**Bài 1 :**

|  |  |  |
| --- | --- | --- |
| **Class** | **Vai trò** | **Thuộc tính** |
| Sách | Quản lí thông tin sách | maSach , tenSach , tacGia , nsb , tinhTrang |
| Độc giả | Đại diện người mượn sách | maDocGia , hoTen , diaChi , soDienThoai |
| Nhân viên | Quản trị thư viện | maNhanVien , hoTen , chucVu |
| Phiếu mượn | Giao dịch mượn | maPhieu , ngayMuon , ngayTra , docGia , sach |
| Hệ thống thư viện | Quản lý toàn cục | danhSachSach , danhSachDocGia , danhSachPhieuMuon |

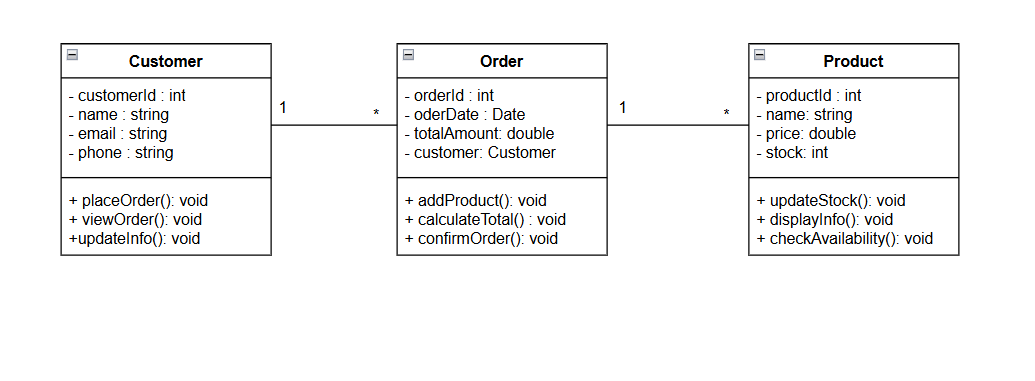
**Bài 2 :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Mối quan hệ** | **Loại quan hệ** | **Giải thích** | **Ký hiệu UML** |
| Một giáo viên giảng dạy nhiều lớp học | Association (Liên kết) | Là mối quan hệ logic giữa hai lớp Giáo viên và Lớp học, thể hiện rằng một giáo viên có thể dạy nhiều lớp, và mỗi lớp có thể do một hoặc nhiều giáo viên giảng dạy. Hai đối tượng tồn tại độc lập, không phụ thuộc vòng đời. | Đường nối liền giữa hai lớp, có thể ghi nhãn 1..\* ở phía Lớp học và 1 ở phía Giáo viên. |
| Một đơn hàng gồm nhiều sản phẩm | Aggregation (Kết hợp - chứa, nhưng không phụ thuộc) | Đơn hàng chứa nhiều sản phẩm, nhưng nếu đơn hàng bị xóa thì sản phẩm vẫn tồn tại trong hệ thống (ví dụ: sản phẩm còn được bán trong các đơn hàng khác). Đây là mối quan hệ “có - chứa”, nhưng không có tính sống chết cùng nhau. | Đường nối có hình thoi rỗng (◇) ở phía lớp Đơn hàng. |
| Một cơ thể gồm nhiều bộ phận không thể tách rời | Composition (Cấu thành - phụ thuộc toàn phần) | Cơ thể gồm nhiều bộ phận (tim, phổi, gan…). Nếu cơ thể bị huỷ thì các bộ phận cũng không tồn tại độc lập. Đây là quan hệ sống chết cùng nhau (strong ownership). | Đường nối có hình thoi đặc (◆) ở phía lớp Cơ thể. |

**Bài 3 :**

| **Thành phần** | **Modifier (Phạm vi truy cập)** | **Giải thích** |
| --- | --- | --- |
| username | private | Dữ liệu nhạy cảm của người dùng, chỉ nên được truy cập thông qua **phương thức getter/setter** để đảm bảo tính bảo mật và kiểm soát việc thay đổi. |
| password | private | Là **thông tin tuyệt mật**, cần được ẩn hoàn toàn khỏi các lớp khác. Không được phép truy cập hoặc sửa trực tiếp từ bên ngoài, chỉ thông qua các hàm xử lý an toàn (ví dụ: resetPassword()). |
| login() | public | Là **chức năng chính** mà các lớp khác (ví dụ: giao diện người dùng, hệ thống xác thực) cần gọi để đăng nhập, nên được công khai. |
| resetPassword() | public | Cần cho phép người dùng hoặc hệ thống gọi để đặt lại mật khẩu khi quên, nên được công khai. |
| lastLoginTime | private | Là dữ liệu nội bộ (lưu thời điểm đăng nhập gần nhất), không nên cho truy cập trực tiếp; có thể cung cấp qua phương thức getLastLoginTime() nếu cần. |

**Bài 4 :**



**\* Chức năng (vai trò) của từng lớp**

- Class: Product

Vai trò: Quản lý thông tin chi tiết của sản phẩm trong hệ thống.

Thuộc tính chính:

productID, name, price, stock

Phương thức chính:

updateStock(): Cập nhật số lượng tồn kho khi có đơn hàng.

displayInfo(): Hiển thị thông tin sản phẩm.

checkAvailability(): Kiểm tra tình trạng còn hàng.

- Class: Customer

Vai trò: Đại diện cho khách hàng sử dụng hệ thống để mua hàng.

Thuộc tính chính:

customerID, name, email, phone

Phương thức chính:

placeOrder(): Tạo đơn hàng mới.

viewOrders(): Xem danh sách các đơn hàng của mình.

updateInfo(): Cập nhật thông tin cá nhân.

- Class: Order

Vai trò: Quản lý thông tin về các giao dịch mua hàng (đơn hàng).

Thuộc tính chính:

orderID, orderDate, totalAmount, customer

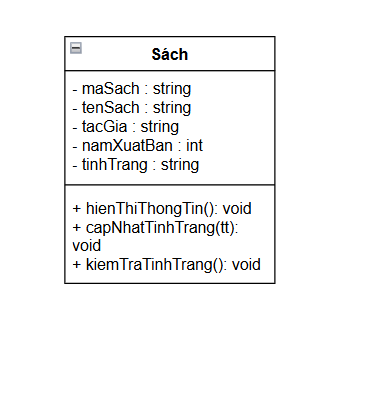
Phương thức chính:

addProduct(): Thêm sản phẩm vào đơn hàng.

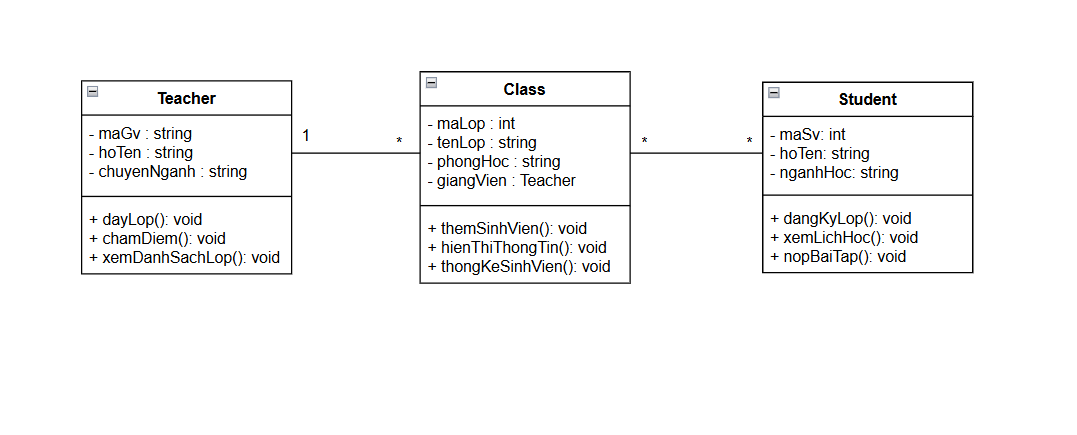
calculateTotal(): Tính tổng tiền đơn hàng.

confirmOrder(): Xác nhận và hoàn tất đơn hàng.

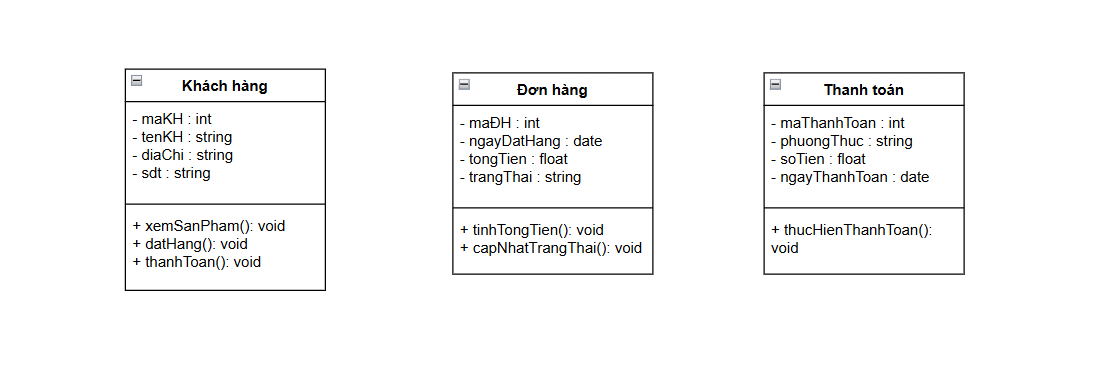
**Bài 5 :**



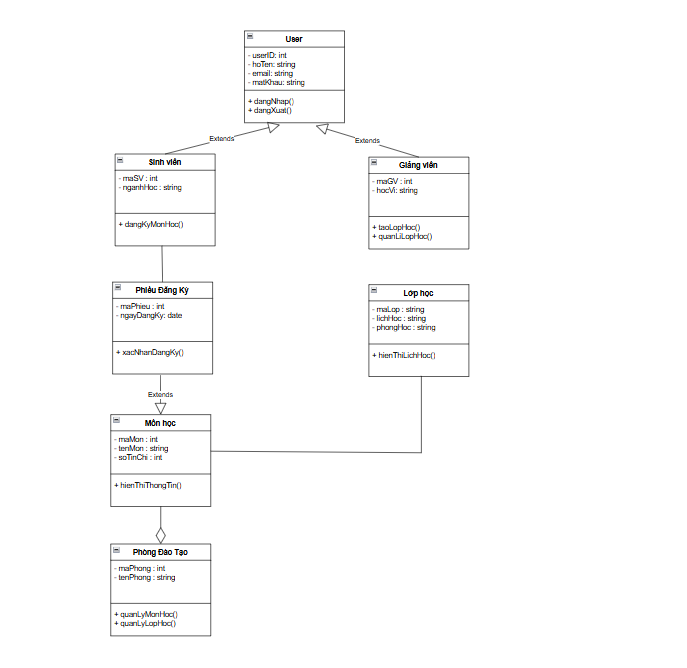
**Bài 6 :**



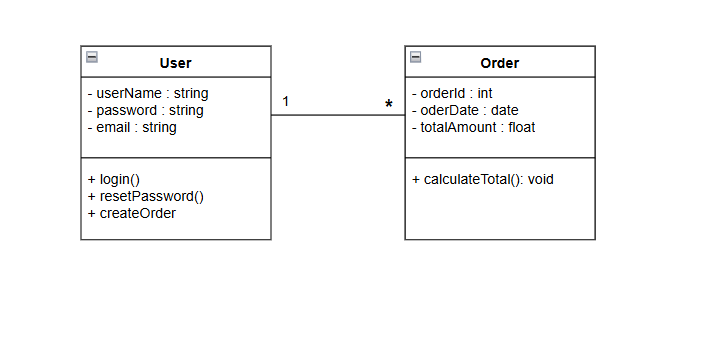
**Bài 7 :**



**Bài 8 :**



**Bài 9 :**



**Bài 10 :**

