

Bài 4: THỰC HÀNH LẬP TRÌNH JAVA HƯỚNG ĐỐI TƯỢNG

1. Hướng dẫn thực hiện bài 4: Bài toán quản lý đăng ký xe.1
2. Hướng dẫn thực hiện bài 5: Xác định và xây dựng lớp - mối quan hệ lớp4
3. Hướng dẫn thực hiện bài 10: Quản lý nhân viên.....8

1. Hướng dẫn thực hiện bài 4: Bài toán quản lý đăng ký xe.

1.1. Đề bài:

Sở giao thông cần theo dõi việc đăng ký xe của người dân. Dựa vào thông tin trị giá xe và dung tích xylanh của xe, sở giao thông cũng tính mức thuế phải đóng trước bạ khi mua xe như sau:

- i) Dưới 100cc, 1% trị giá xe.
- ii) Từ 100 đến 200cc, 3% trị giá xe.
- iii) Trên 200cc, 5% trị giá xe.

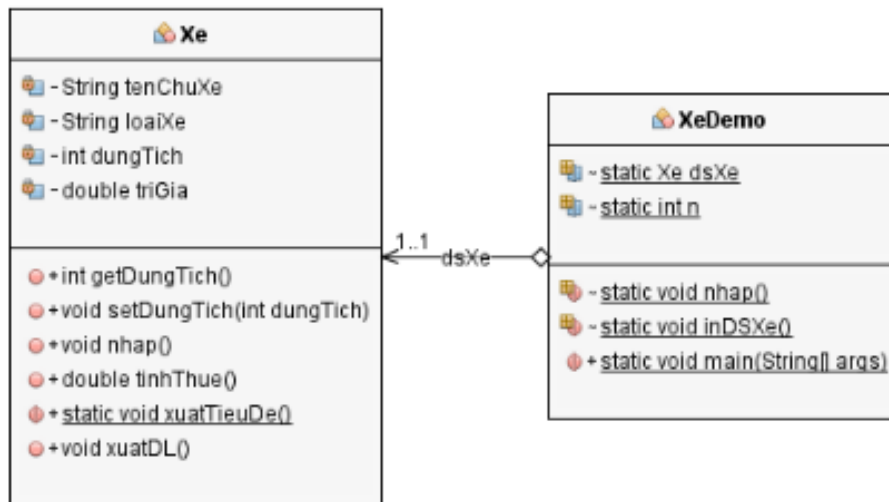
1.2. Yêu cầu bài toán:

1. Thiết kế và cài đặt lớp phương tiện với các thuộc tính và phương thức phù hợp bảo đảm tính đóng gói theo hướng đối tượng.
2. Xây dựng lớp chứa hàm main. Hàm main in ra menu lựa chọn các công việc:
 - Nhập thông tin và tạo danh sách các xe
 - Xuất bảng kê khai tiền thuế trước bạ của các xe.
 - Thoát.

Mẫu thiết kế xuất của chương trình:

Tên chủ xe	Loại xe	Dung tích	Trị giá	Thuế phải nộp
Nguyễn Thu Loan	Future Neo	180	35000000.00	1850000.00
Lê Minh Tính	Ford Ranger	3000	250000000.00	12500000.00
Nguyễn Minh Triết	Landscape	1500	100000000.00	5000000.00

1.3. Mô hình hóa bài toán.



1.4. Hướng dẫn cài đặt.

Xe.java

<pre>public class Xe {</pre>	
<pre>//các thuộc tính: private String tenChuXe; private String loaiXe; private int dungTich; private double triGia;</pre>	Đóng gói và bảo mật thông tin, nên khai báo các thuộc tính có phạm vi private
<pre>//Xây dựng các phương thức get/set cần thiết public int getDungTich() { return dungTich; } public void setDungTich(int dungTich) { this.dungTich = dungTich; }</pre>	
<pre>//Xây dựng phương thức nhập dữ liệu public void nhap() { Scanner s=new Scanner(System.in); System.out.print("nhap ten chur xe, loai xe:"); tenChuXe=s.nextLine(); loaiXe=s.nextLine(); System.out.println("nhap dung tich, gia="); dungTich=s.nextInt(); triGia=s.nextDouble(); }</pre>	Dùng Scanner để nhập dữ liệu minh họa



<pre>//Xây dựng phương thức tính thuế public double tinhThue(){ double thue=0; if (dungTich<100)thue=triGia/100; else if (dungTich<200) thue=triGia*3/100; else thue=triGia*5/100; return thue; }</pre>	<p>Sử dụng cấu trúc if để kiểm tra điều kiện tính thuế</p>
<pre>//Xây dựng phương thức in tiêu đề public static void xuatTieuDe(){ // in xong tiêu đề xuống dòng System.out.printf("%-15s %-15s %15s %15s %15s %n", "Ten chu xe","loai Xe","Dung tic","tri gia","Thue phai nop"); }</pre>	<p>Sử dụng hàm printf để định khoảng cách tiêu đề</p>
<pre>//Xây dựng phương thức in dữ liệu public void xuatDL(){ System.out.printf("%-15s %-15s %15d %15.1f %15.1f %n",tenChuXe, loaiXe,dungTich, triGia, tinhThue()); }</pre>	<p>Sử dụng hàm printf với mẫu hàm và khoảng cách tương tự để đóng dữ liệu</p>
<pre>}//kết thúc lớp</pre>	

Xây dựng lớp chứa hàm main minh họa lớp vừa xây dựng. XeDemo.java

<pre>import java.util.Scanner; public class XeDemo {</pre>	<p>Khai báo thư viện và lớp Xedemo.</p>
<pre>//khai báo mảng đối tượng Xe static Xe dsXe[]; static int n;//số lượng xe</pre>	<p>Sử dụng phạm vi static vì muốn minh họa</p>
<pre>//Xây dựng hàm nhập mảng static void nhap(){ System.out.print("nhập số lượng xe"); Scanner s=new Scanner(System.in); n=s.nextInt(); s.nextLine(); //xin cấp phát mảng dsXe=new Xe[n]; for (int i=0;i<n;i++){ //xin cấp phát cho phần tử mảng thứ i</pre>	<p>tách các nhiệm vụ thực hiện trong hàm main theo kỹ thuật lập trình</p>

<pre> dsXe[i]=new Xe(); dsXe[i].nhap(); } }</pre>	
<pre>//Xây dựng hàm in danh sách xe static void inDSXe(){ //in danh sách Xe.xuatTieuDe(); for (Xe x:dsXe) x.xuatDL(); }</pre>	Gọi in tiêu đề chung cho các xe 1 lần thông qua hàm static xuấtTieuDe() đã xây dựng
<pre>//Xây dựng hàm main public static void main(String[] args) { int chon=0; Scanner s=new Scanner(System.in); do { System.out.println("1.Nhap thông tin xe"); System.out.println("2.Xuất bảng kê thuê trước bạn"); System.out.println("3.Thoát"); System.out.println("Lựa chọn"); chon=s.nextInt(); switch (chon){ case 1: nhap(); break; case 2: inDSXe(); break; case 3: System.exit(0); } } while (chon !=3); }</pre>	
<pre>}//kết thúc lớp XeDemo.</pre>	

2. Hướng dẫn thực hiện bài 5: Xác định và xây dựng lớp - mối quan hệ lớp

2.1. Đề bài:

Thông tin nhân viên trong công ty được quản lý thông qua họ đệm, tên nhân viên, ngày sinh và ngày vào làm. Thông tin về ngày tháng được mô tả thông qua ngày, tháng,

năm. Yêu cầu tháng phải nằm trong khoảng từ 1-12 và ngày phải phù hợp với tháng tương ứng.

2.2. Yêu cầu

Hãy mô tả sơ đồ các lớp tìm được minh họa nhập 1 nhân viên và in ra nhân viên đó sau khi nhập. Mở rộng: Nếu nhân viên khi nhập là một danh sách thông tin mảng sau khi nhập có mẫu như sau thì sơ đồ lớp có thay đổi gì không?.

DS nhan vien sau khi nhap:			
Ho dem	Ten	Ngày sinh	Ngày vào làm
vu van h	ũ	7/8/8	6/7/8
uy	uj	8/8/9	7/7/7
98	98	9/9/9	8/8/8

2.3. Gợi ý cài đặt:

NgàyThang.java

```
import java.util.Scanner;

public class NgàyThang {
    private int thang; // 1-12
    private int ngay; // 1-31 based on month
    private int nam; // any year
    //ghi đè phương thức toString để mô tả
    tháng/ngày/năm
    public String toString() {
        return thang + "/" + ngay + "/" + nam;
    }
    public void nhap() {
        System.out.print("nhap ngay, thang, nam");
        Scanner s=new Scanner(System.in);
        int n=s.nextInt();
        int t=s.nextInt();
        int nam=s.nextInt();
        kiemTra(n,t,nam);
    }
    public void kiemTra( int ngay1,int thang1 , int nam1
) {
        if ( thang1 > 0 && thang1 <= 12 ) // tháng hợp lệ
            this.thang = thang1;
        else {
            this.thang = 1;
            System.out.println( "Tháng " + thang1 + " không
hợp lệ. đặt lại tháng =1." );
        }
        nam = nam1; // nếu năm hợp lệ
        ngay = checkDay( ngay1 ); // kiểm tra ngày hợp lệ
```

```
        System.out.println( "in thông tin ngày tháng " +
toString() );
    }
private int checkDay( int ngayKiemTra ) {
    int
ngayTrongThang[]={0,31,28,31,30,31,30,31,31,30,31,30,31};
    // kiểm tra tháng nếu trong khoảng từ 1-12
    if (ngayKiemTra > 0 && ngayKiemTra <= ngayTrongThang [
thang ] )

        return ngayKiemTra;
    // kiểm tra năm nhuận
    else
    if (((thang==2) && (ngayKiemTra == 29)) &&
        ((nam % 400 == 0) || ((nam % 4 == 0) && (nam % 100
!= 0 ))))
        return ngayKiemTra;
    else {
        System.out.println( "ngày " + ngayKiemTra + " không
hợp lệ, đặt lại =1." );
        return 1;
    }
} //end of checkDay()
```

Xây dựng lớp nhân viên có quan hệ kết tập với lớp Ngày tháng- NhanVien.java

```
//khai báo thư viện và lớp
import java.util.Scanner;
public class NhanVien {
    private String hoDem;
    private String ten;
    private NgayThang ngaySinh=new NgayThang();
    private NgayThang ngayVaoLam=new NgayThang();

    //xây dựng phương thức nhập thông tin nhân viên
    void nhap(){
        Scanner s=new Scanner(System.in);
        System.out.print(" ho dem=");
        hoDem=s.nextLine();
        System.out.print("nhap ten nhan vien=");
        ten=s.nextLine();
        System.out.println("nhap ngay, tháng, năm sinh=:");
        ngaySinh.nhap();
        System.out.println("nhap ngay, tháng, năm vào lam=:");
        ngayVaoLam.nhap();
    }
}
```

```
//xuất tiêu đề
static void xuatTieuDe(){
    System.out.printf("%-15s%-10s%15s%15s\n",
        "Ho dem", "Ten", "Ngày sinh", "Ngày vào làm");
}

//in dữ liệu
public void xuat(){
    System.out.printf("%-15s%-10s%15s%15s\n",
        hoDem, ten, ngaySinh, ngayVaoLam);
}

} //kết thúc xây dựng lớp Nhanvien
```

NhanVienDemo.java

```
public class NhanVienDemo {
    static NhanVien dsnv[]; // = new NhanVien();
    static int n; // số phần tử mảng
    static void nhapDSNV() {
        Scanner s = new Scanner(System.in);
        System.out.println("nhập số phần tử mảng n=");
        n = s.nextInt();
        // xin caps phát n phần tử cho mảng
        dsnv = new NhanVien[n];
        // nhập mảng
        for (int i = 0; i < n; i++) {
            System.out.print("nhập nhân viên thứ " + (i + 1) + ": ");
            dsnv[i] = new NhanVien();
            dsnv[i].nhap();
        }
    }
    static void inDSNV() {
        System.out.println("DS nhân viên sau khi nhập:");
        // gọi hàm in tiêu đề
        NhanVien.xuatTieuDe();
        for (NhanVien nv : dsnv) {
            nv.xuat();
        }
    }
    public static void main(String[] args) {
        nhapDSNV();
        inDSNV();
    }
}
```

3. Hướng dẫn thực hiện bài 10: Quản lý nhân viên.

3.1. Đề bài:

Một công ty sản xuất có 2 loại nhân viên là nhân viên sản xuất và nhân viên văn phòng. Hai loại nhân viên được quản lý thông qua các thông tin: họ tên; năm vào làm, lương, phụ cấp. Lương của hai loại nhân viên được tính như sau:

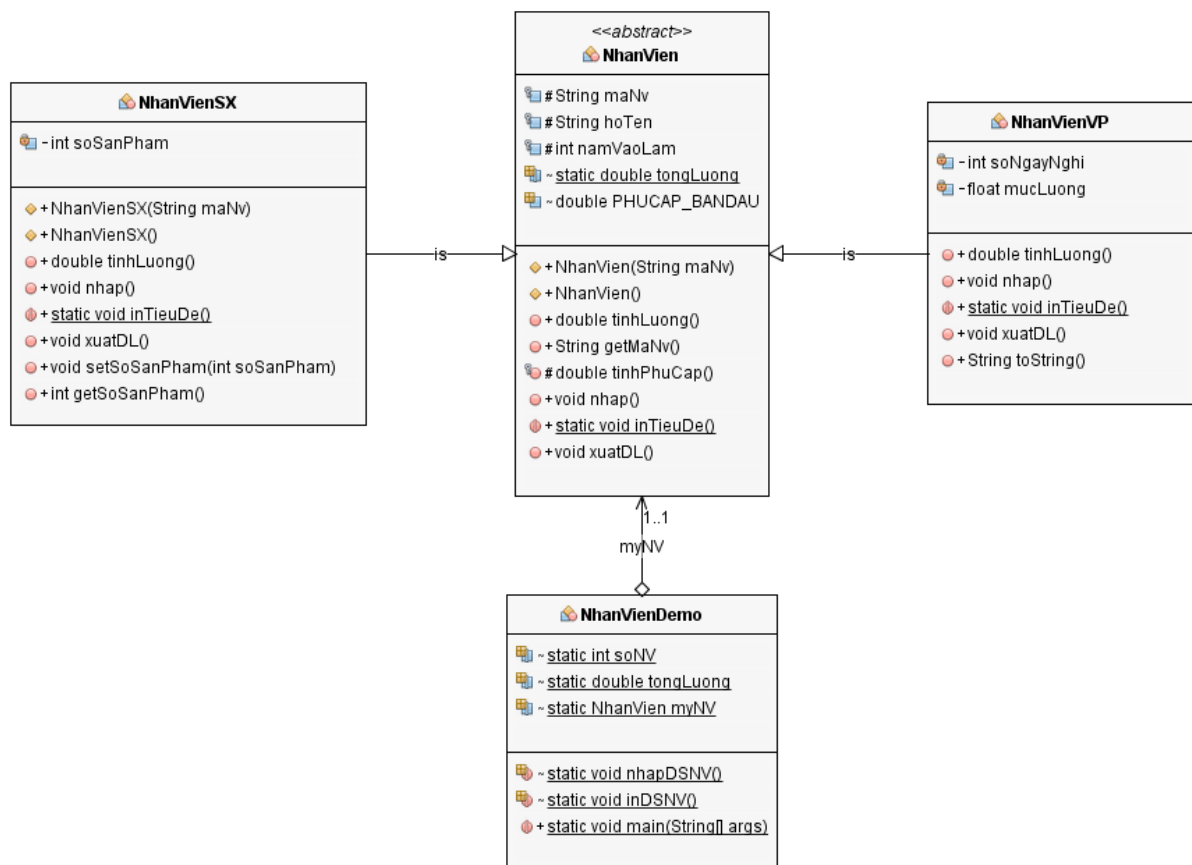
- Nhân viên sản xuất: $\text{lương} = \text{sản phẩm} \times 10000$.
- Nhân viên văn phòng: $\text{lương} = \text{mức lương} - \text{ngày nghỉ} \times 10000$.
- Ngoài lương được tính như trên, mỗi nhân viên còn được phụ cấp một khoảng tiền là 100000. Và khoảng tiền này cứ tăng thêm 20000 cho mỗi năm công tác ở công ty.

3.2. Yêu cầu:

- Viết chương trình: Nhập vào danh sách nhân viên công ty.
- Tính tổng số tiền công ty phải trả cho nhân viên mỗi tháng.

3.3. Hướng dẫn cài đặt:

Mô hình hóa sơ đồ lớp thông qua phân tích, hướng dẫn ta có sơ đồ mô tả quan hệ các lớp như sau.



**NhanVien.java**

```
abstract class NhanVien{
    protected String maNv;
    protected String hoTen;
    protected int namVaoLam;
    static double tongLuong;
    final double PHUCAP_BANDAU=100000;
    abstract public double tinhLuong();// chú ý phương thức trừu tượng
}

//lấy mã nhân viên
    public String getMaNv() {
        return maNv;
    }

protected double tinhPhuCap() {
    Date now = new Date();
    int d=LocalDate.now().getYear();
    return PHUCAP_BANDAU+(d-namVaoLam)*20000;
}

public NhanVien(String maNv) {
    this.maNv = maNv;
    this.hoTen = "";
    this.namVaoLam = 2012;
}

public NhanVien() {
    this.maNv = "nv0";
    this.hoTen = "";
    this.namVaoLam = 2000;
}

public void nhap() {
    Scanner s=new Scanner(System.in);
    System.out.print("Nhap ma nv =");
    maNv=s.nextLine();
    System.out.print("Nhap ho ten=");
    hoTen=s.nextLine();
    System.out.print("Nhap nam vao lam=");
    namVaoLam=s.nextInt();
}

public static void inTieuDe() {
    System.out.printf("%-10s %-15s %6s ", " Ma nv", " ho ten", "năm vào làm" );
}

public void xuatDL(){
```



```
        System.out.printf("%-10s %-15s %6d\n", maNv, hoTen, namVaoLam );
    }
} //end of class
```

NhanVienSX.java

```
class NhanVienSX extends NhanVien{
    private int soSanPham;

    public NhanVienSX(String maNv){
        super(maNv);
        soSanPham=0;
    }

    public NhanVienSX() {
        super();
        soSanPham=0;
    }

    public double tinhLuong()
    {
        return soSanPham*10000;
    }

    public void nhap() {
        super.nhap();
        Scanner s=new Scanner(System.in);
        System.out.print("Nhap so san pham=");
        soSanPham=s.nextInt();
    }

    public static void inTieuDe(){
        NhanVien.inTieuDe();
        System.out.printf("%10s %15s %15s\n", " So sp", " phu cap", " Luong");
    }

    public void xuatDL(){
        super.xuatDL();
        System.out.printf("%10d %15.2f %15.2f\n", soSanPham, tinhPhuCap(), tinhLuong());
    }

    public void setSoSanPham(int soSanPham) {
        this.soSanPham = soSanPham;
    }

    public int getSoSanPham() {
        return soSanPham;
    }
}
```



```
}
```

NhanVienVP.java

```
class NhanVienVP extends NhanVien{
    private int soNgayNghỉ;
    private float mucLuong;

    @Override
    public double tinhLuong(){
        return mucLuong-soNgayNghỉ*10000;
    }

    @Override
    public void nhap() {
        super.nhap();
        Scanner s=new Scanner(System.in);
        System.out.print("Nhập số ngày nghỉ=");
        soNgayNghỉ=s.nextInt();
        System.out.print("Mức lương="); mucLuong=s.nextInt();
    }

    public static void inTieuDe(){
        NhanVien.inTieuDe();
        System.out.printf("%10s %15s %15s %15s\n"," So ngày
ngỉ"," mức lương", " phụ cấp"," Lương");
    }

    public final void xuatDL(){
        super.xuatDL();
        System.out.printf("%10d %15f %15.2f
%15.2f\n",soNgayNghỉ,mucLuong,tinhPhuCap(),tinhLuong());
    }

    public String toString(){
        return super.toString()+ "\t số ngày
ngỉ="+soNgayNghỉ+
        "\tmức lương="+mucLuong;
    }
}
```

NhanVienDemo.java

```
import java.text.DecimalFormat;
import java.util.Scanner;
public class NhanVienDemo{
    static int soNV,loaiNV;
    static double tongLuong=0f;
    static NhanVien myNV[]=new NhanVien[soNV];

    static void nhapDSNV() {
        Scanner s=new Scanner(System.in);
        System.out.print("Nhap so luong nhan vien=");
        soNV=s.nextInt();
        //cấp phát n biến mảng theo lớp cha
        for(int i=0;i<soNV;i++){
            System.out.print("1.Nhập NVSX, 2.NVVP");
            loaiNV=s.nextInt();
            if (loaiNV==1)
                myNV[i]=new NhanVienSX();
            else
                myNV[i]=new NhanVienVP();
            myNV[i].nhap();
            //tính tổng lương
            tongLuong=tongLuong+myNV[i].tinhLuong()
                +myNV[i].tinhPhuCap();
        }
    }

    static void inDSNV() {
        System.out.println("\nDanh sach nhan vien SX cong ty
        la");
        NhanVienSX.inTieuDe();
        for(int i=0;i<soNV;i++)
            if (myNV[i] instanceof NhanVienSX)
                myNV[i].xuatDL();
        System.out.println("\nDanh sach nhan vien VP cong ty
        la");
        NhanVienVP.inTieuDe();
        for(int i=0;i<soNV;i++)
            if (myNV[i] instanceof NhanVienVP)
                myNV[i].xuatDL();
        DecimalFormat f = new DecimalFormat("###,###.0#");
        System.out.println("Tong luong nhan vien:"
        +f.format(tongLuong));
    }
}
```

```
public static void main(String[] args) {  
    nhapDSNV();  
    inDSNV();  
}  
}
```