

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Кафедра «Електронних обчислювальних машин»



Звіт  
з лабораторної роботи № 4  
з дисципліни: «Кросплатформенні засоби програмування»  
на тему: «Спадкування та Інтерфейси»

**Виконав:**

студент групи КІ-35

Андрусяк М.В.

**Прийняв:**

доцент кафедри ЕОМ

Іванов Ю. С.

**Мета роботи:** ознайомитися з спадкуванням та інтерфейсами у мові Java.

### Завдання (варіант № 27)

1. Написати та налагодити програму на мові Java, що розширює клас, що реалізований у лабораторній роботі №3, для реалізації предметної області заданої варіантом. Суперклас, що реалізований у лабораторній роботі №3, зробити абстрактним. Розроблений підклас має забезпечувати механізми свого коректного функціонування та реалізовувати мінімум один інтерфейс. Програма має розміщуватися в пакеті Група.Прізвище.Lab4 та володіти коментарями, які дозволять автоматично згенерувати документацію до розробленого пакету.
2. Автоматично згенерувати документацію до розробленого пакету.
3. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації.
4. Дати відповідь на контрольні запитання.

### Варіант N-27 – Інтелектуальний патрон

#### Текст програми

##### CartridgeApp

```
package Lab4AndrusiakKI35;
import java.io.*;
import java.util.Scanner;

/**
 * Cartridge Application class implements main method for Cartridge class
 * possibilities demonstration
 * @author Andrusiak Marko KI35
 * @version 1.0
 */

public class CartridgeApp {
    /**
     * @param args
     * @throws FileNotFoundException
     */
    public static void main(String[] args) throws FileNotFoundException {
        SmartCartridge smartCartridge = new SmartCartridge(new Bullet(7.62f,
15),
                new Powder("alum", 30),
                new CartridgeCase("round", 1917), 3600 );

        smartCartridge.autoAim();
        smartCartridge.changeShape("acuminate");
    }
}
```

```
        smartCartridge.dispose();  
    }  
}
```

---

## AutoAimCartridge

```
package Lab4AndrusiakKI35;  
  
public interface AutoAimCartridge {  
    void autoAim();  
}
```

---

## ChangeShapeOfCC

```
package Lab4AndrusiakKI35;  
  
public interface ChangeShapeOfCC {  
    void changeShape(String newShape);  
}
```

---

## Bullet

```
package Lab4AndrusiakKI35;  
  
public class Bullet {  
    private float bulletDiameter;  
    private int width;  
  
    public Bullet(float bulletDiameter, int width) {  
        this.bulletDiameter = bulletDiameter;  
        this.width = width;  
    }  
  
    public float getBulletDiameter() {  
        return bulletDiameter;  
    }  
  
    public void setBulletDiameter(float bulletDiameter) {  
        this.bulletDiameter = bulletDiameter;  
    }  
  
    public int getWidth() {  
        return width;  
    }  
  
    public void setWidth(int width) {  
        this.width = width;  
    }  
}
```

---

# Cartridge

```
package Lab4AndrusiakKI35;
import java.io.File;

import java.io.FileNotFoundException;
import java.io.PrintWriter;

public abstract class Cartridge {

    //-----Об'єкти класу-----//
    private Bullet bullet;
    private Powder powder;
    private CartridgeCase cartridgeCase;
    private int bulletEnergy;
    public PrintWriter fout; // об'єкт для запису інформації
    //-----ОБ'ЄКТИ КЛАСУ-----//

    //-----КОНСТРУКТОРИ-----//
    public Cartridge(Bullet bullet, Powder powder, CartridgeCase cartridgeCase,
                    int bulletEnergy) throws FileNotFoundException {
        this.bullet = bullet;
        this.cartridgeCase = cartridgeCase;
        this.powder = powder;
        this.bulletEnergy = bulletEnergy;

        fout = new PrintWriter(new File("MethodDesc.txt"));
        fout.print("The cartridge was created!\n");
    }

    public Cartridge(Bullet bullet, Powder powder) throws FileNotFoundException
    {
        this.bullet = bullet;
        this.powder = powder;

        fout = new PrintWriter(new File("MethodDesc.txt"));
        fout.print("The cartridge was created!\n");
    }

    public Cartridge(CartridgeCase cartridgeCase, int bulletEnergy) throws
    FileNotFoundException {
        this.bulletEnergy = bulletEnergy;
        this.cartridgeCase = cartridgeCase;

        fout = new PrintWriter(new File("MethodDesc.txt"));
        fout.print("The cartridge was created!\n");
    }

    //-----КОНСТРУКТОРИ-----//

    //-----МЕТОДИ КЛАСУ-----//

    public void Fire() { // ІМІТАЦІЯ СТРІЛЬБИ
        fout.print("--->FIRE\n");
        powder.setAmount(0);
    }

    // ПЕРЕВІРКА ПУЛІ НА ПРОБИТТЯ БРОНІ
    public void bulletTest() {
        if (bulletEnergy >= 3000 && bullet.getBulletDiameter() > 5) {
            fout.print("BulletTest : Bullet penetration APPROVED\n");
        } else {
            fout.print("BulletTest : I recommend finding a larger
cartridge!\n");
        }
    }
}
```

```

// ПЕРЕВІРКА НА АКТУАЛЬНІСТЬ ПАТРОНУ
public void isThisCartridgeActual() {
    if (cartridgeCase.getCreationTime() <= 1930) {
        fout.print("ITCA : Sorry man, it`s too OLD\n");
    } else {
        fout.print("ITCA : Bright as a Button!\n");
    }
}

// ГЕТЕРИ
public Bullet getBullet() {
    return bullet;
}

public Powder getPowder() {
    return powder;
}

public CartridgeCase getCartridgeCase() {
    return cartridgeCase;
}

public int getBulletEnergy() {
    return bulletEnergy;
}

// ГЕТЕРИ

// СЕТЕРИ

public void setBullet(float bulletDiameter, int width) {
    fout.print("Changed : " + bullet.getBulletDiameter() + " to " +
bulletDiameter + "\n");
    fout.print("Changed : " + bullet.getWidth() + " to " + width + "\n");

    bullet.setBulletDiameter(bulletDiameter);
    bullet.setWidth(width);
}

public void setPowder(String type, int amount) {
    fout.print("Changed : " + powder.getAmount() + " to " + amount + "\n");
    fout.print("Changed : " + powder.getType() + " to " + type + "\n");

    powder.setAmount(amount);
    powder.setType(type);
}

public void setCartridgeCase(String shape, int creationTime) {
    fout.print("Changed : " + cartridgeCase.getShape() + " to " + shape
+ "\n");
    fout.print("Changed : " + cartridgeCase.getCreationTime() + " to " +
creationTime + "\n");

    cartridgeCase.setShape(shape);
    cartridgeCase.setCreationTime(creationTime);
}

public void setBulletEnergy(int bulletEnergy) {
    this.bulletEnergy = bulletEnergy;
}

// СЕТЕРИ

public void dispose() {
    fout.close();
}

```

```
}  
}
```

---

## CartridgeCase

```
package Lab4AndrusiakKI35;  
  
public class CartridgeCase {  
    private String shape;  
    private int creationTime;  
  
    public CartridgeCase(String shape, int creationTime) {  
        this.shape = shape;  
        this.creationTime = creationTime;  
    }  
  
    public String getShape() {  
        return shape;  
    }  
  
    public void setShape(String shape) {  
        this.shape = shape;  
    }  
  
    public int getCreationTime() {  
        return creationTime;  
    }  
  
    public void setCreationTime(int creationTime) {  
        this.creationTime = creationTime;  
    }  
}
```

---

## Powder

```
package Lab4AndrusiakKI35;  
  
public class Powder {  
    private String type;  
    private int amount;  
  
    public Powder(String type, int amount) {  
        this.type = type;  
        this.amount = amount;  
    }  
  
    public String getType() {  
        return type;  
    }  
  
    public void setType(String type) {  
        this.type = type;  
    }  
  
    public int getAmount() {  
        return amount;  
    }  
  
    public void setAmount(int amount) {
```

```

        this.amount = amount;
    }
}

-----

SmartCartridge

package Lab4AndrusiakKI35;

import java.io.FileNotFoundException;

public class SmartCartridge extends Cartridge implements AutoAimCartridge,
ChangeShapeOfCC {

    public SmartCartridge(Bullet bullet, Powder powder, CartridgeCase
cartridgeCase, int bulletEnergy) throws FileNotFoundException {
        super(bullet, powder, cartridgeCase, bulletEnergy);
    }

    public SmartCartridge(Bullet bullet, Powder powder) throws
FileNotFoundException {
        super(bullet, powder);
    }

    public SmartCartridge(CartridgeCase cartridgeCase, int bulletEnergy) throws
FileNotFoundException {
        super(cartridgeCase, bulletEnergy);
    }

    @Override
    public void autoAim() {
        fout.print("The target is captured!\n");
    }

    @Override
    public void changeShape(String newShape) {
        String oldShape = super.getCartridgeCase().getShape();
        super.setCartridgeCase(newShape, 2022);
    }
}

```

## Результат виконання програми

```

MethodDesc.txt × ChangeShapeOfCC.java ×
1 The cartridge was created!
2 The target is captured!
3 Changed : round to acuminate
4 Changed : 1917 to 2022
5

```

## Відповіді на контрольні запитання

### Синтаксис реалізації спадкування:

```
class Підклас extends Суперклас  
{  
    Додаткові поля і методи  
}
```

В термінах мови Java базовий клас найчастіше називається суперкласом, а похідний клас – підкласом. Дана термінологія запозичена з теорії множин, де підмножина міститься у супермножині.

**Висновок :** на даній лабораторній роботі я ознайомився із спадкуванням та інтерфейсами у мові Java. В результаті виконання роботи, я написав та налагодив програму, що розширює клас, що реалізований у лабораторній роботі №3.