

BÀI TẬP LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG JAVA

Phần 3: Kế Thừa – Đa Hình – Trừu Tượng – Giao Diện

19) Công ty du lịch V quản lý thông tin là các **chuyến xe**. Thông tin của 2 loại chuyến xe:

- Chuyến xe **nội thành**: **Mã số chuyến**, **Họ tên tài xế**, **số xe**, số tuyến, số km đi được, **doanh thu**.
- Chuyến xe **ngoại thành**: **Mã số chuyến**, **Họ tên tài xế**, **số xe**, nơi đến, số ngày đi được, **doanh thu**.

Thực hiện các yêu cầu sau:

- ✓ Cài đặt các lớp với chức năng **kế thừa**.
- ✓ Viết chương trình quản lý các chuyến xe theo dạng cây thừa kế với các phương thức sau:
 - **Nhập** chuyến xe nội thành
 - Nhập chuyến xe ngoại thành
 - **Xuất** chuyến xe nội thành
 - Xuất chuyến xe ngoại thành
 - Xuất **tất cả** chuyến xe
 - Tính **tổng** doanh thu cho **từng loại xe**.
 - **Sắp xếp** các **chuyến xe tăng dần** theo **doanh thu**.
 - Sắp xếp các **chuyến xe ngoại thành giảm dần** theo **doanh thu**.
 - **Xóa xe nội thành** có doanh thu ít nhất.
 - Sắp xếp các **chuyến xe nội thành giảm dần** theo **họ tên tài xế**.

20) Thư viện X quản lý danh sách các loại **sách**. Thông tin về các loại sách:

- **Sách giáo khoa**: **Mã sách**, **ngày nhập (ngày, tháng, năm)**, **đơn giá**, **số lượng**, **nhà xuất bản**, tình trạng (mới, cũ).
 - ✓ Nếu tình trạng sách là mới thì: **thành tiền** = số lượng * đơn giá.
 - ✓ Nếu tình trạng sách là cũ thì: **thành tiền** = số lượng * đơn giá * 50%
- **Sách tham khảo**: **Mã sách**, **ngày nhập (ngày, tháng, năm)**, **đơn giá**, **số lượng**, **nhà xuất bản**, thuế. **Thành tiền** = số lượng * đơn giá + thuế

Thực hiện các yêu cầu sau:

- Cài đặt các lớp với chức năng thừa kế.
- Nhập danh sách các loại sách.
- Xuất danh sách các loại sách.
- Tính tổng thành tiền cho từng loại.
- **Tính trung bình cộng** đơn giá của các sách tham khảo.
- Xuất ra các sách giáo khoa của nhà xuất bản X.

21) **Nhân viên** trong một cơ quan được lĩnh lương theo các dạng khác nhau:

- Dạng người lao động hưởng lương tư ngân sách nhà nước gọi là cán bộ, công chức (**dạng biên chế**).
- Dạng người lao động lĩnh lương tư ngân sách của cơ quan gọi là người làm **hợp đồng**.

Như vậy, hệ thống có hai đối tượng: **Biên chế** và **Hợp đồng**. Hai loại đối tượng này có đặc tính chung là viên chức làm việc cho cơ quan.

Từ đây có thể tạo nên lớp cơ sở để quản lý một viên chức (**lớp Người**) bao gồm **mã số**, **họ tên**, **lương**.

Hai lớp kế thừa từ lớp cơ sở trên:

- + Lớp **BienChe** gồm các thuộc tính: **hệ số lương**, **lương cơ bản** (= 500.000) **tiền phụ cấp chức vụ**.
- + Lớp **HopDong** gồm các thuộc tính: **tiền công lao động**, **số ngày làm việc trong tháng**, **hệ số vượt giờ** (hệ số vượt giờ = **số giờ vượt** * 50.000)

Biết rằng:

- + *Lương của nhân viên biên chế = hệ số lương * lương cơ bản + tiền phụ cấp chức vụ*
- + *Lương của nhân viên hợp đồng = tiền công lao động * số ngày làm việc + hệ số vượt giờ*

Thực hiện các yêu cầu sau:

- Hãy thiết kế lớp trên
- Viết chương trình minh họa.
- Tìm** nhân viên biên chế có lương cao nhất
- Tìm nhân viên hợp đồng có lương cao nhất

22) Xây dựng chương trình quản lý danh sách hoá đơn tiền điện của **khách hàng**. Thông tin bao gồm các loại khách hàng:

- **Khách hàng Việt Nam**: **mã khách hàng**, **họ tên**, **ngày ra hoá đơn** (ngày, tháng, năm), **đối tượng khách hàng** (sinh hoạt, kinh doanh, sản xuất): **số lượng** (số KW tiêu thụ), **đơn giá**, **định mức**. **Thành tiền** được tính như sau:
 - ✓ Nếu số lượng <= định mức thì: thành tiền = số lượng * đơn giá.
 - ✓ Ngược lại thì: thành tiền = số lượng * đơn giá * định mức + số lượng KW vượt định mức * Đơn giá * 2.5.
- **Khách hàng nước ngoài**: **mã khách hàng**, **họ tên**, **ngày ra hoá đơn** (ngày, tháng, năm theo định dạng dd/mm/yyyy), **quốc tịch**, **số lượng**, **đơn giá**. **Thành tiền** được tính = số lượng * đơn giá.

Thực hiện các yêu cầu sau:

- Xây dựng các lớp với chức năng **thừa kế** thể hiện tính **đa hình**.
- Nhập xuất danh sách các hóa đơn khách hàng.
- Tính tổng số lượng cho từng loại khách hàng.
- Tính trung bình thành tiền của khách hàng người nước ngoài.
- Xuất ra các hoá đơn trong tháng 09 năm 2013 (của cả 2 loại khách hàng).
- Xóa khách hàng ra khỏi danh sách

23) Sử dụng NNLT Java xây dựng chương trình sau:

- Tạo **lớp sinh viên** (Sinhvien) gồm các thuộc tính sau: **mã sinh viên** (MaSV) kiểu số nguyên, **tên sinh viên** (TenSV) kiểu chuỗi ký tự. Các phương thức:
 - + **setSV()**: Đưa thông tin vào lớp sinh viên

- + viewSV(): Đưa thông tin sinh viên ra màn hình
- Tạo **lớp sinh viên CNTT**(SinhvienCNTT) kế thừa từ lớp sinh viên và có thêm các thuộc tính sau: **Điểm toán (dtoan)**, **Điểm tin (dtin)**. Các phương thức:
 - + setSVCNTT(): Đưa thông tin vào lớp sinh viên CNTT
 - + viewSVCNTT(): Đưa thông tin sinh viên CNTT ra màn hình
 - + tongdiem(): Tính tổng điểm = điểm toán + điểm tin

Thực hiện yêu cầu sau:

- a) Xây dựng các lớp với chức năng **thừa kế** thể hiện tính **đa hình**.
- b) Tạo 1 danh sách sinh viên CNTT gồm n sinh viên ($1 \leq n \leq 5$)
 - + Đưa ra màn hình các sinh viên có tổng điểm > 10
 - + Đưa ra màn hình các sinh viên có tổng điểm < 5
 - + Cho biết sinh viên có điểm toán lớn nhất.
 - + Sắp xếp sinh viên theo mã sinh viên

24) Các thí sinh dự thi đại học bao gồm các thí sinh thi khối A, thí sinh thi khối B, thí sinh thi khối C

- + Các thí sinh cần quản lý các thuộc tính: Số báo danh, họ tên, địa chỉ, ưu tiên.
 - + Thí sinh thi khối A thi các môn: Toán, lý, hoá
 - + Thí sinh thi khối B thi các môn: Toán, Hoá, Sinh
 - + Thí sinh thi khối C thi các môn: văn, Sử, Địa
- a) Xây dựng các lớp để quản lý các thí sinh sao cho sử dụng lại được nhiều nhất.
 - b) Xây dựng lớp TuyenSinh cài đặt các phương thức thực hiện các nhiệm vụ sau:
 - Nhập thông tin về các thí sinh dự thi
 - Hiển thị thông tin về một thí sinh
 - Tìm kiếm theo số báo danh
 - Kết thúc chương trình.

25) Một đơn vị sản xuất gồm có các **nhân viên (NV)** gồm nhân viên văn phòng, nhân viên kinh doanh, công nhân

- Mỗi nhân viên cần quản lý: **họ tên, địa chỉ, giới tính, năm sinh, số năm công tác, lương cơ bản**, biết rằng lương cơ bản là 9.500.000đ. Trong đó **lương** của từng loại nhân viên được tính như sau:

+ Nhân viên Văn phòng (**NVVP**)

lương = **hệ số lương** * LCB + **phụ cấp**

(phụ cấp = 1*LCB nếu số năm công tác <10; phụ cấp = 1.2*LCB nếu số năm công tác >=10)

+ Nhân viên Kinh doanh (**NVKD**)

lương = LCB + **thưởng**

(thưởng = **doanh thu** * **phần trăm hoa hồng**)

+ Công nhân (**CN**)

lương = **hệ số lương** * LCB + **thưởng**

(thưởng = 1*LCB nếu **sản lượng** <1000; thưởng = 1.5*LCB nếu sản lượng >=1000)

- Xây dựng lớp QLNV thể hiện tính đa hình cài đặt các phương thức thực hiện các chức năng sau:

- + Nhập vào một dãy gồm n nhân viên (n - nhập từ bàn phím).
- + Nhập thông tin mới cho các loại nhân viên
- + Xuất thông tin
- + Tìm kiếm theo họ tên
- + Hiện thị ra tất cả những nhân viên sinh năm 1975
- + Cho biết có bao nhiêu nhân viên sinh năm 1975 và quê ở Thái Nguyên
- + Hiện thị ra tất cả những nhân viên có tổng lương dưới 5.000.000.

26) Giả sử cần xây dựng chương trình quản lý dùng cho một học viện nghiên cứu giảng dạy và ứng dụng. Đối tượng quản lý bao gồm các sinh viên đang theo học, các nhân viên đang làm việc tại học viện, các khách hàng đến giao dịch mua bán sản phẩm ứng dụng. Dựa vào một số đặc tính của từng đối tượng, người quản lý cần đưa ra cách thức đánh giá khác nhau.

Vậy hãy xây dựng các lớp sau:

- Lớp **Person**: bao gồm các thuộc tính họ tên, địa chỉ, phương thức toString.

- Các lớp **Student**, **Employee**, **Customer** (mô tả dưới đây) thừa kế lớp **Person**.

- o Lớp **Student**: bao gồm các thuộc tính điểm môn học 1, điểm môn học 2, và các phương thức: tính điểm TB, **overriding** phương thức toString trả về bảng điểm sinh viên (gồm thông tin thuộc tính và điểm TB).
- o Lớp **Employee**: bao gồm thuộc tính *heSoLuong*, và các phương thức: tính lương, **overriding** phương thức toString trả về bảng lương cho nhân viên (gồm thông tin thuộc tính đối tượng và tiền lương).
- o Lớp **Customer**: bao gồm thuộc tính tên công ty, trị giá hóa đơn và phương thức toString trả về thông tin hóa đơn cho khách hàng (gồm các thuộc tính của đối tượng).

- Lớp có 1 *biến* danh sách để lưu các sinh viên, nhân viên, khách hàng (dùng 1 biến array Person), biến lưu tổng số người có trong danh sách, constructor mặc định khởi tạo array với dung lượng cho trước, phương thức thêm một người vào danh sách (thông số Person), xóa 1 người khỏi danh sách (nhận thông số là họ tên của người cần xóa), sắp xếp danh sách theo thứ tự họ tên, phương thức xuất danh sách. Khi danh sách đầy thì tự động tăng dung lượng dãy lên 50%.

- Viết lớp với phương thức main cho phần kiểm nghiệm. Giao tiếp với người dùng bằng menu (*thể hiện tính đa hình – polymorphism bằng cách cho phép lựa chọn nhập thông tin là sinh viên, nhân viên hay khách hàng*).

27) **Hàng hóa** quản lý trong kho của một siêu thị gồm có hàng thực phẩm, hàng sành sứ và hàng điện máy.

- Mỗi loại hàng đều có **mã hàng** (**không được để trống**), **tên hàng** (**không được rỗng**), **số lượng tồn** (≥ 0), **đơn giá** (> 0).
- **Hàng thực phẩm** thì cần quan tâm đến thông tin **ngày sản xuất**, **ngày hết hạn** (**ngày hết hạn phải sau hoặc là ngày sản xuất**) và **nhà cung cấp**.

- **Hàng điện máy** cần biết **thời gian bảo hành** bao nhiêu tháng (≥ 0), **công suất** bao nhiêu KW (> 0).
- **Hàng sành sứ** thì cần biết thông tin về **nhà sản xuất** và **ngày nhập kho**.

Ngoài ra, người quản lý cần quan tâm đến **số lượng tồn kho** và các yếu tố khác của từng loại hàng hóa để **đánh giá mức độ bán buôn**, **tiền VAT từng loại hàng hóa**. Biết rằng VAT của hàng điện máy và sành sứ là 10%, VAT của hàng thực phẩm là 5%.

a) Dựa vào các thông tin trên, hãy xác định:

- Các lớp có thể có. Lớp nào là lớp trừu tượng (*abstract class*), lớp nào là lớp cụ thể.
- Các thuộc tính cho từng lớp.
- Các phương thức cho từng lớp (phương thức nào là phương thức trừu tượng (*abstract method*), danh sách các tham số có thể có cho từng phương thức và kiểu trả về của phương thức).
- Thiết kế mô hình class (*xây dựng cây thừa kế, các giao diện nếu có*).

b) Dùng java IDE, tạo một project. Thực hiện cài đặt tường minh cho mỗi loại hàng cụ thể trên. Trong đó, để đánh giá mức độ bán buôn thì:

- Hàng điện máy, nếu số lượng tồn kho < 3 thì được đánh giá là bán được.
 - Hàng thực phẩm, nếu vẫn còn tồn kho và bị hết hạn thì đánh giá là khó bán.
 - Hàng sành sứ, nếu số lượng tồn kho > 50 và thời gian lưu kho > 10 ngày thì đánh giá là bán chậm.
 - Các trường hợp còn lại xem như không đánh giá.
- Hãy viết lớp quản lý danh sách hàng hóa. Dùng Array để lưu trữ danh sách hàng hóa.
- Tạo constructor khởi tạo danh sách.
 - Viết phương thức thêm một hàng hóa vào danh sách (*thêm thành công nếu không bị trùng mã hàng, thể hiện tính đa hình – polymorphism bằng cách cho phép lựa chọn nhập thông tin*)
 - Viết phương thức in toàn bộ danh sách các hàng hóa.
- Tạo lớp cho phần thử nghiệm, với menu lựa chọn để thực hiện các chức năng theo yêu cầu.

28)

a) Xây dựng lớp trừu tượng “Hình” như sau:

```
abstract class Hình
{
    public abstract float getArea();
}
```

b) Xây dựng lớp “HìnhVuong”, “HìnhTron”, “HìnhTamGiac”, “HìnhBinhHanh”, “HìnhThoi” bằng cách kế từ lớp Hình và ghi đè phương thức getArea để tính diện tích của các hình tương ứng.

Hướng dẫn

Bài toán gồm có 6 đối tượng: class Hình, class HìnhVuong, class HìnhTron, class HìnhBinhHanh, class Hìnhthoi

Các thuộc tính và phương thức của các lớp:

abstract class Hinh

```
{  
    public abstract float getArea();  
}
```

class Hinhvuong extends Hinh

```
{  
    float canh;  
    public Hinhvuong()  
    public Hinhvuong(float a)  
    public float getArea()  
    public void nhap()  
    public void hien()  
}
```

class Hinhtron extends Hinh

```
{  
    private float r;  
    public Hinhtron()  
    public Hinhtron(float r)  
    public float getArea ()  
}
```

class Hinh tamgiac extends Hinh

```
{  
    private float cao, day  
    public Hinh tamgiac()  
    public Hinh tamgiac(float cao, float day)  
    public float getArea()  
}
```

class Hinhvuong extends Hinh

```
{  
    float canh;  
    public Hinhvuong()  
    public Hinhvuong(float a)  
    public float getArea()  
    public void nhap()  
    public void hien()  
}
```

class Hinhtron extends Hinh

```
{  
    private float r;  
    public Hinhtron()  
    public Hinhtron(float r)  
    public float getArea ()  
}
```

```

class HinhTamgiac extends Hinh
{
    private float cao, day
    public HinhTamgiac()
    public HinhTamgiac(float cao, float day)
    public float getArea()
}

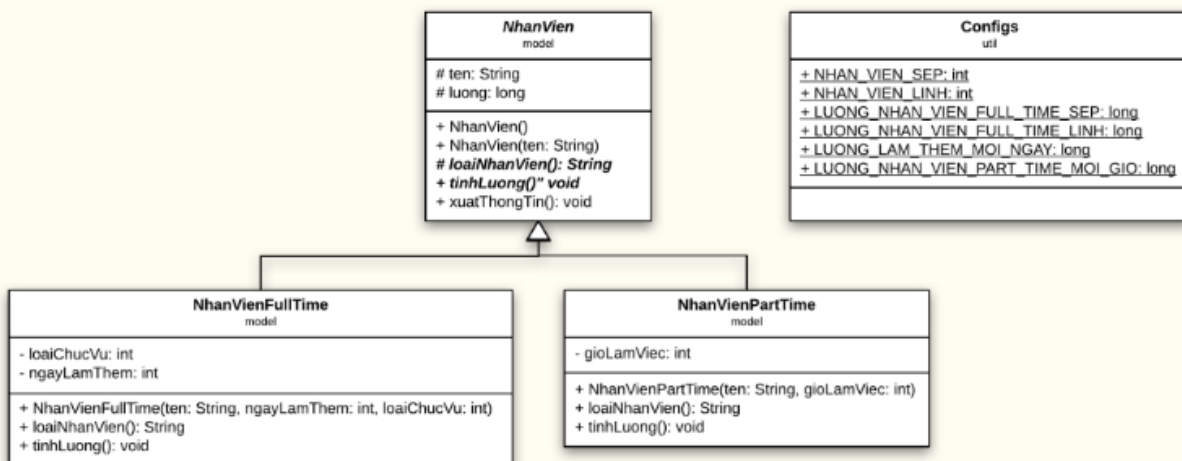
```

- 29) Công ty có hai loại nhân viên: nhân viên toàn thời gian và nhân viên thời vụ.
- Nhân viên toàn thời gian **là lính** sẽ hưởng lương 10 triệu / tháng. Nhân viên toàn thời gian **là sếp** sẽ hưởng lương 20 triệu / tháng.
 - Nhân viên toàn thời gian nếu làm thêm ngày nào thì sẽ được cộng thêm 800 ngàn mỗi ngày, bất kể chức vụ.
 - Nhân viên thời vụ cứ làm mỗi giờ được 100 ngàn, không phân biệt chức vụ gì cả. Làm nhiều thì hưởng nhiều.

Ứng dụng sẽ cho phép người dùng nhập vào số lượng nhân viên. Sau đó với từng nhân viên, người dùng phải nhập vào **tên nhân viên**, **loại nhân viên** toàn thời gian hay bán thời gian, **nhân viên toàn thời gian** thì là nhân viên lính hay nhân viên sếp, **có làm thêm ngày nào không**, **nhân viên thời vụ** thì làm được mấy giờ.

Cuối cùng dựa vào các thông tin đó, sẽ xuất ra màn hình lương tương ứng cho tất cả nhân viên.

Cài đặt các lớp thể hiện tính **trừu tượng** và **kế thừa**.



Lớp **NhanVien** khi này là lớp *Trừu tượng*. Và đây là code cho lớp **NhanVien**, bạn có thể so sánh với hình ảnh code trên kia của lớp này để xem sự thay đổi.

```
1 package model;
2
3 public abstract class NhanVien {
4
5     protected String ten;
6     protected long luong;
7
8     public NhanVien() {
9     }
10
11     public NhanVien(String ten) {
12         this.ten = ten;
13     }
14
15     // Lớp con phải override để lo vụ loại nhân viên này
16     protected abstract String loaiNhanVien();
17
18     // Lớp con phải override để lo vụ tính lương này
19     public abstract void tinhLuong();
20
21     public void xuấtThôngTin() {
22         System.out.println("==== Nhân viên: " + ten + " =====");
23         System.out.println("- Loại nhân viên: " + loaiNhanVien());
24         System.out.println("- Lương: " + luong + " VND");
25     }
26 }
```

<https://yellowcodebooks.com/2017/09/26/java-bai-32-tinh-truu-tuong-abstraction/>

30) Xây dựng giao diện nhân viên bao gồm hai phương thức: Nhập và hiển thị thông tin của cán bộ.

Ta có thể xây dựng thêm một giao diện nhân viên mới mở rộng từ giao diện nhân viên. Giao diện nhân viên mới này xây phương thức cách tính lương cho mỗi loại nhân viên.

Hãy xây dựng lớp nhân viên sản xuất thực thi hai giao diện trên. Biết cách tính lương nhân viên như sau: $\text{Lương} = \text{Hệ số lương} * \text{Lương cơ bản}$

31) Công ty Yamaha Việt Nam là một công ty chuyên sản xuất xe máy với các model khác nhau như Sirius, Exciter và cung cấp cho thị trường Việt Nam. Công ty có nhu cầu làm một phần mềm quản lý hệ thống phân phối các loại xe trên. Bạn là một thành viên trong nhóm phát triển phần mềm này và bạn được giao phải thực hiện các công việc sau:

- Tạo interface có tên IMotor chứa các phương thức
 - void InputInfo();
 - void DisplayInfo();
- Tạo lớp MyMotor cài đặt IMotor và có các thuộc tính sau
 - public String code; //Tối thiểu 4 ký tự và không trùng

- public string name; //Tối thiểu 6 ký tự
- public int capacity; //Capacity >= 50
- public string type; //Xe số hoặc xe ga
- Tạo lớp Exciter kế thừa từ lớp MyMotor. Trong lớp Exciter bổ sung thêm thuộc tính warranty (thời gian bảo hành) có kiểu int và giá trị >= 1 (đơn vị năm)
- Tạo lớp Sirius kế thừa từ lớp MyMotor. Trong lớp Sirius bổ sung thêm thuộc tính color có kiểu string (màu xe) và giá trị của nó hoặc là xanh, vàng, trắng.
- Tạo lớp thực thi Yamaha cho phép người dùng lựa chọn chức năng
 - Nhập thông tin xe exciter
 - Nhập thông tin xe sirius
 - Hiển thị thông tin xe exciter
 - Hiển thị thông tin xe sirius
 - Tìm kiếm xe exciter
 - Tìm kiếm xe sirius
 - Thoát chương trình
 - Lựa chọn của bạn: