**TÀI LIỆU THIẾT KẾ HƯỚNG ĐỐI TƯỢNG (MÔ HÌNH LỚP)**

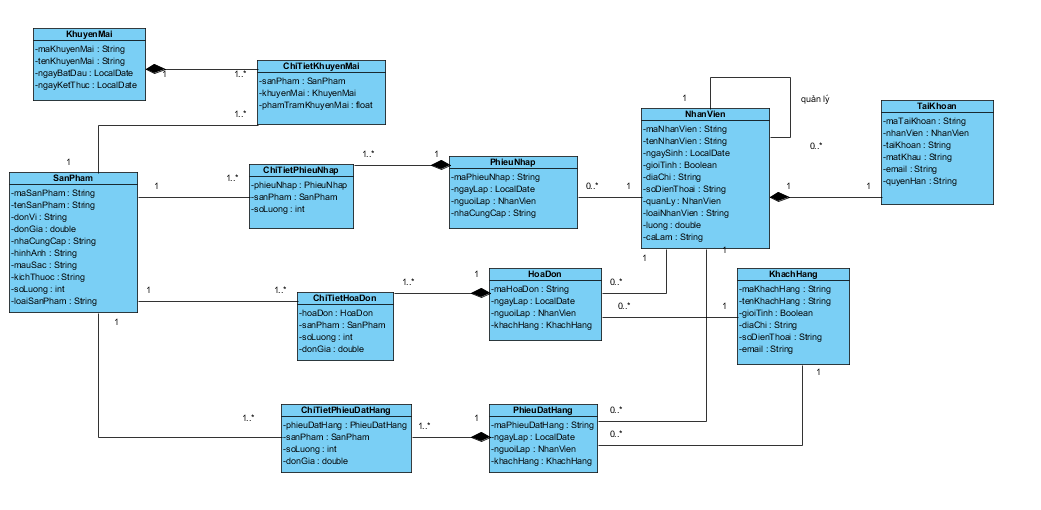
Nhóm 12 - Thành viên nhóm

1. Trương Chi Bảo (Nhóm trưởng)
2. NguyễnVăn Đạt
3. Văn Chí Hiếu

*Tên ứng dụng:* **CHƯƠNG TRÌNH QUẢN LÝ BÁN HÀNG TẠI CỬA HÀNG BÁN QUẦN ÁO THỜI TRANG**

Thời gian thực hiện: Từ 28/08/2023 đến 12/11/2023 (11 tuần)

1. **Mô hình lớp (Class Diagram)**

****

*Hình 1: Mô hình lớp (Class Diagram)*

1. **Đặc tả Class**

## **2.1 Thực thể TaiKhoan:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maTaiKhoan | String | maTaiKhoan chính là mã nhân viên của nhân viên.  Mã nhân viên có 6 ký tự.  Hai ký tự đầu tiên là "TK"  Bốn ký tự cuối cùng là số từ 0001 đến 1000. | {id}:VD: Mã tài khoản :  maTaiKhoan: "TK0001" |
| 1.2 | tenTaiKhoan | String | Không được rỗng. không chứa ký tự đặc biệt ,tối thiểu 6 ký tự và tối đa 20 ký tự | taiKhoan: "john\_doe"  Tài khoản với email làm tên đăng nhập:  taiKhoan: "john.doe@example.com"  Tài khoản với mã số sinh viên:  taiKhoan: "S123456" |
| 1.3 | nhanVien | NhanVien |  |  |
| 1.4 | matKhau | String | Không được rỗng,tối thiểu 8 ký tự và tối đa 20 ký tự |  |
| 1.5 | email | String | Chuỗi username@domain.tld  Trong đó:  username là một chuỗi ký tự có thể chứa chữ cái, số, dấu gạch dưới, dấu gạch nối và dấu chấm.  domain là một chuỗi ký tự có thể chứa chữ cái, số, dấu gạch dưới và dấu chấm.  tld là một chuỗi ký tự có thể chứa hai đến sáu chữ cái. |  |
| 1.6 | quyenHan | String | Không được để trống. Mỗi tài khoản cần được gán một quyền hạn cụ thể. | Vídụ:"ADMIN," "USER,"hoặc "STAFF." |
| **2** | **Khai báo phương thức getter/setter :** |  |  |  |
| 2.1 | setMaTaiKhoan(String maTaiKhoan) | void | Kiểm tra tham số maTaiKhoan không được rỗng | throw exception “oyr” nếu tham số maTaiKhoan không hợp lệ |
| 2.2 | setTaiKhoan(String taiKhoan) | void | Kiểm tra tham số taiKhoan không được rỗng | Nếu taiKhoan là null hoặc rỗng, có thể ném ra ngoại lệ hoặc xử lý lỗi |
| 2.3 | setNhanVien(NhanVien nhanVien) | void | Kiểm tra tham số nhanVien không được rỗng | Nếu nhanVien là null, có thể ném ra một ngoại lệ hoặc xử lý lỗi. |
| 2.4 | setMatKhau(String matKhau) | void | Kiểm tra tham số matKhau không được rỗng | throw exception “Mật khẩu không được rỗng” nếu tham số matKhau không hợp lệ |
| 2.5 | setEmail(String email) | void |  |  |
| 2.6 | setQuyenHan(String quyenHan) | void |  |  |
| 2.7 | getMaTaiKhoan() | String |  |  |
| 2.8 | getTenTaiKhoan() | String |  |  |
| 2.9 | getNhanVien() | NhanVien |  |  |
| 2.10 | getMatKhau() | String |  |  |
| 2.11 | getEmail() | String |  |  |
| 2.12 | getQuyenHan() | String |  |  |
| **3** | **Viết các constructor, hashCode, equals** |  |  |  |
| 3.1 | TaiKhoan() |  |  |  |
| 3.2 | TaiKhoan(maTaiKhoan) |  |  |  |
| 3.3 | TaiKhoan(maTaiKhoan, nhanVien, taiKhoan, matKhau,email,quyenHan) |  |  |  |
| **4** | **Viết phương thức toString()** | String |  | trả về chuỗi chứa giá trị các thuộc tính của tài khoản |

* 1. **Thực thể NhanVien:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maNhanVien | String | Mã có độ dài 6 ký tự.  Hai ký tự đầu tiên là "NV" Bốn ký tự cuối cùng là một số tự tăng từ 0001 đến 1000. | Ví dụ: ”NV0001”,  “NV0012”,”NV0100” |
| 1.2 | tenNhanVien | String | không được rỗng, không chứa các ký tự đặc biệt, mỗi chữ cái đầu của mỗi từ là chữ in hoa | Ví dụ:  Tên hợp lê:  “Nguyễn Văn A”, ”Lê Thị B”  Tên không hợp lệ:  “A.B”,”,,,Nguy ễn Văn” |
| 1.3 | ngaySinh | LocalDate | Ngày sinh phải là một ngày hợp lệ, phải nằm trong một phạm vi có thể xác định (ví dụ: không thể có ngày 30/02 vì tháng 2 chỉ có 28 hoặc 29 ngày).  Ngày sinh của một người phải nhỏ hơn ngày hiện tại để đảm bảo người đó đã sinh ra trước ngày hiện tại. |  |
| 1.4 | gioiTinh | boolean | Gồm 2 giá trị: True và False | True là nam,  False là nữ |
| 1.5 | diaChi | String | Chuỗi các ký tự không bao gồm các ký tự đặc biệt có ít nhất 2 ký tự và tối đa 255 ký tự |  |
| 1.6 | soDienThoai | String | Chuỗi gồm các số có tối đa 10 ký số |  |
| 1.7 | quanLy | NhanVien | Đối tượng NhanVien là một nhân viên có thể làm vai trò người quản lý.  Tuổi của người quản lý phải lớn hơn 18 tuổi. |  |
| 1.8 | loaiNhanVien | String | Gồm 2 loại:  nhân viên quản lý và nhân viên bán hàng |  |
| 1.9 | luong | double | >0 |  |
| 1.10 | caLam | String | Gồm 2 ca:  Sáng và tối | Ca sáng bắt đầu từ 9g sáng đến 4 g chiều  Ca tối bắt đầu từ 4g chiều đến 10 đêm |
| **2** | **Khai báo phương thức getter/setter:** |  |  |  |
| 2.1 | setMaNhanVien(String maNhanVien ) | void | Phát sinh day số theo quy luật dựa trên ràng buộc đã đưa ra | Phát sinh tự động |
| 2.2 | setTenNhanVien(String tenNhanVien) | void | Tên không được rỗng | throw exception “TenNhanVien không được rỗng” nếu tham số rỗng |
| 2.3 | setngaySinh(LocalDate ngaySinh ) | void | Ngày sinh phải nhỏ hơn ngày hiện tại ,và ngày hiện tại trừ ngày sinh phải lớn hơn 18 tuổi |  |
| 2.4 | setGioiTinh(boolean gioiTinh ) | void | Giới tính không được rỗng | throw exception “gioiTinh không không được rỗng” nếu tham số rỗng |
| 2.5 | setDiaChi(String diaChi) | void |  |  |
| 2.6 | setSoDienThoai(String soDienThoai) | void | Số điện thoại không được rỗng | throw exception “số điện thoại không không được rỗng” nếu tham số soDienThoai rỗng |
| 2.7 | setquanLy(NhanVien quanLy) | void | Tuổi phải lớn hơn 18 |  |
| 2.8 | setLoaiNhanVien(String loaiNhanVien) | void |  |  |
| 2.9 | setLuong(double luong) | void | Lương không được rỗng | throw exception “lương không không được rỗng” nếu tham số luong rỗng |
| 2.10 | setCaLam(String caLam) | void | Ca làm không được rỗng | throw exception “ca làm không không được rỗng” nếu tham số caLam rỗng |
| 2.11 | getMaNhanVien() | String |  |  |
| 2.12 | getTenNhanVien(tenNhanVien) | String |  |  |
| 2.13 | getngaySinh(ngaySinh ) | LocalDate |  |  |
| 2.14 | getGioiTinh() | boolean |  |  |
| 2.15 | getDiaChi() | String |  |  |
| 2.16 | getSoDienThoai() | String |  |  |
| 2.17 | getquanLy() | NhanVien |  |  |
| 2.18 | getLoaiNhanVien() | String |  |  |
| 2.19 | getLuong() | double |  |  |
| 2.20 | getCaLam() | String |  |  |
| **3** | **Viết các constructor, hashCode, equals** |  |  |  |
| 3.1 | NhanVien() |  |  |  |
| 3.2 | NhanVien(maNhanVien) |  |  |  |
| 3.3 | NhanVien(maNhanVien, tenNhanVien, ngaySinh, gioiTinh, diaChi, soDienThoai, quanLy,loaiNhanVien,luong,caLam) |  |  |  |
| **4** | **Viết phương thức toString()** | String |  | Trả về chuối chứa giá trị các thuộc tính NhanVien |

* 1. **Thực thể KhuyenMai:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maKhuyenMai | String | maKhuyenMai phải có độ dài là 6 ký tự.  Hai ký tự đầu của maKhuyenMai phải là "KM".  Bồn ký tự cuối cùng của maKhuyenMai phải là số thứ tự từ 0001 đến 1000. | Vd: "KM0001", "KM0025", "KM0099". |
| 1.2 | tenKhuyenMai | String | tenKhuyenMai không được để trống. |  |
| 1.3 | ngayBatDau | LocalDate | ngayBatDau phải là một ngày trong quá khứ, nghĩa là không thể đặt ngayBatDau ở tương lai hoặc vào ngày hiện tại.  ngayBatDau không được để trống (không được là null) |  |
| 1.4 | ngayKetThuc | LocalDate | ngayKetThuc phải là một ngày sau hoặc bằng ngayBatDau. Nghĩa là ngayKetThuc không thể trước ngày bắt đầu hoặc bằng ngày bắt đầu.  ngayKetThuc không được để trống (không được là null). |  |
| **2** | **Khai báo phương thức getter/setter:** |  |  |  |
| 2.1 | setMaKhuyenMai(String maKhuyenMai) | void | Phát sinh day số theo quy luật dựa trên ràng buộc đã đưa ra | Phát sinh tự động |
| 2.2 | setTenKhuyenMai(String tenKhuyenMai) | void |  |  |
| 2.3 | setNgayBatDau(LocalDate ngayBatDau) | void | ngayBatDau không được để trống (không được là null).  ngayBatDau phải là một ngày trong quá khứ hoặc bằng ngày hiện tại. |  |
| 2.4 | setNgayKetThuc(LocalDate ngayKetThuc) | void | ngayKetThuc không được để trống (không được là null).  ngayKetThuc phải là một ngày sau hoặc bằng ngayBatDau. |  |
| 2.5 | getMaKhuyenMai() | SS String |  |  |
| 2.6 | getTenKhuyenMai() | String |  |  |
| 2.7 | getNgayBatDau() | LoalDate |  |  |
| 2.8 | getNgayKetThuc() | LoalDate |  |  |
| **3** | **Viết các constructor, hashCode, equals** |  |  |  |
| 3.1 | KhuyenMai() |  |  |  |
| 3.2 | KhuyenMai(maKhuyenMai) |  |  |  |
| 3.3 | KhuyenMai(maKhuyenMai, tenKhuyenMai, ngayBatDau, ngayKetThuc) |  |  |  |
| **4** | **Viết phương thức toString()** | String |  | Trả về một chuỗi chứa thông tin về đối tượng KhuyenMai |

* 1. **Thực thể KhachHang:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maKhachHang | String | maKhachHang phải có độ dài là 6 ký tự.  Hai ký tự đầu của maKhachHang phải là "KH" để chỉ định mã của khách hàng.  Bốn ký tự cuối cùng của maKhachHang phải là số tự tăng từ 0001 đến 1000 để đảm bảo tính duy nhất của mã. | Vd: "KH0001", "KH0012", "KH0099". |
| 1.2 | tenKhachHang | String | không được rỗng, không chứa các ký tự đặc biệt, mỗi chữ cái đầu của mỗi từ là chữ in hoa | Ví dụ:  Tên hợp lê:  “Nguyễn Văn A”, ”Lê Thị B”  Tên không hợp lệ:  “A.B”,”,,,Nguy ễn Văn” |
| 1.3 | gioiTinh | boolean | Gồm 2 giá trị: Trua và False | True là nam ,False là nữ |
| 1.4 | diaChi | String | Chuỗi các ký tự không bao gồm các ký tự đặc biệt có ít nhất 2 ký tự và tối đa 255 ký tự |  |
| 1.5 | soDienThoai | String | Chuỗi gồm các số có tối đa 10 ký số |  |
| 1.6 | Email | String | Chuỗi username@domain.tld  Trong đó:  username là một chuỗi ký tự có thể chứa chữ cái, số, dấu gạch dưới, dấu gạch nối và dấu chấm.  domain là một chuỗi ký tự có thể chứa chữ cái, số, dấu gạch dưới và dấu chấm.  tld là một chuỗi ký tự có thể chứa hai đến sáu chữ cái. |  |
| **2** | **Khai báo phương thức getter/setter:** |  |  |  |
| 2.1 | setMaKhachHang(String maKhachHang) | void | Phát sinh dãy số theo quy luật dựa trên ràng buộc đã đưa ra | phát sinh tự động |
| 2.2 | setTenKhachHang(String tenKhachHang) | void | Tên khách hàng không được để trống | throw exception “tên khách hàng không được bỏ trống “ |
| 2.3 | setGioiTinh(Boolean gioiTinh ) | void |  | True: nam, false: nữ |
| 2.4 | setdiaChi(String diaChi) | void |  | throw exception “Địa chỉ không được bỏ trống” |
| 2.5 | setSoDienThoai(String soDienThoai) | void |  | throw exception “Số điện thoại không được bỏ trống và phải là ký tự số” |
| 2.6 | setEmail(String email) | void |  |  |
| 2.7 | getMaKhachHang() | String |  |  |
| 2.8 | getTenKhachHang() | String |  |  |
| 2.9 | getGioiTinh() | Boolean |  |  |
| 2.10 | getdiaChi() | String |  |  |
| 2.11 | getSoDienThoai() | String |  |  |
| 2.12 | getEmail() | String |  |  |
| **3** | **Viết các constructor, hashCode, equals** |  |  |  |
| 3.1 | KhachHang() |  |  |  |
| 3.2 | KhachHang(maKhachHang) |  |  |  |
| 3.3 | KhachHang(maKhachHang, tenKhachHang, gioiTinh, diaChi, soDienThoai, email) |  |  |  |
| **4** | **Viết phương thức toString()** | String |  | Trả về chuối chứa giá trị các thuộc tính khách hàng |

* 1. **Thực thể ChiTietKhuyenMai:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | sanPham | SanPham | sanPham không được để trống (không được là null). |  |
| 1.2 | khuyenMai | KhuyenMai | khuyenMai không được để trống (không được là null). |  |
| 1.3 | phamTramKhuyenMai | float | phamTramKhuyenMai phải là một giá trị từ 0 đến 100 (bao gồm cả 0 và 100). |  |
| **2** | **Khai báo phương thức getter/setter:** |  |  |  |
| 2.1 | setSanPham(SanPham sanPham) | void |  |  |
| 2.2 | setKhuyenMai(KhuyenMai khuyenMai) | void |  |  |
| 2.3 | setPhamTramKhuyenMai(float phamTramKhuyenMai) | void |  |  |
| 2.4 | getSanPham() | SanPham |  |  |
| 2.5 | getKhuyenMai() | KhuyenMai |  |  |
| 2.6 | getPhamTramKhuyenMai() | float |  |  |
| **3** | **Viết các constructor, hashCode, equals** |  |  |  |
| 3.1 | ChiTietKhuyenMai() |  |  |  |
| 3.2 | ChiTietKhuyenMai(SanPham sanPham, KhuyenMai khuyenMai, float phamTramKhuyenMai) |  |  |  |
| **4** | **Viết phương thức toString()** | String |  | trả về một chuỗi chứa thông tin về đối tượng ChiTietKhuyenMai |

* 1. **Thực thể SanPham:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi Chú** |
| 1. | **Khai báo thuộc tính** |  |  |  |
| 1.1 | maSanPham | String | Mã sản phẩm có 6 ký tự:  Hai ký tự đầu là SP  Bốn ký tự sau là số thứ tự từ 0001-1000 | Vd: "SP0001", "SP0012", "SP0099". |
| 1.2 | tenSanPham | String | Không được rỗng, không chứa ký tự đặc biệt. |  |
| 1.3 | donVi | String | Không được rỗng, không chứa ký tự đặc biệt. |  |
| 1.4 | donGia | double | > 0 |  |
| 1.5 | nhaCungCap | String | Không được rỗng, không chứa ký tự đặc biệt. |  |
| 1.6 | hinhAnh | String | Không được rỗng |  |
| 1.7 | mauSac | String | Gồm một số màu:  Đen,trắng,xanh,đỏ,vàng,xanh lá,hồng |  |
| 1.8 | kichThuoc | String | Gồm các kích thước:  “S”,”M”,”L”,”XL”,”XXL” |  |
| 1.9 | soLuong | int | > 0 |  |
| 1.10 | loaiSanPham | String | Gồm các loại:”quần”,”áo”,”váy”,”áo khoác” |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaSanPham(String maSanPham) | Void | Phát sinh dãy số theo quy luật dựa trên ràng buộc đã đưa ra | Phát sinh tự động |
| 2.2 | setTenSanPham(String tenSanPham) | void | Kiểm tra tham số tenSanPham không được trống | Throw exception “Tên sản phẩm không được để trống” khi nhập vào mã rỗng. |
| 2.3 | setDonVi | void |  |  |
| 2.4 | setDonGia(double donGia) | void | Kiểm tra tham số donGia không được <= 0 |  |
| 2.5 | setNhaCungCap(String nhaCungCap) | Void | Kiểm tra tham nhaCungCap không được trống | Throw exception “Nhà cung cấp không được để rỗng” khi tham số kích thước nhập vào không hợp lệ |
| 2.6 | setHinhAnh(String anhSanPham) | void | Kiểm tra tham số anhSanPham chỉ lấy những ảnh có đúng 1 trong những định dạng: png, jpg, raw, jpeg | Throw new Exception("Chỉ chấp nhận các ảnh có định dạng png, jpg, raw, jpeg") khi ảnh chọn không đúng định dạng (png, jpg, raw, jpeg) |
| 2.7 | setMauSac(String mauSac) | void | Kiểm tra tham số mauSac không được rỗng | Throw exception “Màu sắc không được để rỗng” khi tham số kích thước nhập vào không hợp lệ |
| 2.8 | setKichThuoc(String kichThuoc) | void | Kiểm tra tham số kichThuoc không được rỗng | Throw exception “Kích thước không được để rỗng” khi tham số kichThuoc là null |
| 2.9 | setSoLuong(int soLuong) | void | Kiểm tra tham số soLuong không được <= 0 |  |
| 2.10 | setLoaiSanPham(String loaiSanPham) | void | Kiểm tra tham số loaiSanPham không được rỗng |  |
| 2.11 | getMaSanPham() | String |  |  |
| 2.12 | getTenSanPham() | String |  |  |
| 2.13 | getDonVi() | String |  |  |
| 2.14 | getDonGia() | double |  |  |
| 2.15 | getNhaCungCap() | String |  |  |
| 2.16 | getHinhAnh | String |  |  |
| 2.17 | getMauSac | String |  |  |
| 2.18 | getKichThuoc | String |  |  |
| 2.19 | getLoaiSanPham | String |  |  |
| **3** | **Viết các constructor** |  |  |  |
| 3.1 | SanPham() |  |  |  |
| 3.2 | SanPham(maSanPham) |  |  |  |
| 3.3 | SanPham(maSanPham, tenSanPham, donVi, donGia, nhaCungCap, hinhAnh, mauSac, kichThuoc, loaiSanPham) |  |  |  |
| **4** | **Viết phương thức toString** | String |  | Trả về chuỗi chứa giá trị các thuộc tính của sản phẩm |

* 1. **Thực thể PhieuNhap:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi Chú** |
| 1. | **Khai báo thuộc tính** |  |  |  |
| 1.1 | maPhieuNhap | String | Mã phiếu nhập gồm 6 ký tự:  Hai ký tự đầu là PN  Bốn ký tự sau là số thứ tự từ 0001-1000 | Ví dụ:  “PN0001”, “PN0002” |
| 1.2 | ngayLap | LocalDate | Không được rỗng,  Không được sau ngày hiện tại |  |
| 1.3 | nguoiLap | NhanVien | Không được rỗng, không chứa ký tự đặc biệt. |  |
| 1.4 | nhaCungCap | String | Không được rỗng, không chứa ký tự đặc biệt. |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setMaPhieuNhap (String maPhieuNhap) | void | Phát sinh dãy số theo quy luật dựa trên ràng buộc đã đưa ra | Phát sinh tự động |
| 2.2 | setNgayLap(LocalDate ngayLap) | void | Kiểm tra tham số ngayLap có sau ngày hiện tại | throw exception “Ngày lập không được sau ngày hiện tại” nếu tham số ngayLap không hợp lệ |
| 2.3 | setNguoiLap(NhanVien nguoiLap) | void | Kiểm tra tham số nguoiLap không được rỗng |  |
| 2.4 | setNhaCungCap(String nhaCungCap) | void | Kiểm tra tham nhaCungCap không được trống | Throw exception “Nhà cung cấp không được để rỗng” khi tham số kích thước nhập vào không hợp lệ |
| 2.5 | getMaPhieuNhap | String |  |  |
| 2.6 | getNgayLap() | LocalDate |  |  |
| 2.7 | getNguoiLap() | NhanVien |  |  |
| 2.8 | getNhaCungCap() | double |  |  |
| **3** | **Viết các constructor** |  |  |  |
| 3.1 | PhieuNhap() |  |  |  |
| 3.2 | PhieuNhap(maPhieuNhap) |  |  |  |
| 3.3 | PhieuNhap(maPhieuNhap, ngayLap, nguoiLap, nhaCungCap) |  |  |  |
| **4** | **Viết phương thức toString** | String |  | Trả về chuỗi chứa giá trị các thuộc tính của phiếu nhập |

* 1. **Thực thể ChiTietPhieuNhap:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi Chú** |
| 1. | **Khai báo thuộc tính** |  |  |  |
| 1.1 | phieuNhap | PhieuNhap |  |  |
| 1.2 | sanPham | SanPham |  |  |
| 1.3 | soLuong | int | > 0 |  |
| **2** | **Viết các phương thức getter, setter** |  |  |  |
| 2.1 | setPhieuNhap (PhieuNhap phieuNhap) | void |  |  |
| 2.2 | setSanPham (SanPham sanPham) | void |  |  |
| 2.3 | setSoLuong(int soLuong) | void | Kiểm tra tham số soLuong không được <= 0 |  |
| 2.5 | getPhieuNhap | PhieuNhap |  |  |
| 2.6 | getSanPham() | SanPham |  |  |
| 2.7 | getSoLuong() | int |  |  |
| **3** | **Viết các constructor** |  |  |  |
| 3.1 | ChiTietPhieuNhap() |  |  |  |
| 3.2 | ChiTietPhieuNhap(PhieuNhap phieuNhap,SanPham sanPham, int soLuong) |  |  |  |
| **4** | **Viết phương thức toString** | String |  | Trả về chuỗi chứa giá trị các thuộc tính của chi tiết phiếu nhập |

* 1. **Thực thể ChiTietHoaDon:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | hoaDon | HoaDon |  |  |
| 1.2 | sanPham | SanPham |  |  |
| 1.3 | soLuong | int | >0 |  |
| 1.4 | donGia | double | >0 và đơn giá bán = đơn giá nhập \*120% |  |
| **2** | **Khai báo phương thức getter/setter :** |  |  |  |
| 2.1 | setHoaDon(HoaDon hoaDon) | void |  |  |
| 2.2 | setSanPham(SanPham sanPham) | void |  |  |
| 2.3 | setSoLuong (int soLuong) | void | Kiểm tra tham số soLuong không được <= 0 |  |
| 2.4 | setDonGia(double donGia) | void | Kiểm tra tham số donGia không được <= 0 |  |
| 2.5 | getHoaDon() | HoaDon |  |  |
| 2.6 | getSanPham() | SanPham |  |  |
| 2.7 | getSoLuong() | int |  |  |
| 2.8 | getDonGia() | double |  |  |
| **3** | **Viết các constructor, hashCode, equals** |  |  |  |
| 3.1 | ChiTietHoaDon() |  |  |  |
| 3.2 | ChiTietHoaDon(hoaDon, sanPham, soLuong, donGia) |  |  |  |
| **4** | **Viết phương thức toString()** | String |  | trả về chuỗi chứa giá trị các thuộc tính của chi tiết hóa đơn |

* 1. **Thực thể HoaDon:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maHoaDon | String | Mã phiếu nhập gồm 6 ký tự:  Hai ký tự đầu là HD  Bốn ký tự sau là số thứ tự từ 0001-1000 | Ví dụ:  “HD0001”,”HD0002” |
| 1.2 | ngayLap | LocalDate | Phải là ngày hiện tại |  |
| 1.3 | nguoiLap | NhanVien | Không được rỗng |  |
| 1.4 | khachHang | KhachHang | Không được rỗng |  |
| **2** | **Khai báo phương thức getter/setter :** |  |  |  |
| 2.1 | setmaHoaDon(String maHoaDon) | void | Phát sinh dãy số theo quy luật dựa trên ràng buộc đã đưa ra | Phát sinh tự động |
| 2.2 | setNgayLap(LocalDate ngayLap) | void | Kiểm tra tham số ngayLap là ngày hiện tại | throw exception “Ngày lập phải là ngày hiện tại” nếu tham số ngayLap không hợp lệ |
| 2.3 | setNguoiLap (NhanVien nhanVien) | void | Kiểm tra tham số nhanVien không được rỗng |  |
| 2.4 | setKhachHang(KhachHang khachHang) | void | Kiểm tra tham số khachHang không được rỗng |  |
| 2.5 | getMaHoaDon() | String |  |  |
| 2.6 | getNgayLap() | LocalDate |  |  |
| 2.7 | getNguoiLap() | NhanVien |  |  |
| 2.8 | getKhachHang() | KhachHang |  |  |
| **3** | **Viết các constructor, hashCode, equals** |  |  |  |
| 3.1 | HoaDon() |  |  |  |
| 3.2 | HoaDon(maHoaDon) |  |  |  |
| 3.2 | HoaDon(maHoaDon, ngayLap, nguoiLap, khachHang) |  |  |  |
| **4** | **Viết phương thức toString()** | String |  | trả về chuỗi chứa giá trị các thuộc tính của hóa đơn |

* 1. **Thực thể PhieuDatHang:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | maPhieuDatHang | String | Mã phiếu nhập gồm 6 ký tự:  Hai ký tự đầu là DH  Bốn ký tự sau là số thứ tự từ 0001-1000 | Ví dụ:  “DH0001”,”DH0002” |
| 1.2 | ngayLap | LocalDate | Phải là ngày hiện tại |  |
| 1.3 | nguoiLap | NhanVien | Không được rỗng |  |
| 1.4 | khachHang | KhachHang | Không được rỗng |  |
| **2** | **Khai báo phương thức getter/setter :** |  |  |  |
| 2.1 | setmaPhieuDatHang(String maPhieuDatHang) | void | Phát sinh dãy số theo quy luật dựa trên ràng buộc đã đưa ra | Phát sinh tự động |
| 2.2 | setNgayLap(LocalDate ngayLap) | void | Kiểm tra tham số ngayLap là ngày hiện tại | throw exception “Ngày lập phải là ngày hiện tại” nếu tham số ngayLap không hợp lệ |
| 2.3 | setNguoiLap (NhanVien nhanVien) | void | Kiểm tra tham số nhanVien không được rỗng |  |
| 2.4 | setKhachHang(KhachHang khachHang) | void | Kiểm tra tham số khachHang không được rỗng |  |
| 2.5 | getMaPhieuDatHang() | String |  |  |
| 2.6 | getNgayLap() | LocalDate |  |  |
| 2.7 | getNguoiLap() | NhanVien |  |  |
| 2.8 | getKhachHang() | KhachHang |  |  |
| **3** | **Viết các constructor, hashCode, equals** |  |  |  |
| 3.1 | PhieuDatHang() |  |  |  |
| 3.2 | PhieuDatHang(maPhieuDatHang) |  |  |  |
| 3.2 | PhieuDatHang(maPhieuDatHang, ngayLap nguoiLap, khachHang) |  |  |  |
| **4** | **Viết phương thức toString()** | String |  | trả về chuỗi chứa giá trị các thuộc tính của phiếu đặt hàng |

* 1. **Thực thể ChiTietPhieuDatHang:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Công việc** | **Kiểu dữ liệu** | **Ràng buộc** | **Ghi chú** |
| **1** | **Khai báo thuộc tính:** |  |  |  |
| 1.1 | phieuDatHang | PhieuDatHang |  |  |
| 1.2 | sanPham | SanPham |  |  |
| 1.3 | soLuong | int | >0 |  |
| 1.4 | donGia | double | >0 và đơn giá bán = đơn giá nhập \*120% |  |
| **2** | **Khai báo phương thức getter/setter :** |  |  |  |
| 2.1 | setPhieuDatHang(PhieuDatHang phieuDatHang) | void |  |  |
| 2.2 | setSanPham(SanPham sanPham) | void |  |  |
| 2.3 | setSoLuong (int soLuong) | void | Kiểm tra tham số soLuong không được <= 0 |  |
| 2.4 | setDonGia(double donGia) | void | Kiểm tra tham số donGia không được <= 0 |  |
| 2.5 | getPhieuDatHang() | PhieuDatHang |  |  |
| 2.6 | getSanPham() | SanPham |  |  |
| 2.7 | getSoLuong() | int |  |  |
| 2.8 | getDonGia() | double |  |  |
| **3** | **Viết các constructor, hashCode, equals** |  |  |  |
| 3.1 | ChiTietPhieuDatHang() |  |  |  |
| 3.2 | ChiTietPhieuDatHang (hoaDon, sanPham, soLuong, donGia) |  |  |  |
| **4** | **Viết phương thức toString()** | String |  | trả về chuỗi chứa giá trị các thuộc tính của chi tiết phiếu đạt hàng |