**Event 2**

**-----Örnek 8 Window örneği. Pencerenin aktif olup olmadığını, ikon halinde olduğunu veya açık olduğunu yazan prog.**

**import** java.awt.Color;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**public** **class** TestWindowEvent **extends** JFrame {

**private** JTextArea t=**new** JTextArea();

**public** TestWindowEvent() { // Yapılandırıcı

add(t);

addWindowListener(**new** Dinle()); //Dinleyici atandı

}

**protected** **class** Dinle **implements** WindowListener { //Dinleyici class yazılıyor

**public** **void** windowDeiconified(WindowEvent event) {

t.append("Window ikondan açıldı\n"); // TextArea'ya yazı eklendi.

}

**public** **void** windowIconified(WindowEvent event) {

t.setForeground(Color.*BLACK*);

t.append("Window ikon haline getirildi\n");

}

**public** **void** windowActivated(WindowEvent event) {

t.append("Window aktif oldu\n");

}

**public** **void** windowDeactivated(WindowEvent event) {

t.append("Window aktif değil\n");

}

**public** **void** windowOpened(WindowEvent event) {

t.setForeground(Color.*RED*);

t.append("Window açıldı\n");

}

**public** **void** windowClosing(WindowEvent event) {

System.*out*.println("Window kapanıyor\n");

}

**public** **void** windowClosed(WindowEvent event) {

System.*out*.println("Window kapandı\n");

}

}

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

TestWindowEvent frame = **new** TestWindowEvent();

frame.setLocationRelativeTo(**null**);

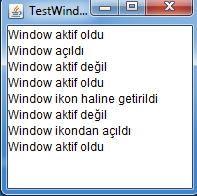
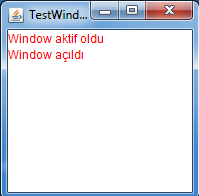
frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

frame.setTitle("TestWindowEvent");

frame.setSize(200, 200);

frame.setVisible(**true**);

}}



**--------------Örnek 9 Fare hareketleri. Fare tıklanıp sürüklenirken string’in hareket etmesi. (paint var)**

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**public** **class** FareDemo1 **extends** JPanel { // Panel bir mesaj yazar

**private** String mesaj ;

**private** **int** x = 20;

**private** **int** y = 20;

**public** FareDemo1(String s) { // s mesajını yazan bir panel

mesaj= s;

addMouseMotionListener(**new** Dinle());

}

**protected** **class** Dinle **implements** MouseMotionListener{

**public** **void** mouseDragged(MouseEvent e) { // Fare sürükleme hareketi durumunda yapılacak işlem

x = e.getX();// Yeni yeri ve rengi belirleyip paneli yeniden çiz

y = e.getY();

setForeground(Color.*BLUE*);

repaint();

}

**public** **void** mouseMoved(MouseEvent e) { // Fare hareketi durumunda yapılacak işlem

setForeground(Color.*RED*); // rengi kırmızı yapıp yeniden çiz

repaint();

}

}// iç class’ın sonu

**protected** **void** paintComponent(Graphics g) {

**super**.paintComponent(g);

g.drawString(mesaj, x, y);

}

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

FareDemo1 p = **new** FareDemo1("Java'ya hoşgeldiniz");

p.setLayout(**new** BorderLayout());

JFrame frame = **new** JFrame();

frame.setTitle("Fare hareketi Demo");

frame.setSize(400, 200);

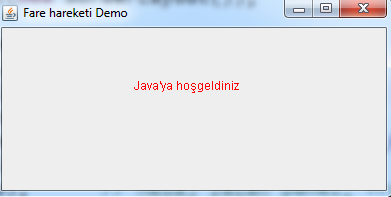
frame.setLocationRelativeTo(**null**);

frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

frame.setVisible(**true**);

frame.add(p); // Mesaj yazan panel, frame'e yapıştırıldı

}

}

**----------Örnek 10 Ekrandaki bir karakteri klavyeden hareket ettiren ve karakteri değiştiren program (paint var)**

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**public** **class** TusDemo **extends** JPanel {

**private** **int** x = 100;

**private** **int** y = 100;

**private** **char** harf = 'A'; // ilk değeri

**public** TusDemo() {

addKeyListener(**new** Dinle());

}

**protected** **class** Dinle **implements** KeyListener{

**public** **void** keyPressed(KeyEvent e) {

**switch** (e.getKeyCode()) {

**case** KeyEvent.*VK\_DOWN*: y += 10; **break**;

**case** KeyEvent.*VK\_UP*: y -= 10; **break**;

**case** KeyEvent.*VK\_LEFT*: x -= 10; **break**;

**case** KeyEvent.*VK\_RIGHT*: x += 10; **break**;

**default**: harf = e.getKeyChar(); // yeni değeri

}

repaint();

}

**public** **void** keyReleased(KeyEvent e) {}

**public** **void** keyTyped(KeyEvent e) {}

}

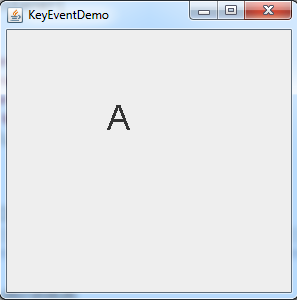
**protected** **void** paintComponent(Graphics g) { // Karakter’in çizilmesi

**super**.paintComponent(g);

g.setFont(**new** Font("TimesNewRoman", Font.*PLAIN*, 34));

g.drawString(String.*valueOf*(harf), x, y);

}

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

JFrame frame = **new** JFrame();

frame.setTitle("Tus Demo");

frame.setLocationRelativeTo(**null**);

frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

frame.setSize(300, 300);

frame.setVisible(**true**);

TusDemo klavyePanel = **new** TusDemo();

klavyePanel.setFocusable(**true**);

frame.add(klavyePanel);

}}

**Örnek 11: Ekrana bir resim basan ve o resmi tuşlarla hareket ettiren program. (paint var). Panel için setFocusable(true) metodu çağırılması gerekiyor.**

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**public** **class** TusResimDemo **extends** JPanel {

**private** **int** x = 100;

**private** **int** y = 100;

**private** ImageIcon resim1=**new** ImageIcon("resim/trGif.gif");

**private** Image resim=resim1.getImage();

**public** TusResimDemo() {

addKeyListener(**new** Dinle());

}

**public** **class** Dinle **implements** KeyListener{

**public** **void** keyPressed(KeyEvent e) {

**switch** (e.getKeyCode()) {

**case** KeyEvent.*VK\_DOWN*: y += 10; **break**;

**case** KeyEvent.*VK\_UP*: y -= 10; **break**;

**case** KeyEvent.*VK\_LEFT*: x -= 10; **break**;

**case** KeyEvent.*VK\_RIGHT*: x += 10; **break**;

}

repaint();

}

**public** **void** keyReleased(KeyEvent e) {}

**public** **void** keyTyped(KeyEvent e) {}

}

**protected** **void** paintComponent(Graphics g) {

**super**.paintComponent(g);

//g.setFont(**new** Font("TimesNewRoman", Font.*PLAIN*, 34));

**if**(resim!=**null**)

g.drawImage(resim, x, y,50,50,**this**);

}

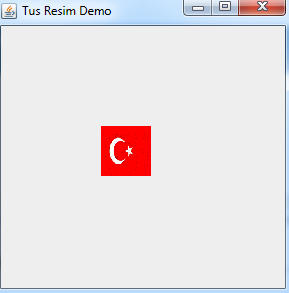
**public** **static** **void** main(String[] args) {

JFrame frame = **new** JFrame();

frame.setTitle("Tus Resim Demo");

frame.setLocationRelativeTo(**null**);

frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

 frame.setSize(300, 300);

frame.setVisible(**true**);

TusResimDemo klavyePanel = **new** TusResimDemo();

klavyePanel.setFocusable(**true**); // önemli

frame.add(klavyePanel);

}

}

**------------------ Örnek 12 Kronometre**

**import** javax.swing.JTextField;

**import** javax.swing.\*;

**import** javax.swing.Timer;

**import** java.awt.\*;

**import** java.awt.event.\*;

**public** **class** Kronometre **extends** JFrame{

**private** JTextField b1=**new** JTextField();

**private** **int** counter;

**private** Timer t;

**public** Kronometre(){

b1.setFont(**new** Font("TimesNewRoman",Font.*BOLD*,20));

b1.setHorizontalAlignment(JTextField.*CENTER*);

add(b1);

b1.addActionListener(**new** Dinle() );

counter=0;

t=**new** Timer(1000, **new** Dinle());

t.start();

}

**class** Dinle **implements** ActionListener{

**public** **void** actionPerformed(ActionEvent e){

counter++;

b1.setText(""+counter);

repaint();

}

}

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

Kronometre pencerem=**new** Kronometre();

pencerem.setTitle("Kronometre");

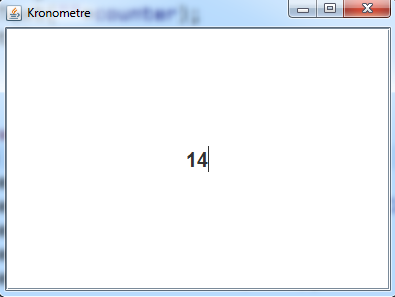
pencerem.setDefaultCloseOperation(*EXIT\_ON\_CLOSE*);

pencerem.setSize(400,300);

pencerem.setVisible(**true**);

pencerem.setLocationRelativeTo(**null**);

}

}

**------------Örnek 13 Daireyi buton ile küçültme ve büyültme**

**import** java.awt.\*;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** javax.swing.\*;

**public** **class** Daire **extends** JPanel{

**private** JButton buyuk=**new** JButton("Büyük");

**private** JButton kucuk=**new** JButton("Küçük");

**private** **int** yaricap;

**public** Daire(){

yaricap=5;

add(kucuk);

add(buyuk);

Dinle d=**new** Dinle();

buyuk.addActionListener(d);

kucuk.addActionListener(d);

}

**public** **class** Dinle **implements** ActionListener{

**public** **void** actionPerformed(ActionEvent e){

**if**(e.getSource()==buyuk)

yaricap++;

**else**

**if**(e.getSource()==kucuk)

yaricap--;

repaint();

}

}

**protected** **void** paintComponent(Graphics g){

**super**.paintComponent(g);

g.fillOval(100, 100, 2\*yaricap, 2\*yaricap);

}

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

JFrame pencere=**new** JFrame();

pencere.setTitle("Daire");

pencere.setSize(300,400);

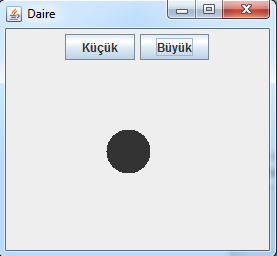
pencere.setVisible(**true**);

pencere.setLocationRelativeTo(**null**);

pencere.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

Daire p=**new** Daire(); // panel oluşturuluyor. (Butonlarla birlikte)

p.setFocusable(**true**); // önemli

 pencere.add(p);

}

}