Bài 1:

Dưới đây là danh sách **class cốt lõi** rút ra trực tiếp từ mô tả và vai trò của chúng (ngắn gọn, đúng phạm vi “mượn–trả”):

| **Class** | **Vai trò (mô tả)** | **Thuộc tính gợi ý** | **Gợi ý quan hệ/ghi chú** |
| --- | --- | --- | --- |
| **Book** (Sách) | Đại diện **tựa sách** trong thư viện (thông tin mô tả chung). | bookId (mã), title (tên), author, publishedYear | 1 *Book* có thể có nhiều **BookCopy**. |
| **BookCopy** (Bản sao sách) | Mỗi **cuốn sách vật lý** có mã riêng để quản lý mượn–trả. | copyId, status (available/loaned/lost), shelfLocation | Nhiều *BookCopy* thuộc 1 *Book*. Tham gia giao dịch **Loan**. |
| **Reader** (Độc giả) | Người dùng mượn và trả sách. | readerId, name, contact, membershipStatus | 1 *Reader* có 0..n **Loan**. |
| **Librarian** (Nhân viên) | Xử lý nghiệp vụ mượn–trả, cập nhật hệ thống. | staffId, name, role | Thực hiện/duyệt **Loan**. |
| **Loan** (Phiếu mượn) | Giao dịch **mượn–trả** giữa *Reader* và *BookCopy*. | loanId, loanDate, dueDate, returnDate, fineAmount | Liên kết: 1 *Loan* ↔ 1 *Reader*, 1 *BookCopy*, (tùy chọn) 1 *Librarian*. returnDate dùng cho “trả sách”. |

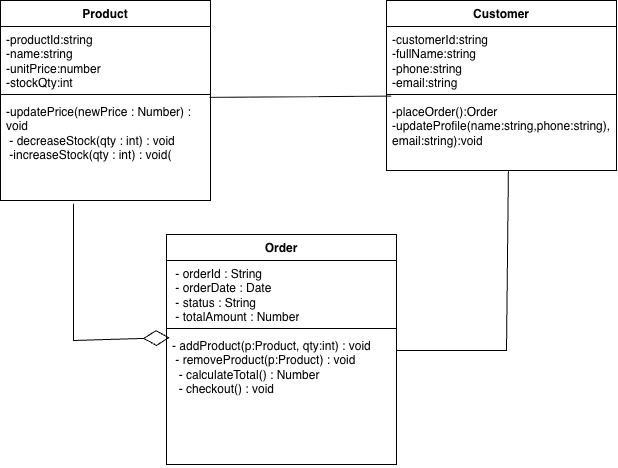
Bài2 :

| **Mối quan hệ** | **Loại** | **Giải thích ngắn** | **Ký hiệu UML (nếu có)** |
| --- | --- | --- | --- |
| **“Một giáo viên giảng dạy nhiều lớp học.”** | **Association** (kết hợp thông thường) | Hai lớp **tồn tại độc lập**, liên kết bằng vai trò “giảng dạy”. Nếu giáo viên nghỉ việc hay lớp giải thể, lớp *khác* và giáo viên *khác* vẫn có thể tồn tại; không có vòng đời phụ thuộc. | Đường thẳng có/không tên vai trò: Teacher "1" — "0..\*" Class (có thể thêm vai trò teaches, taughtBy). |
| **“Một đơn hàng gồm nhiều sản phẩm.”** | **Aggregation** (tập hợp – “has-a/whole–part” yếu) | **Đơn hàng** là “whole”, **Sản phẩm** là “part”, **part có thể tồn tại độc lập** với whole (sản phẩm vẫn tồn tại trong kho, có thể thuộc đơn hàng khác). Quan hệ whole–part nhưng **không ràng buộc vòng đời**. | Hình **diamond rỗng** ở phía whole: Order ◇— "1" Order "0..\*" Product. |
| **“Một cơ thể gồm nhiều bộ phận không thể tách rời.”** | **Composition** (thành phần – whole–part mạnh) | **Cơ thể** là whole, **bộ phận** là part; **vòng đời phụ thuộc chặt**: part **không tồn tại độc lập** ngoài whole (vd. *Tim*, *Não* của *Cơ thể*). Khi whole bị hủy, các part cũng mất. | Hình **diamond đặc** ở phía whole: Body ◆— "0..\*" Organ (hoặc Body ◆— Head/Torso/...). |

Bài 3:

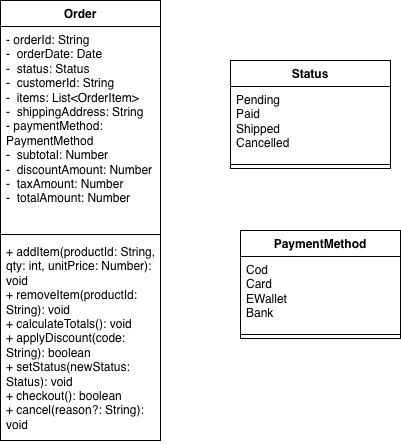
| **Thành phần** | **Modifier** | **Giải thích** |
| --- | --- | --- |
| username | private (hoặc protected nếu muốn lớp con dùng trực tiếp) | Thông tin định danh người dùng. Không nên cho truy cập trực tiếp từ bên ngoài đối tượng để tránh sửa sai / giả mạo username. Truy cập nên thông qua getter/setter hợp lệ. |
| password | private | Mật khẩu là dữ liệu nhạy cảm. Không được phép truy cập trực tiếp từ bên ngoài hay từ lớp khác. Chỉ nội bộ class User được đọc/ghi (thường cũng không lưu plaintext mà lưu hash). |
| lastLoginTime | private (đôi khi protected) | Thời điểm đăng nhập gần nhất. Đây là thông tin liên quan bảo mật/tài khoản. Ta không muốn code bên ngoài tự ý chỉnh sửa timestamp này. Có thể cho phép đọc qua getter, hoặc để protected nếu lớp con như AdminUser cần truy cập. |
| login() | public | Hành vi mà mã bên ngoài (ví dụ: màn hình đăng nhập, API) sẽ gọi để thực hiện đăng nhập. Phải có khả năng gọi từ ngoài hệ thống, nên để public. |
| resetPassword() | public | Được gọi từ bên ngoài (ví dụ: “Quên mật khẩu?”). Vì đây là hành động nghiệp vụ mà hệ thống khác / UI cần kích hoạt, nó nên là public. (Bên trong hàm có thể kiểm tra OTP/email/token.) |

Bài4:

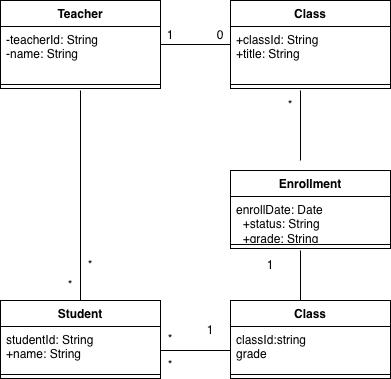


Bài5:

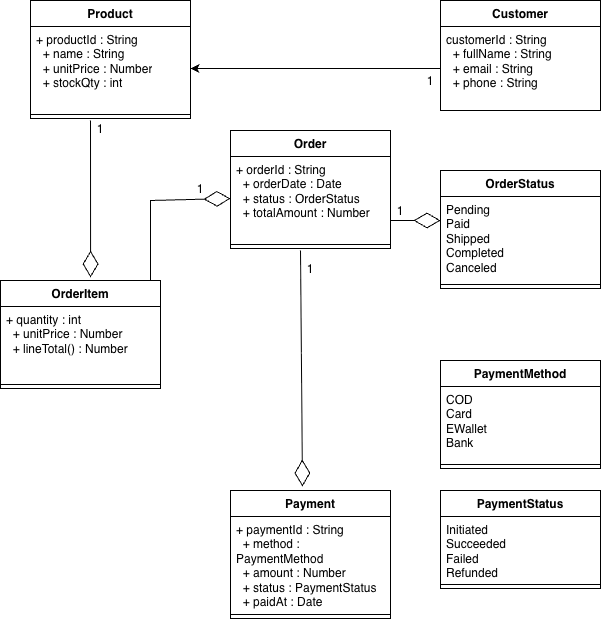
| **Loại** | **Tên** | **Kiểu dữ liệu (gợi ý)** | **Mô tả ngắn** |
| --- | --- | --- | --- |
| **Thuộc tính** | orderId | String | Mã đơn hàng duy nhất |
|  | orderDate | Date | Ngày tạo đơn |
|  | status | Enum{Pending, Paid, Shipped, Completed, Canceled} | Trạng thái xử lý |
|  | customerId | String | Mã khách hàng tạo đơn |
|  | items | List<OrderItem> | Danh sách dòng hàng (sản phẩm + số lượng + đơn giá) |
|  | shippingAddress | String | Địa chỉ giao hàng |
|  | paymentMethod | Enum{COD, Card, EWallet, Bank} | Phương thức thanh toán |
|  | subtotal | Number | Tổng tạm tính (không gồm thuế/giảm giá) |
|  | discountAmount | Number | Tiền giảm giá |
|  | taxAmount | Number | Thuế |
|  | totalAmount | Number | Tổng thanh toán cuối cùng |
| **Loại** | **Tên phương thức** | **Chữ ký (gợi ý)** | **Mục đích** |
| **Phương thức** | addItem | addItem(productId: String, qty: int, unitPrice: Number): void | Thêm 1 dòng hàng |
|  | removeItem | removeItem(productId: String): void | Xoá dòng hàng theo sản phẩm |
|  | calculateTotals | calculateTotals(): void | Tính subtotal, taxAmount, totalAmount |
|  | applyDiscount | applyDiscount(code: String): boolean | Áp mã giảm giá, cập nhật discountAmount |
|  | setStatus | setStatus(newStatus: Status): void | Cập nhật trạng thái đơn |
|  | checkout | checkout(): boolean | Khóa đơn + tạo giao dịch thanh toán |
|  | cancel | cancel(reason?: String): void | Huỷ đơn nếu chưa giao |



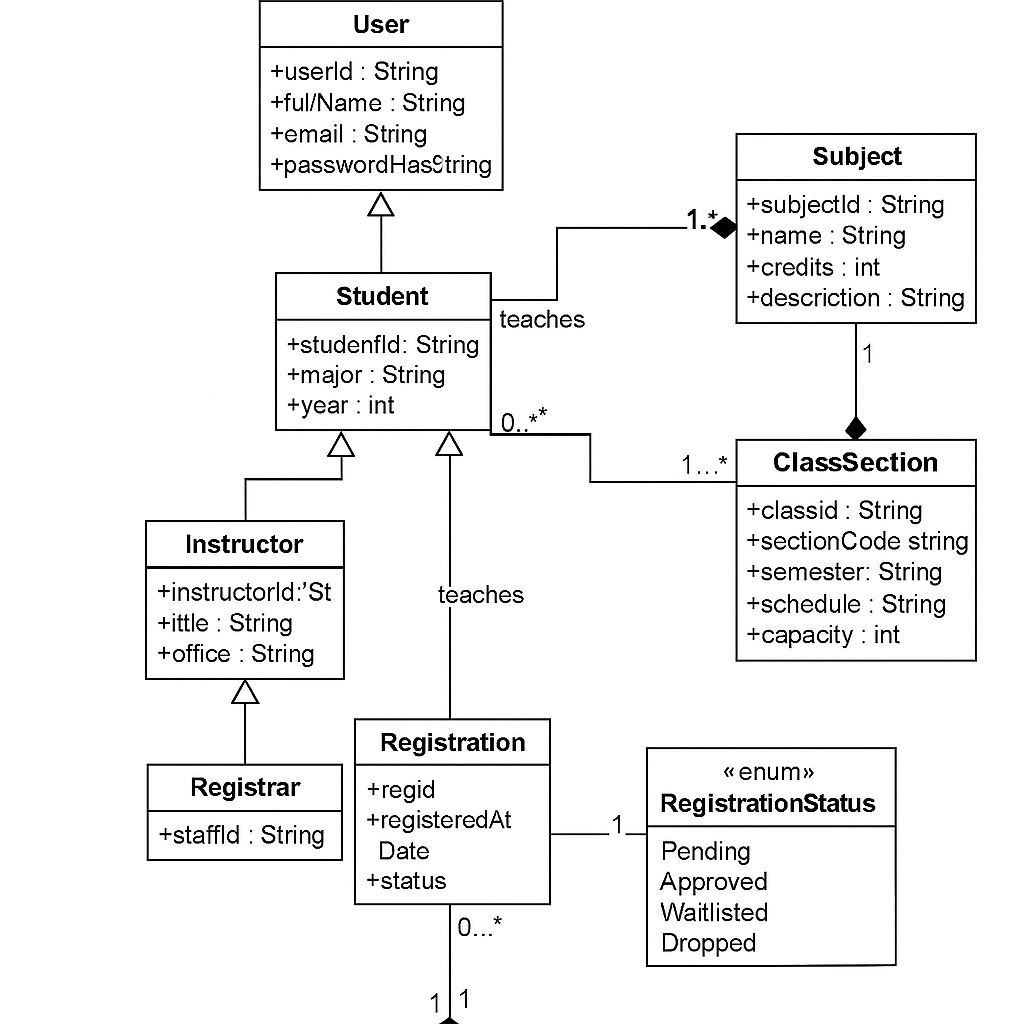
Bài6:



Bài7:



BÀI8:



BTTH:

