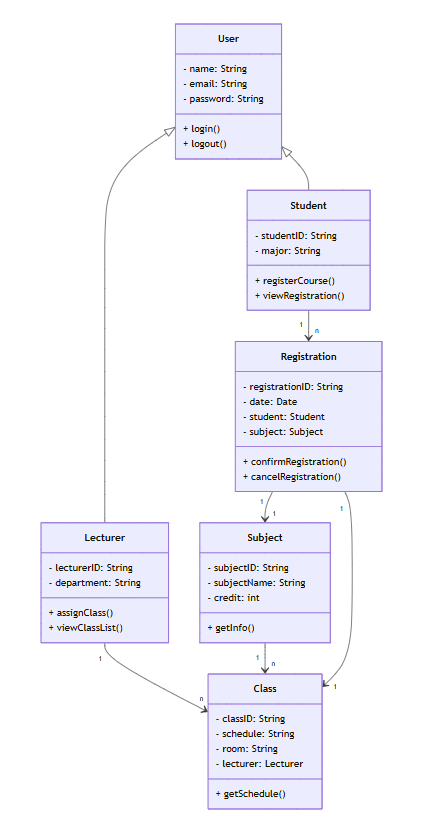
Ex6:



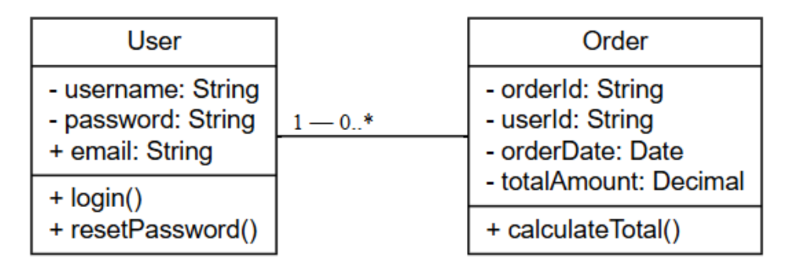
Ex7:



Ex8:



Ex9:



**Những lỗi đã sửa:**

* Multiplicity
* Access modifier
* Khóa tham chiếu
* Đặt tên nhất quán

BTTH:

