

Session 9

Modifiers and Packages

Phần I - Thực hiện trong 120 phút

1.1 Mục tiêu

- ✓ Hiểu được ý nghĩa của bổ từ final, static.
- ✓ Khai báo biến class.
- ✓ Khai báo biến, phương thức static và sử dụng.
- ✓ Xây dựng mã nguồn Java trong các package (gói).
- ✓ Đóng gói mã nguồn trong file thư viện jar.

1.2 Thực hiện

Bài thực hành 1: Viết một chương trình Java cơ bản kiểm tra tính chất của biến khi cài đặt từ khóa final.

Bước 1: Viết class tên là MainClass có hàm main

```
package demo.jp1.lab07;

/**
 *
 * @author minhvuvc
 */
public class MainClass {
    //Bước 2: Khai báo biến toàn cục tên là PI cài đặt là final và gán
    //giá trị mặc định 3.14
    final float PI = 3.14F;

    //Bước 3: Viết hàm display() nhận tham số truyền vào là một tham số kiểu dữ
    //liệu float, tham số này sẽ được gán cho biến tạo bước 2.
    public void display(float pi) {
        System.out.println("The value of PI is:" + PI);
    }

    //Bước 4: Hàm main khởi tạo đối tượng MainClass và từ đối tượng mới
    //gọi tới hàm display().
    public static void main(String[] args) {
        // Instantiate the Final class
        final MainClass objFinal = new MainClass();

        // Invoke the display() method
    }
}
```

```
objFinal.display(22.7F);  
}  
}
```

Bước 5: Chạy và xem kết quả

Bài thực hành 2: Viết một chương trình Java cơ bản kiểm tra tính chất của từ khóa static.

Bước 1: Viết class tên là NhanVien.

```
package demo.jp1.lab07;  
  
/**  
 *  
 * @author minhvuvc  
 */  
  
public class NhanVien {  
  
    String name;  
  
    static int cnt; // Biến đếm số lượng nhân viên  
  
}
```

Bước 2: Viết class NhanVienMainClass có hàm main, khởi tạo mảng 3 đối tượng nhân viên.

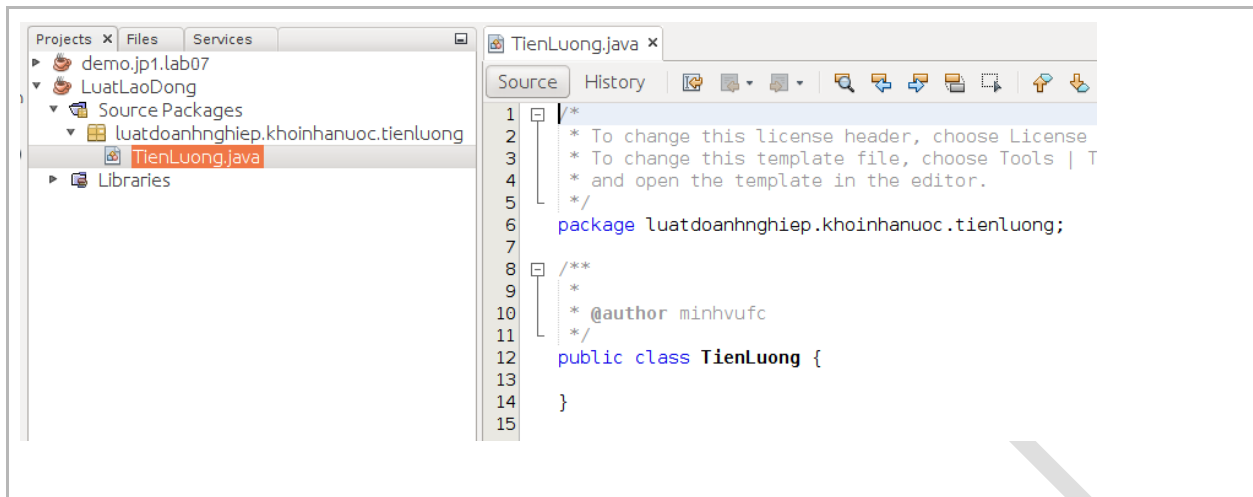
```
package demo.jp1.lab07;  
  
/**  
 *  
 * @author minhvuvc  
 */  
public class NhanVienMainClass {  
  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {
```

```
//Khai báo và khởi tạo mảng 3 đối tượng nhân viên.  
NhanVien[] arrNhanVien = new NhanVien[3];  
  
//Bước 3: Gán tên cho 3 nhân viên đồng thời tăng giá trị biến đếm cnt thêm 1  
đơn vị  
// Nhan vien 1  
arrNhanVien[0] = new NhanVien();  
arrNhanVien[0].name = "Ngoc Trinh";  
arrNhanVien[0].cnt++;  
  
// Nhan vien 2  
arrNhanVien[1] = new NhanVien();  
arrNhanVien[1].name = "Ba Tung";  
arrNhanVien[1].cnt++;  
  
// Nhan vien 3  
arrNhanVien[2] = new NhanVien();  
arrNhanVien[2].name = "Kieu Max";  
arrNhanVien[2].cnt++;  
  
//Bước 4: Chạy vòng lặp hiển thị tên của nhân viên đồng thời xem giá trị  
//của biến đếm là bao nhiêu.  
for (int i = 0; i < arrNhanVien.length; i++) {  
    NhanVien nv = arrNhanVien[i];  
    System.out.println(nv.name + " cnt = " + nv.cnt);  
}  
}
```

Bước 5: Nhận xét về tính chất của của biến khi cài đặt static: biến cnt mặc dù khởi tạo ra làm 3 đối tượng nhưng nó vẫn chỉ lưu ở 1 vùng nhớ duy nhất.

Bài thực hành 3: Viết một thư viện Java cho ứng dụng tính lương với bậc lương có 3 cấp: Trung cấp, Cao đẳng và Đại Học lần lượt có hệ số là 1.55 - 1.80 - 2.34, mức lương cơ bản là 2.889.000, công ty phải đóng bảo hiểm cho nhân viên là 16% so với mức lương của nhân viên, mỗi năm thâm niên công tác được tính thêm vào lương là 5% mức lương cơ bản. Viết một chương trình nhận vào thông số bậc lương và thâm niên công tác để tính số tiền lương phải trả và số tiền bảo hiểm công ty phải đóng.

Bước 1: Tạo project tên là **LuatLaoDong**, trong project tạo package **luatdoanhnghiep.khoinhannuoc.tienluong**, trong package này tạo 1 class tên là **TienLuong**.



Bước 2: Thực hiện theo các bước như hướng dẫn sau:

```
package luatdoanhnghiep.khoinhauoc.tienluong;
```

```
/**
 *
 * @author minhvuvc
 */
public class TienLuong {
```

//Bước 2: Khai báo các bậc lương là biến kiểu float, dạng static, final.

```
public static final float BAC_TRUNG_CAP = 1.55f;
public static final float BAC_CAO_DANG = 1.80f;
public static final float BAC_DAI_HOC = 2.34f;
```

//Bước 3: Khai báo lương cơ bản, mức % đóng bảo hiểm, hệ số thâm niên công tác theo năm.

```
//Tất cả khai báo với bộ từ static, final
public static final double LUONG_CO_BAN = 2889000;
public static final float HE_SO_BHXH = 16;
public static final float HE_SO_THAM_NIEN = 5;
```

//Bước 4: Viết static method tính lương cơ bản của nhân viên và trả về kiểu dữ liệu double

```
public static double luongCoBan(float hocVi, int namCongTac) {
    double luongCoBan = (hocVi * LUONG_CO_BAN) + namCongTac * (5 / 100 *
LUONG_CO_BAN);
    return luongCoBan;
}
```

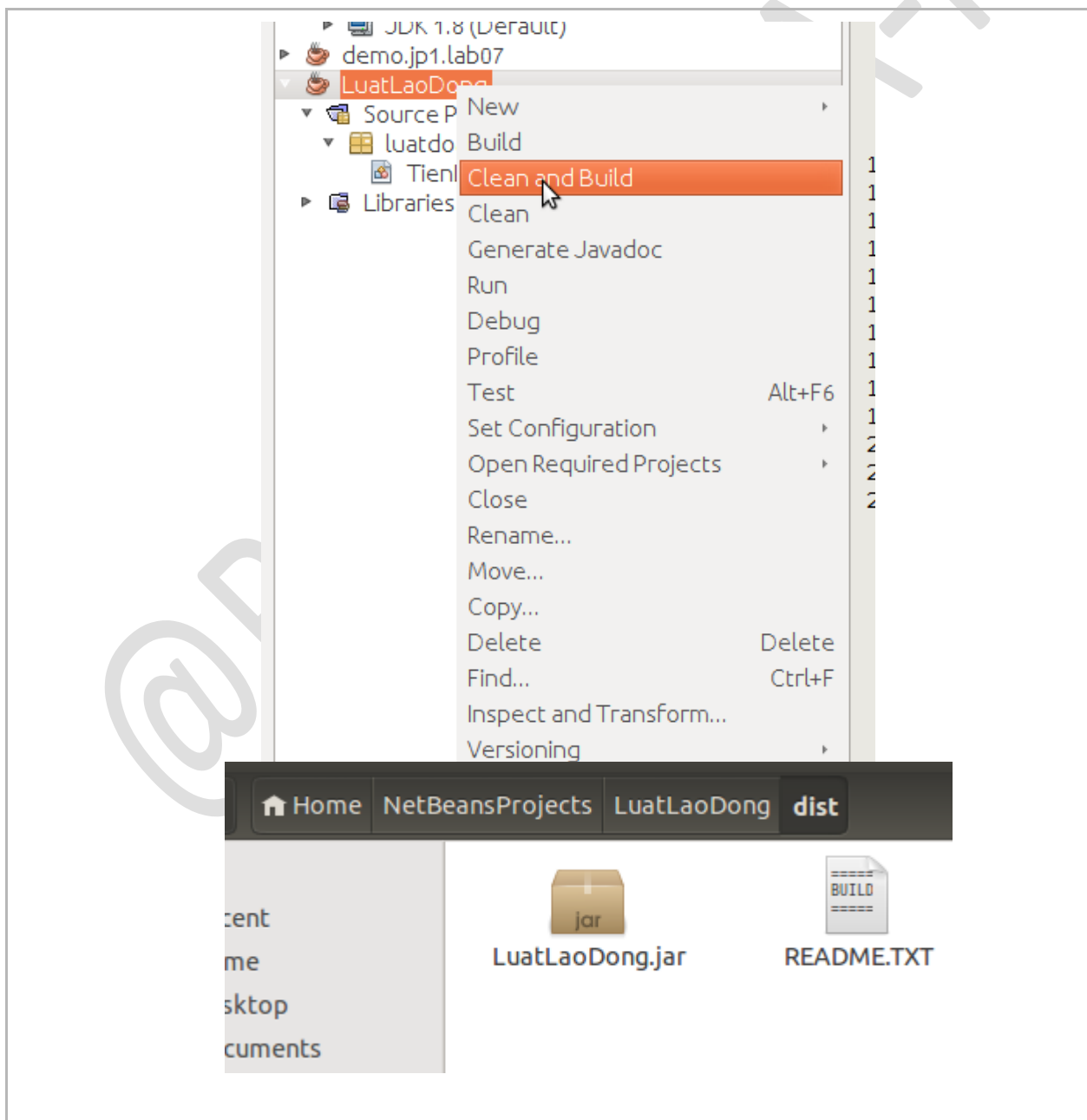
//Bước 5: Viết static method tính số tiền bảo hiểm mà công ty phải đóng dựa trên số tiền lương.

```
public static double soTienBaoHiem(double luong) {
    return (HE_SO_BHXH / 100) * luong;
```

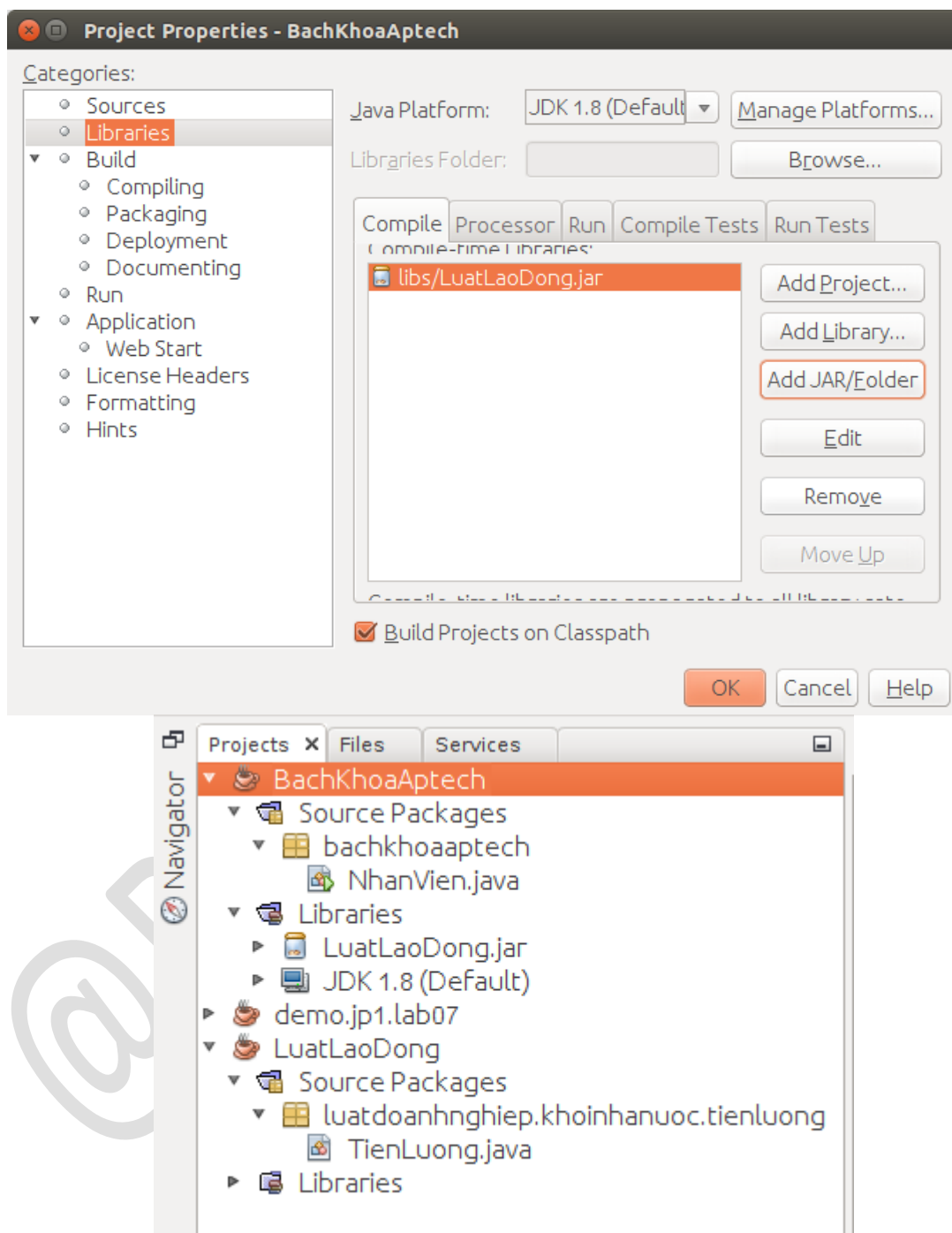
```
}  
}
```

Bước 6: Build thư viện trên và tìm tới thư mục dist nằm cùng cấp với thư mục project.

Chú giải: Bản chất mỗi project Java sau khi build có thể đóng gói thành thư viện Java và sử dụng trong các project khác. Thư viện Java có đuôi mở rộng là *.jar, tương tự như thư viện *.dll bên lập trình .NET



Bước 7: tạo project tên là **BachKhoaAptech**, có class chứa hàm main tên là **NhanVien**. Copy thư viện **.jar** này rồi bỏ vào thư mục **libs** nằm cùng cấp thư mục với project. Import project trên dưới dạng thư viện file **jar**.

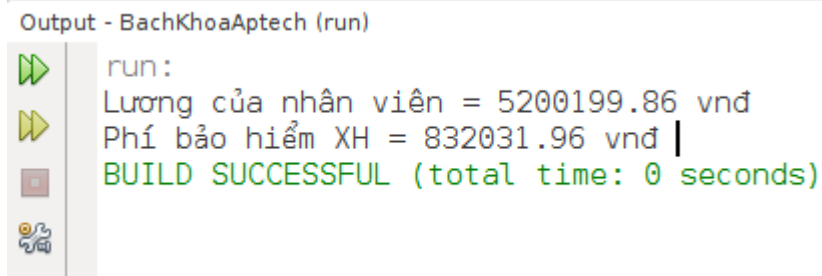


```
package bachkhoaaptech;
```

```
import luatdoanhnghiep.khoinhauoc.tienluong.TienLuong;
```

```
/**
 *
 * @authorminhvufc
 */
public class NhanVien {
    public static void main(String[] args) {
        int tham_nien_cong_tac = 8;
        double luong = TienLuong.luongCoBan(TienLuong.BAC_CAO_DANG,
        tham_nien_cong_tac);
        System.out.printf("Lương của nhân viên = %.2f VNĐ", luong);
        System.out.printf("\nPhí bảo hiểm XH = %.2f VNĐ \n",
        TienLuong.soTienBaoHiem(luong));
    }
}
```

Bước 8: Nhập dữ liệu để kiểm thử.



```
Output - BachKhoaAptech (run)
run:
Lương của nhân viên = 5200199.86 VNĐ
Phí bảo hiểm XH = 832031.96 VNĐ |
BUILD SUCCESSFUL (total time: 0 seconds)
```

Phần II - Bài tập tự làm**Bài 1:**

1. Tạo class City gồm các thông tin:

```
private String cityId; //Mã thành phố  
private String cityName; //Tên thành phố  
private String country; //Tên nước  
private double area; //diện tích
```

Tạo 2 constructors.

Tạo đầy đủ các phương thức set/get cho các thuộc tính của lớp.

Tạo hàm nhập() và hàm hienthi()

2. Tạo class ManagerCity, trong có hàm main và thực hiện theo menu sau:

1. Nhập thông tin n thành phố
2. Hiển thị thông tin vừa nhập
3. Sắp xếp thông tin tăng dần theo tên nước.
4. Tìm kiếm thông tin theo cityId
5. Thoát.

Bài 2:

1. Tạo class Flower gồm các thông tin:

```
private String flowerID; //mã nhập  
private String flowerName; // tên hoa  
private String color; // màu hoa  
private int kindOfFlower; //số chủng loại của loài hoa này (Hoa Hồng: Hồng  
Đỏ, Hồng Vàng, Hồng Trắng => số chủng loại = 3)
```

Tạo 2 constructors.

Tạo các phương thức get/set

Tạo hàm input(), display().

2. Tạo class FlowerTest, có hàm main, tạo menu và thực thi theo menu sau:

- a. Nhập thông tin n loài hoa
- b. Hiển thị thông tin vừa nhập

- c. Sắp xếp danh sách tăng dần theo tên hoa
- d. Tìm kiếm theo flowerID
- e. Hiển thị tất cả các loại hoa có màu trắng
- f. Thoát.

@BKAP-JAVA-I