

LAB 4: LỚP VÀ ĐỐI TƯỢNG

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Xây dựng một lớp hoàn chỉnh
- ✓ Biết cách sử dụng các đặc tả truy xuất cho các thành viên bên trong một lớp

BÀI 1

Tạo lớp SanPham gồm 3 thuộc tính là tên, giá và giảm giá. Lớp cũng gồm 2 phương thức là tính thuế nhập khẩu (10% giá sản phẩm) và xuất thông tin ra màn hình. Thông tin xuất ra màn hình gồm:

- ✓ Tên sản phẩm:
- ✓ Đơn giá:
- ✓ Giảm giá:
- ✓ Thuế nhập khẩu:

HƯỚNG DẪN:

- ✓ Khai báo lớp theo mô hình sau

SanPham
+tenSp: String +donGia: double +giamGia: double
+getThueNhapKhau(): double +xuat(): void +nhap():void

BÀI 2

Viết chương trình tạo ra 2 sản phẩm thông tin được nhập từ bàn phím sau đó gọi phương thức xuất để xuất thông tin 2 đối tượng sản phẩm đã tạo

HƯỚNG DẪN:

- ✓ Tạo lớp chứa phương thức main()
- ✓ Trong phương thức main() tạo 2 đối tượng sp1 và sp2 từ lớp sản phẩm
- ✓ Gọi phương thức nhap() của 2 đối tượng sp1 và sp2 để nhập dữ liệu từ bàn phím
- ✓ Gọi phương thức xuất() của 2 đối tượng sp1 và sp2 để xuất thông tin của mỗi đối tượng ra màn hình

BÀI 3

Nâng cấp lớp SanPham bằng cách bổ sung public cho các đặt tả truy xuất cho phương thức xuất() và private cho getThueThuNhap(). Đồng thời bổ sung 2 hàm tạo, hàm tạo thứ nhất gồm 3 tham số là tên, giá và giảm giá, hàm tạo thứ 2 gồm 2 tham số là tên và giá (ngầm hiểu không giảm giá).

Viết chương trình tạo 2 sản phẩm có giảm giá và không giảm giá sau đó xuất thông tin 2 sản phẩm ra màn hình

HƯỚNG DẪN:

- ✓ Bổ sung 2 hàm tạo

```
public SanPham(String tenSp, double donGia, double giamGia){
    this.tenSp = tenSp;
    ...
}
public SanPham(String tenSp, double donGia){
    this(tenSp, donGia, 0)
}
```
- ✓ Sử dụng hàm tạo để tạo sản phẩm

```
SanPham sp1 = new SanPham(tenSp, donGia)
```

BÀI 4

Nâng cấp lớp SanPham bằng cách khai báo các trường dữ liệu với đặc tả truy xuất là private để hạn chế truy xuất trực tiếp đến các trường này sau đó bổ sung các phương thức getter và setter để đọc ghi dữ liệu các trường.

HƯỚNG DẪN:

- ✓ Cú mỗi trường dữ liệu được khai private bạn cần định nghĩa một cặp phương thức getter/setter để cho phép đọc ghi dữ liệu thông qua các phương thức này. Với cách làm này bạn dễ dàng nâng cấp lớp để bảo vệ hoặc thực hiện các tính toán cần thiết trước khi thực hiện các thao tác đọc ghi dữ liệu

```
public String getTenSp(){  
    return this.tenSp;  
}  
  
public void setTenSp(String tenSp){  
    this.tenSp = tenSp;  
}
```

BÀI 5

- 1) Tạo lớp HìnhTron có thuộc tính là bán kính, màu. Bổ sung hàm tạo, phương thức getter và setter để đọc ghi dữ liệu các trường, phương thức tính chu vi và diện tích của hình tròn.

HƯỚNG DẪN:

- ✓ Khai báo lớp theo mô hình sau

HìnhTron
- bankinh: double - mau : String
+ HìnhTron() + HìnhTron (bankinh: double, mau: String) + getBankKinh(): double + getMau(): String + setBankKinh(bankinh: double) : void + setMau(mau: String) + tinhDienTich() : double + tinhChuVi() : double

- 2) Viết chương trình tạo 2 hình tròn, sau đó xuất thông tin diện tích và chu vi của 2 hình tròn ra màn hình.

BÀI 6

- 1) Tạo lớp TaiKhoan có thuộc tính là số tài khoản (soTK), tên tài khoản (tenTK), số dư (soDu) . Bổ sung hàm tạo, phương thức getter và setter để đọc ghi dữ liệu các trường, phương thức rút tiền, gửi tiền và chuyển khoản.

HƯỚNG DẪN:

- ✓ Khai báo lớp theo mô hình sau

TaiKhoan
- soTaiKhoan: String - tenTaiKhoan : String - soDu : int
+ TaiKhoan() + TaiKhoan(soTaiKhoan: String, tenTaiKhoan : String, soDu : int) + getSoTaiKhoan(): String + getTenTaiKhoan(): String + getsoDu(): int + setSoTaiKhoan(soTaiKhoan: String) : void + setTenTaiKhoan(tenTaiKhoan: String) : void + rutTien(soTien: int) : void + guiTien(soTien: int) : void + chuyenKhoan (other : TaiKhoan, soTien: int) : void

- 2) Viết chương trình khởi tạo 2 tài khoản, sau đó thực hiện các thao tác rút tiền, gửi tiền và chuyển khoản trên 2 tài khoản vừa tạo. Xuất thông tin 2 tài khoản ra màn hình.