Міністерство освіти і науки України

Національний університет «Львівська політехніка»

# Кафедра ЕОМ



Звіт

до лабораторної роботи №2

з дисципліни: «Кросплатформенні засоби програмування»

На тему: «КЛАСИ ТА ПАКЕТИ »

Виконав: ст. гр. КІ-307  
Ластовецький М.Я  
Прийняв:  
Іванов Ю.С

**Львів 2023**

**Лабораторна робота №2**

**Мета:** ознайомитися з процесом розробки класів та пакетів мовою Java.   
**Варіант №13**

13. Телефон

**Код програми:**

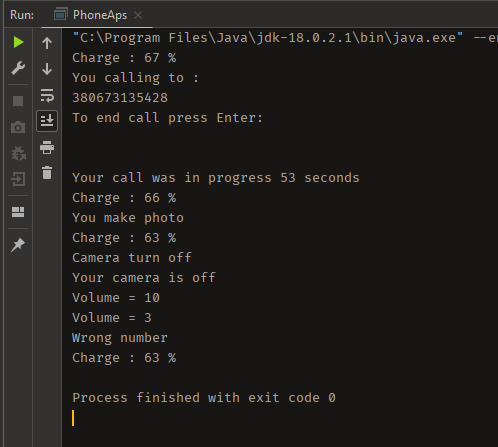
**Файл PhoneAps.java**

/\*\*  
 \* Lab3 package  
 \*/  
package Lab\_3\_lahan;  
  
  
import java.io.\*;  
  
/\*\*  
 \* Phone Application class implements main method for Phone  
 \* class possibilities demonstration  
 \* @author roman  
 \* @version 1.0  
 \* @see Phone  
 \*/  
  
public class PhoneAps {  
  
 public static void main(String[] args) throws FileNotFoundException {  
 Phone xphone = new Phone(67);  
 xphone.showCharge();  
 //Call  
 xphone.callTO("380673135428");  
 xphone.showCharge();  
 //make Picture  
 xphone.makePicture();  
 xphone.showCharge();  
 //turn off camera  
 xphone.turnOn\_Off();  
 //Make picture with turned off camera  
 xphone.makePicture();  
 //Change Volume  
 xphone.clickUpButton();  
 xphone.clickDownButton(7);  
 //Invalid call  
 xphone.callTO("1111111");  
 xphone.showCharge();  
 //End  
 xphone.fileClose();  
  
 }  
}

**Файл Phone.java**

/\*\*  
 \* Lab 3 package  
 \*/  
package Lab\_3\_lahan;  
  
import java.io.\*;  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
import java.util.Scanner;  
  
/\*\*  
 \* Class <code>Phone</code> implements phone  
 \* @author roman  
 \* @version 1.0  
 \*/  
public class Phone {  
 private VolButton Button;  
 private Battery Bat;  
 private PrintWriter fout;  
 private Camera Camera;  
  
 /\*\*  
 \* Constructor  
 \*  
 \* @throws FileNotFoundException the file not found exception  
 \*/  
 public Phone() throws FileNotFoundException {  
 Button = new VolButton();  
 Bat = new Battery();  
 Camera = new Camera();  
 fout = new PrintWriter(new File("Log.txt"));  
 }  
  
 /\*\*  
 \* Constructor  
 \*  
 \* @param charge the Battery Charge value  
 \* @throws FileNotFoundException the file not found exception  
 \*/  
 public Phone(int charge) throws FileNotFoundException {  
 Button = new VolButton();  
 Bat = new Battery(charge);  
 Camera = new Camera();  
 fout = new PrintWriter(new File("Log.txt"));  
 }  
  
 /\*\*  
 \* Method checks if the phone number is correct  
 \* @param s the phone number  
 \* @return is phone number Valid  
 \*/  
 private static boolean isValid(String s) {  
 Pattern p = Pattern.compile("380\\d{9}");  
 Matcher m = p.matcher(s);  
 return (m.matches());  
 }  
  
 /\*\*  
 \* Method calls the phone number  
 \*  
 \* @param phoneNumber the Phone number  
 \*/  
 public void callTO(String phoneNumber) {  
 //Is phone number correct  
 if (isValid(phoneNumber)) {  
 //is phone battery charged  
 if (Bat.getBatteryCharge() >= 0) {  
 //Start calling  
 System.out.println("You calling to :\n" + phoneNumber);  
 //Time in start calling  
 long time1 = System.currentTimeMillis();  
 //End call  
 System.out.println("To end call press Enter:");  
 Scanner s = new Scanner(System.in);  
 s.nextLine();  
 s.close();  
 //Time in end calling  
 long time2 = System.currentTimeMillis();  
  
 System.out.println("Your call was in progress " + (time2 - time1) / 1000 + " seconds");  
  
 Bat.setBatteryCharge((int) (Bat.getBatteryCharge() - (time2 - time1) / 30000));  
  
 fout.println("You call to " + phoneNumber + "\t and spoke " + (time2 - time1) / 1000 + " seconds");  
  
 } else {  
 System.out.println("Your battery is drained");  
 }  
 } else {  
 System.out.println("Wrong number");  
 }  
  
 }  
  
 /\*\*  
 \* Method make picture on camera  
 \*/  
 public void makePicture() {  
  
 if (Camera.getState()) {  
 // camera on  
  
 if (Bat.getBatteryCharge() >= 0) {  
 // battery charged  
 Bat.setBatteryCharge(Bat.getBatteryCharge() - 3);  
 System.out.println("You make photo");  
 fout.println("You made photo");  
  
 } else  
 // battery is not charged  
 System.out.println("Your battery is drained");  
  
 } else  
 // camera off  
  
 System.out.println("Your camera is off");  
  
 }  
  
 /\*\*  
 \* Method changes the state of camera to opposite  
 \*/  
 public void turnOn\_Off(){  
 if(Camera.getState())  
 {  
 Camera.setState(false);  
 System.out.println("Camera turn off");  
 }  
 else  
 {  
 Camera.setState(true);  
 System.out.println("Camera turn on");  
 }  
 }  
  
 /\*\*  
 \* Method click on upper volume button  
 \*/  
 public void clickUpButton()  
 {  
  
 Button.ClickUpButton();  
 System.out.println("Volume = "+Button.getVolume());  
 fout.println("Volume changed to "+Button.getVolume());  
  
 }  
  
 /\*\*  
 \* Method click on upper volume button n times  
 \*  
 \* @param n the n  
 \*/  
 public void clickUpButton(int n)  
 { for(int i =0;i<n;i++)  
 Button.ClickUpButton();  
 System.out.println("Volume = "+Button.getVolume());  
 fout.println("Volume changed to "+Button.getVolume());  
 }  
  
 /\*\*  
 \* Method click on lower volume button  
 \*/  
 public void clickDownButton()  
 {  
 Button.ClickDownButton();  
 System.out.println("Volume = "+Button.getVolume());  
 fout.println("Volume changed to "+Button.getVolume());  
 }  
  
 /\*\*  
 \* Method click on lower volume button n times  
 \*  
 \* @param n the n  
 \*/  
 public void clickDownButton(int n )  
 { for(int i =0;i<n;i++)  
 Button.ClickDownButton();  
 System.out.println("Volume = "+Button.getVolume());  
 fout.println("Volume changed to "+Button.getVolume());  
 }  
  
 /\*\*  
 \* Method close file  
 \*/  
 public void fileClose(){fout.close();}  
  
 /\*\*  
 \* Method print in console phone charge  
 \*/  
 public void showCharge(){  
 System.out.println("Charge : "+Bat.getBatteryCharge()+" %");  
 }  
  
  
  
}  
  
/\*\*  
 \* Class <code>Battery</code> implements phone battery  
 \*/  
class Battery {  
 private int BatteryCharge;  
 private static final int Max\_BatteryCharge = 100;  
 private static final int Min\_BatteryCharge = 0;  
  
 /\*\*  
 \* Constructor  
 \*/  
 public Battery() {  
 BatteryCharge = 100;  
 }  
  
 /\*\*  
 \* Constructor  
 \*  
 \* @param num the Battery Charge value  
 \*/  
 public Battery(int num) {  
 if (num > Max\_BatteryCharge) {  
 BatteryCharge = Max\_BatteryCharge;  
 } else BatteryCharge = Math.max(num, Min\_BatteryCharge);  
  
 }  
  
 /\*\*  
 \* Method set Battery Charge value in range[Min\_BatteryCharge,Max\_BatteryCharge]  
 \*  
 \* @param num the Battery Charge value  
 \*/  
 public void setBatteryCharge(int num) {  
 if (num > Max\_BatteryCharge) {  
 BatteryCharge = Max\_BatteryCharge;  
 } else BatteryCharge = Math.max(num, Min\_BatteryCharge);  
 }  
  
 /\*\*  
 \* Method return Battery Charge value  
 \*  
 \* @return the Battery Charge value  
 \*/  
 public int getBatteryCharge() {  
 return BatteryCharge;  
 }  
}  
  
/\*\*  
 \* Class <code>Camera</code> implements camera  
 \*/  
class Camera {  
 /\*\*  
 \* The State.  
 \*/  
 boolean state;  
  
 /\*\*  
 \* Constructor  
 \*/  
 public Camera() {  
 state = true;  
 }  
  
 /\*\*  
 \* Constructor  
 \*  
 \* @param status the status  
 \*/  
 public Camera(boolean status) {  
 state = status;  
 }  
  
 /\*\*  
 \* Method returns Camera State  
 \*  
 \* @return the camera state  
 \*/  
 public boolean getState() {  
 return state;  
 }  
  
 /\*\*  
 \* Method set Camera state  
 \*  
 \* @param status the status  
 \*/  
 public void setState(boolean status) {  
 state = status;  
 }  
  
  
  
}  
  
/\*\*  
 \* Class <code>VolButton</code> implements Volume button  
 \*/  
class VolButton {  
 private static final int Min\_Volume = 0;  
 private static final int MAX\_Volume = 10;  
 private int Volume;  
  
 /\*\*  
 \* Constructor  
 \*/  
 public VolButton() {  
 Volume = MAX\_Volume;  
  
 }  
  
 /\*\*  
 \* Constructor  
 \*  
 \* @param Vol Volume in range[Min\_Volume,Max\_Volume]  
 \*/  
 public VolButton(int Vol) {  
 if (Vol > MAX\_Volume) {  
 Volume = MAX\_Volume;  
 } else Volume = Math.max(Vol, Min\_Volume);  
 }  
  
 /\*\*  
 \* Method Simulate increasing Volume by 1 in range[Min\_Volume,Max\_Volume]  
 \*/  
 public void ClickUpButton() {  
 if (Volume != MAX\_Volume)  
 Volume++;  
 }  
  
 /\*\*  
 \* Method Simulate decreasing Volume by 1 in range[Min\_Volume,Max\_Volume]  
 \*/  
 public void ClickDownButton() {  
 if (Volume != Min\_Volume)  
 Volume--;  
  
 }  
  
 /\*\*  
 \* Method returns Value of Volume  
 \*  
 \* @return Volume volume  
 \*/  
 public int getVolume() {  
 return Volume;  
 }  
  
 /\*\*  
 \* Method set Volume in range[Min\_Volume,Max\_Volume]  
 \*  
 \* @param Vol the Volume value  
 \*/  
 public void setVolume(int Vol) {  
 if (Vol > MAX\_Volume) {  
 Volume = MAX\_Volume;  
 } else Volume = Math.max(Vol, Min\_Volume);  
 }  
  
}

**Приклад виконання програми:**



**Висновок:** На даній лаборатоній роботі я ознайомився з процесом розробки класів та пакетів мовою Java.