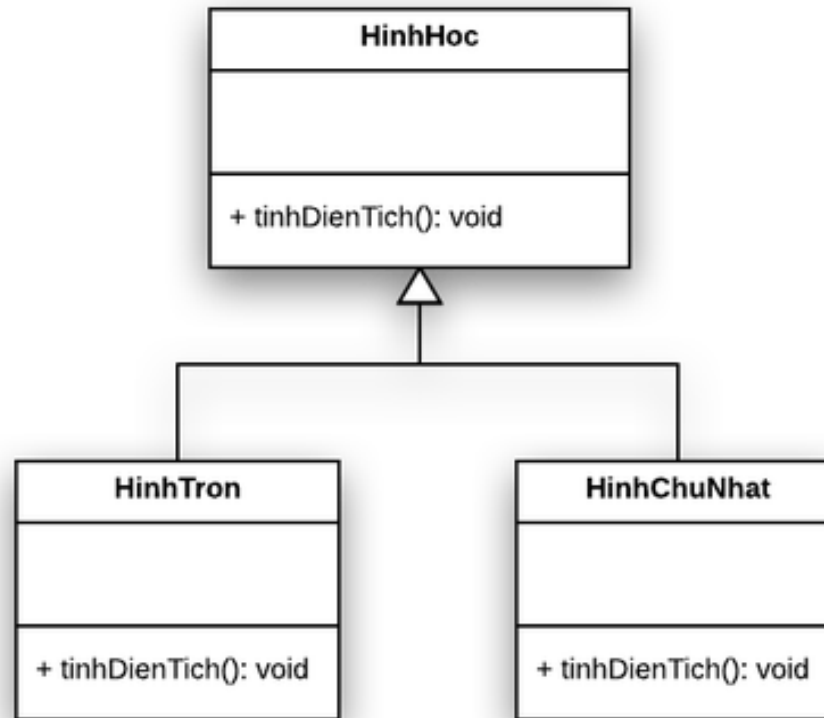


Chương 4. Tính đa hình – Polymorphism

Khái niệm



- *Đa hình* cho phép 1 đối tượng có nhiều thể hiện khác nhau tùy vào ngữ cảnh
- Ví dụ:



Chương 4. Tính đa hình – Polymorphism

Sử dụng tính đa hình



- **HinhHoc:**

```
public class HinhHoc {  
    public void tinhDienTich() {  
        System.out.println("Chưa biết hình nào");  
    }  
}
```

- **HinhTron:**

```
public class HinhTron extends HinhHoc {  
    @Override  
    public void tinhDienTich() {  
        System.out.println("Đây là Diện tích hình Tròn");  
    }  
}
```

- **HinhChuNhat:**

```
public class HinhChuNhat extends HinhHoc {  
    @Override  
    public void tinhDienTich() {  
        System.out.println("Đây là Diện tích hình Chữ nhật");  
    }  
}
```

Chương 4. Tính đa hình – Polymorphism

Sử dụng tính đa hình



▪ Hàm Main:

```
public class MainClass {
```

```
    public static void main(String[] args) {
```

```
        HìnhHoc hìnhHoc = new HìnhHoc();
```

```
        hìnhHoc.tinhDienTich(); // in ra "Chưa biết hình nào"
```

```
        // hìnhHoc đóng vai trò là HìnhTron trong một ngữ cảnh nào đó
```

```
        hìnhHoc = new HìnhTron();
```

```
        hìnhHoc.tinhDienTich(); // in ra "Đây là Diện tích Hình tròn"
```

```
        // hìnhHoc đóng vai trò là HìnhChuNhat trong một ngữ cảnh nào đó
```

```
        hìnhHoc = new HìnhChuNhat();
```

```
        hìnhHoc.tinhDienTich(); // in ra "Đây là Diện tích Chữ nhật"
```

```
    }
```

```
}
```

Lập trình hướng đối tượng

Chương 4. Tính đa hình – Polymorphism

Ví dụ về tính đa hình



Ứng dụng của chúng ta phục vụ một công ty nhỏ. Với một số nguyên tắc sau.

- Công ty này có hai loại nhân viên, đó là ***nhân viên toàn thời gian*** và ***nhân viên thời vụ***.
- ***Nhân viên toàn thời gian*** là ***lính*** sẽ hưởng lương ***10 triệu*** một tháng. ***Nhân viên toàn thời gian*** là ***sếp*** sẽ hưởng lương ***20 triệu*** một tháng.
- ***Nhân viên toàn thời gian*** nếu làm thêm ngày nào thì sẽ được cộng thêm ***800 000*** mỗi ngày, bất kể chức vụ.
- ***Nhân viên thời vụ*** cứ làm mỗi giờ được ***100 000***, không phân biệt chức vụ gì cả. Làm nhiều thì hưởng nhiều.

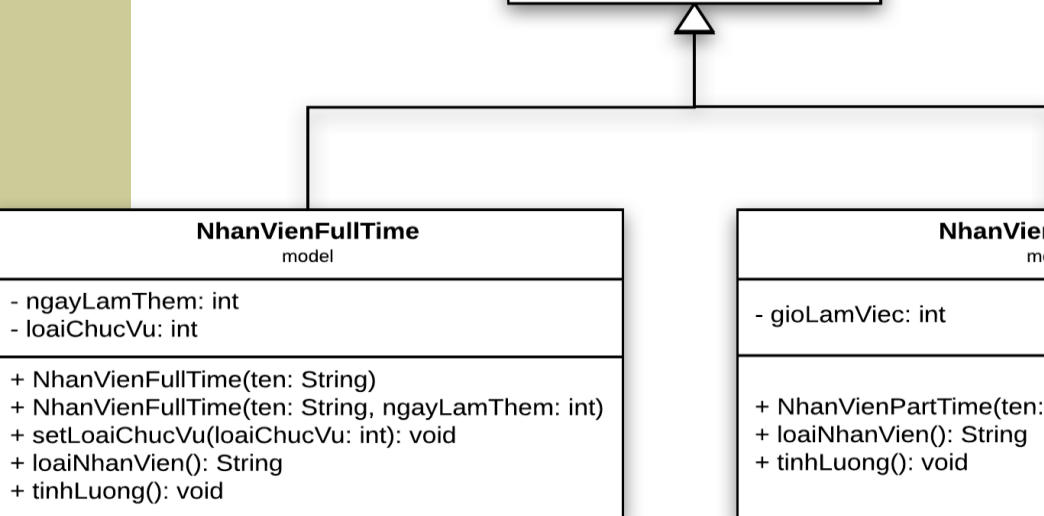
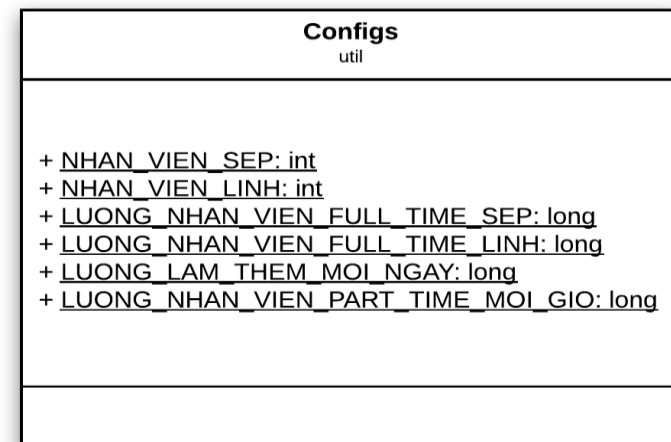
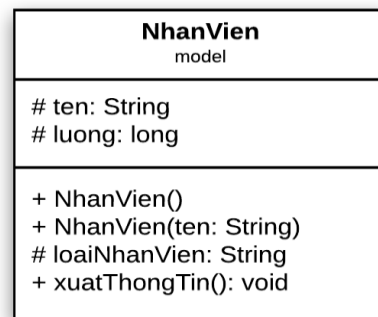
Ứng dụng sẽ cho phép nhập vào từng nhân viên. Mỗi nhân viên có tên nhân viên, loại nhân viên toàn thời gian hay bán thời gian. Nhân viên toàn thời gian thì là nhân viên lính hay nhân viên sếp, có làm thêm ngày nào không. Nhân viên thời vụ thì làm được mấy giờ. Kết quả sẽ xuất ra màn hình thông tin và lương tương ứng.

Chương 4. Tính đa hình – Polymorphism

Ví dụ về tính đa hình



Mô hình:



Lập trình hướng đối tượng

Chương 4. Tính đa hình – Polymorphism



```
package util;

public class Configs {
    // Loại nhân viên
    public static final int NHAN_VIEN_SEP = 1;
    public static final int NHAN_VIEN_LINH = 2;
    // Lương nhân viên
    public static final long LUONG_NHAN_VIEN_FULL_TIME_SEP = 20000000;
    // Lương tháng của sếp
    public static final long LUONG_NHAN_VIEN_FULL_TIME_LINH = 10000000;
    // Lương tháng của lính
    public static final long LUONG_LAM_THEM_MOI_NGAY = 800000;
    // Làm thêm mỗi ngày của nhân viên toàn thời gian được 800 k
    public static final long LUONG_NHAN_VIEN_PART_TIME_MOI_GIO = 100000;
    // Lương nhân viên thời vụ mỗi giờ 100 k
}
```

Chương 4. Tính đa hình – Polymorphism

Ví dụ - Lớp nhân viên



package model;

public class NhanVien {

protected String ten;

protected long luong;

public NhanVien() { }

public NhanVien(String ten) {

this.ten = ten;

}

protected String loaiNhanVien() {

return "";

public void xuấtThôngTin() {

System.out.println("==== Nhân viên: " + ten + " =====");

System.out.println("- Loại nhân viên: " + loaiNhanVien());

System.out.println("- Lương: " + luong + " VND");

}

}

Lập trình hướng đối tượng

Chương 4. Tính đa hình – Polymorphism

NhanVienFullTime



```
package model;
```

```
import util.Configs;
```

```
public class NhanVienFullTime extends NhanVien {
```

```
    private int ngayLamThem; // Ngày làm thêm của nhân viên
```

```
    private int loaiChucVu; // Chức vụ là lính hay sếp
```

```
    public NhanVienFullTime(String ten) {
```

```
        super(ten);
```

```
        this.loaiChucVu = Configs.NHAN_VIEN_LINH; } // Mặc định là lính
```

```
    public NhanVienFullTime(String ten, int ngayLamThem) {
```

```
        super(ten);
```

```
        this.ngayLamThem = ngayLamThem;
```

```
        this.loaiChucVu = Configs.NHAN_VIEN_LINH; // Mặc định là lính
```

```
    }
```

```
    public void setLoaiChucVu(int loaiChucVu) {
```

```
        this.loaiChucVu = loaiChucVu; }
```

Lập trình hướng đối tượng

Chương 4. Tính đa hình – Polymorphism



@Override

```
public String loaiNhanVien() {  
    if (loaiChucVu == Configs.NHAN_VIEN_LINH) {  
        return "Lính toàn thời gian" + (ngayLamThem > 0 ? " (có làm  
thêm ngày)":"" );  
    } else {  
        return "Sếp toàn thời gian" + (ngayLamThem > 0 ? " (có làm  
thêm ngày)":"" );  
    }  
}
```

Chương 4. Tính đa hình – Polymorphism



```
public void tinhLuong() {  
    if (loaiChucVu == Configs.NHAN_VIEN_LINH) {  
        luong = Configs.LUONG_NHAN_VIEN_FULL_TIME_LINH +  
        ngayLamThem * Configs.LUONG_LAM_THEM_MOI_NGAY;  
    } else if (loaiChucVu == Configs.NHAN_VIEN_SEP) {  
        luong = Configs.LUONG_NHAN_VIEN_FULL_TIME_SEP +  
        ngayLamThem * Configs.LUONG_LAM_THEM_MOI_NGAY;  
    }  
}
```

Chương 4. Tính đa hình – Polymorphism

Nhóm NhanVienPartTime



```
package model;

import util.Configs;

public class NhanVienPartTime extends NhanVien {

    private int gioLamViec; // Tổng số giờ làm việc của nhân viên

    public NhanVienPartTime(String ten, int gioLamViec) {
        this.ten = ten;
        this.gioLamViec = gioLamViec;
    }

    @Override
    public String loaiNhanVien() {
        return "Nhân viên thời vụ";
    }

    public void tinhLuong() {
        luong = Configs.LUONG_NHAN_VIEN_PART_TIME_MOI_GIO * gioLamViec;
    }
}
```

Lập trình hướng đối tượng

Chương 4. Tính đa hình – Polymorphism

Lớp Main



```
package main;

import model.NhanVienFullTime;
import model.NhanVienPartTime;
import util.Configs;

public class MainClass {

    public static void main(String[] args) {
        // Công ty có 3 nhân viên toàn thời gian, trong đó có 1 sếp, sếp không làm thêm ngày nào
        NhanVienFullTime sep = new NhanVienFullTime("Trần Văn Sếp");
        sep.setLoaiChucVu(Configs.NHAN_VIEN_SEP);
        NhanVienFullTime linh1 = new NhanVienFullTime("Nguyễn Văn Linh"); // Không làm thêm
        NhanVienFullTime linh2 = new NhanVienFullTime("Lê Thị Linh", 3); // Làm thêm 3 ngày
```

Chương 4. Tính đa hình – Polymorphism



```
// Công ty đang thuê 1 nhân viên thời vụ
```

```
    NhanVienPartTime thoiVu = new
```

```
    NhanVienPartTime("Phan Thị Thời Vụ", 240);
```

```
// Tính lương cho nhân viên
```

```
    sep.tinhLuong();
```

```
    linh1.tinhLuong();
```

```
    linh2.tinhLuong();
```

```
    thoiVu.tinhLuong();
```

```
// In thông tin nhân viên
```

```
    sep.xuatThongTin();
```

```
    linh1.xuatThongTin();
```

```
    linh2.xuatThongTin();
```

```
    thoiVu.xuatThongTin();
```

```
}
```

Lập trình hướng đối tượng

Chương 4. Tính đa hình – Polymorphism

Từ khóa instanceof



- Dùng để kiểm tra dữ liệu của 1 đối tượng có phải là KDL đang nghi ngờ hay không. Nếu đúng trả về true, sai trả về false
- VD: `HinhHoc hh1=new HinhChuNhat();`
→ `boolean kt=hh1 instanceof HinhChuNhat;`
→ `kt=true;`