The instruction to the ‘KeyReader’

*KeyReader* is a program, that is needed to read symbols from keyboards.

To start using this program you should download it from the link (www.keyreader.org)

To run the *KeyReader* you should open a MVC.jar file. File directory is (MVC\dist).

When you open this program you will see one button (“Run”). If you press this button, the program will start. It will be reading all the key symbols if the amount of key symbols does not exceed 100 or until you press F12. After the program is closed, you can check the root folder to see the readed symbols. Log file’s name is “save”.

**Target Users**

You can use this program if:

* You forget the buttons position
* You want to play
* You want to reverse the String
* You want to see information about PC

**System environment**

Mac OS

Windows OS (All)

Unix OS

Code listing

import java.io.\*;

import org.jnativehook.GlobalScreen;

import org.jnativehook.keyboard.NativeKeyEvent;

import org.jnativehook.keyboard.NativeKeyListener;

public class Model2 implements NativeKeyListener{

String output = "";

String [] str = new String[100];

int i = 0;

public void main(){

try{

GlobalScreen.registerNativeHook();

}

catch(Exception e){

e.printStackTrace();

}

GlobalScreen.getInstance().addNativeKeyListener(new Model2());

}

@Override

public void nativeKeyPressed(NativeKeyEvent e) {

str[i] = NativeKeyEvent.getKeyText(e.getKeyCode());

i = i + 1;

if (e.getKeyCode() == NativeKeyEvent.VK\_F12 || i >= str.length) {

for (int j = 0; j < str.length; j++) {

if (str[j] != null) {

output += str[j] + " ";

}

}

try (FileWriter writer = new FileWriter("save.txt", false)) {

writer.write(output);

} catch (IOException ex) {

System.out.println(ex.getMessage());

}

}

}

@Override

public void nativeKeyReleased(NativeKeyEvent nke) {

}

@Override

public void nativeKeyTyped(NativeKeyEvent nke) {

}

}

public class View2 extends JFrame{

Model2 model2 = new Model2();

private JLabel additionLabel = new JLabel("Press Run to start keylogger");

private JButton run = new JButton("RUN");

private JLabel additionLabe2 = new JLabel("To finish press F12 or enter 1000 symbols");

View2(){

JPanel calcPanel = new JPanel();

this.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

this.setSize(385, 200);

this.setLocation(700, 300);

this.setTitle("KeyReader");

calcPanel.add(additionLabel);

calcPanel.add(run);

calcPanel.add(additionLabe2);

this.add(calcPanel);

}

public void set(){

model2.main();

}

void addRunListener(ActionListener listenForCalcButton){

run.addActionListener(listenForCalcButton);

}

void displayErrorMessage(String errorMessage){

JOptionPane.showMessageDialog(this, errorMessage);

}

}

public class Controller2 {

private Model2 theModel = new Model2();

private View2 theView = new View2();

public Controller2(View2 theView, Model2 theModel) {

this.theView = theView;

this.theModel = theModel;

this.theView.addRunListener(new Listener());

}

class Listener implements ActionListener{

public void actionPerformed(ActionEvent e) {

theView.set();

}

}

}

public class KeyReader {

public void runKey(){

Model2 model2 = new Model2();

View2 view2 = new View2();

Controller2 controller = new Controller2(view2, model2);

view2.setVisible(true);

}

}

ScreenShot

