**計時器**

import java.awt.Color;

import java.awt.Container;

import java.awt.FlowLayout;

import java.awt.Font;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.\*;

public class finaltest extends JFrame

implements ActionListener{

private static final long serialVersionUID = 1L;

private int ms,s,m,h;

private String s\_ms,s\_s,s\_m,s\_h;

private Timer timer;

private JLabel JL;

private JPanel JP;

public finaltest(){

super("計時器");

timer = new Timer (100,this);

timer.setInitialDelay(0);

Container c = getContentPane();

c.setLayout(new FlowLayout());

c.setBackground(Color.white);

JL = new JLabel(); //按履歷表概念，在JLabel裡面撕一張JL起來用

JP = new JPanel();

JL.setFont(new Font("標楷體",Font.PLAIN,20));

JP.add(JL);

c.add(JP);

timer.start();}

public void actionPerformed(ActionEvent arg0) {ms++;

if (ms<10){s\_ms = Integer.toString(ms);

//累計的數字轉字串}

else if(ms>10&&ms<99){

s\_ms = Integer.toString(ms);}

else{s++;ms=0;}

if (s<60){

s\_s = Integer.toString(s);}

else{m++;s=0;}

if (m<60){

s\_m = Integer.toString(m);}

else{h++;m=0;}

if(h<24){

s\_h = Integer.toString(h);}

JL.setText("時間:"+ s\_h +":"+ s\_m +":"+ s\_s +":"+ "0"+s\_ms); }

public static void main(String[] args)

{finaltest app = new finaltest();

app.setDefaultCloseOperation(DISPOSE\_ON\_CLOSE);

app.setSize(1000, 500);

app.setLocation(100, 100);

app.setVisible(true);

app.setTitle("計時器");}}

**賽馬**

import java.awt.BorderLayout;

import java.awt.Color;

import java.awt.Container;

import java.awt.Dimension;

import java.awt.Graphics;

import java.awt.Image;

import java.awt.Insets;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.\*;

public class Race\_exercise extends JFrame implements ActionListener {

private static final long serialVersionUID = 1L;

private AnimationPane animationPane;

//設定 圖檔來源

ImageIcon p1 = new ImageIcon("hours1.png");---p2;---

//Button 移動位置

int o1;---o2;---

//宣告 Timer

private Timer timer;

//設定 終點

int width = 735;

Container c = getContentPane();

Border bLine;

JButton bt1;---bt2;---

JPanel jp = new JPanel();

public Race\_exercise() {

//設定 Button 物件

bt1 = new JButton("粉紅小馬");---bt2---

timer = new Timer(200,this);

timer.setInitialDelay(0); //setInitialDelay = 0

c.setLayout(null);

c.setBackground(Color.white);

bLine = BorderFactory.createLineBorder(Color.blue);

JPanel jpane1 = new JPanel();

jpane1.setBounds(800, 1, 1, 500);

jpane1.setBorder(bLine);

c.add(jpane1, BorderLayout.EAST);

timer.start();//開始 timer

}

class AnimationPane extends JPanel{

Image image;

public AnimationPane(Image image) {

// 建構子

setPreferredSize(new Dimension(250, 100));

setBackground(Color.black);

this.image = image;

}}

public void actionPerformed(ActionEvent arg0) {

if ((o1) >= width) {

timer.stop(); // 結束timer

JOptionPane.showMessageDialog(jp, "粉紅小馬win~","\*\*比賽結果\*\*", JOptionPane.WARNING\_MESSAGE);}

else if ((o2) >= width) {----}

else {repaint();//重繪}

}

public void paint(Graphics g) { // 執行賽馬

Insets ins = getInsets(); // 取得邊線尺寸

int width = getWidth() - (ins.left + ins.right);

int height = getHeight() - (ins.top + ins.bottom);

g.setColor(Color.WHITE);

g.fillRect(ins.left, ins.top, width, height);

//隨機產生 - 移動距離

int rnd1 = (int)Math.ceil(Math.random()\*40+1); //隨機 1 ~ n , ex. 1-40:Math.random()---

//累加上次移動距離

o1 = o1 + rnd1;---o2

//設定 物件位置

bt1.setBounds(o1, 30, 65, 70); // 設定 X: 水平距離, Y:垂直距離

//設定 button 圖片

bt1.setIcon(p1);

c.add(bt1);

g.setColor(Color.blue);

g.drawLine(810, 0, 810, 700);//畫終點線

// 觀眾席

int number = (int) (Math.random() \* 4) + 1;

if (number == 1) {

g.setColor(Color.green);

g.drawString("加油!! 粉紅小馬!!", o1, getHeight() - 20);}

public static void main(String[] args) {

// TODO Auto-generated method stub

Race\_exercise app = new Race\_exercise();

app.setDefaultCloseOperation(DISPOSE\_ON\_CLOSE);

app.setSize(1000, 500);

app.setLocation(100, 100);

app.setVisible(true);

app.setTitle("賽彩虹小馬");

}}

**儲存檔案**

import java.io.\*;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

// 繼承JFrame類別

public class exp2 extends JFrame implements ActionListener{

private static final String NULL = null;

private JLabel lbl;

private JButton save, Calculate , Remove;

private int intValue,i=0;

// 建立擁有捲動軸的文字區域元件

final JTextArea area = new JTextArea(10,15);

JScrollPane scroll = new JScrollPane(area);

// 建立JFileChooser元件

final JFileChooser jfc = new JFileChooser();

public exp2(){ // 建構子

super("加總計算機");

Container c = getContentPane();

// 建立儲存檔案按鈕

JButton save = new JButton("儲存檔案");

save.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent evt) {try{

int output = jfc.showSaveDialog(exp2.this);

//儲存檔案對話框

if(output == JFileChooser.APPROVE\_OPTION){

File file = jfc.getSelectedFile(); //取得選取的檔案

BufferedWriter input = new

BufferedWriter(new

FileWriter(file.getAbsolutePath()+".txt"));

String lines[] = area.getText().split("\n");

for(int i = 0 ; i < lines.length ; i++){

input.write(lines[i]);

input.newLine();}

input.close();

area.append("\n已寫入檔案\n");

area.append("檔案名稱: "+file.getName() + "\n");

area.append("-------------------------\n");

area.append("1加到" + intValue + "的總和:" + i ); }}

catch(IOException ioe){ioe.printStackTrace();}}});

lbl = new JLabel("請輸入數值:");

Calculate = new JButton("計算");

Calculate.addActionListener(this);

Remove = new JButton("清除");

Remove.addActionListener(this);

JPanel jpane = new JPanel(); // 建立JPanel物件

jpane.add(save);

jpane.add(Calculate);

jpane.add(Remove);

c.add(lbl, BorderLayout.NORTH);

c.add(scroll, BorderLayout.CENTER);

c.add(jpane, BorderLayout.SOUTH);}

public void actionPerformed(ActionEvent evt) {if(evt.getSource() == Calculate){

int j = 1;

intValue = Integer.valueOf(area.getText());

while(j <= intValue){

i +=j; j++; }

area.append("\n1加到"+intValue+"的總和:"+i);}

if(evt.getSource() == Remove){

area.setText(null);}}

// 主程式

public static void main(String[] args) {

// 建立Swing應用程式

exp2 app = new exp2();

app.setDefaultCloseOperation(DISPOSE\_ON\_CLOSE);

app.setSize(300,200); // 設定視窗尺寸

app.setVisible(true); // 顯示視窗}}

**畫畫**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

// 繼承JFrame類別

public class exp1 extends JFrame {

// 建構子

public exp1() { super("JFrame畫布"); }

// 繪圖方法

public void paint(Graphics g) {

Insets ins = getInsets(); // 取得邊線尺寸

// 計算實際的繪圖尺寸

int width = getWidth() - (ins.left + ins.right);

int height = getHeight() - (ins.top + ins.bottom);

int a=130,b=40,r=20,R=40,L=80;

g.setColor(Color.white); // 填入背景色彩

g.fillRect(ins.left,ins.top,width,height);

g.setColor(Color.green); // 繪頭

g.fillOval(a,b,R,R);

// 主程式

public static void main(String[] args) {

exp1 app = new exp1();

app.setDefaultCloseOperation(DISPOSE\_ON\_CLOSE);

app.setSize(300,200); // 設定視窗尺寸

app.setVisible(true); // 顯示視窗}}

**執行緒**

class UserThread extends Thread {

// 執行緒類別, 繼承Thread類別

private int length;

// 建構子

public UserThread(int length, String name) {

super(name); //呼叫Thread裡的function

this.length = length;}

// 執行執行緒

public void run() {

int temp = 0;

for ( int i = 1; i <= length; i++ ) temp += i;

System.out.println("執行緒:"+Thread.currentThread() + " 總和 = " + temp+" 數量:"+Thread.activeCount() );

//Thread[main,5,main] 第一個main是執行緒名，5是指優先權，第二個main是指來自於哪個父類別，此處是來自main thread }}

public class ex1 { // 主類別

public static void main(String[] args) { // 主程式

System.out.println("執行緒: " + Thread.currentThread());

UserThread ut1 = new UserThread(5, "執行緒A"); // 建立執行緒物件

UserThread ut2 = new UserThread(10, "執行緒B");

ut1.start(); ut2.start(); // 啟動執行緒}}

**攝/華**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

// 繼承JFrame類別, 實作ActionListener介面

public class exp1 extends JFrame

implements ActionListener {

private JLabel lb1,lb2;

private JTextField Centigrade,Fahrenheit;

private JButton btn1,btn2;

public exp1() { // 建構子

super ("攝氏/華氏 轉換");

Container c = getContentPane();

c.setLayout(new FlowLayout());

c.setBackground(Color.blue);

Centigrade = new JTextField(12);// 建立文字方塊

Fahrenheit = new JTextField(12);

// 建立標籤

JLabel tLbl = new JLabel("攝氏 °C :");

tLbl.setLabelFor(Centigrade); // 屬於文字方塊的標籤

JLabel tLb2 = new JLabel("華氏 °F :");

tLb2.setLabelFor(Fahrenheit);

//顯示輸入文字內容的標籤

lb1 = new JLabel("");lb2 = new JLabel("");

btn1 = new JButton(">");

btn1.addActionListener(this);

btn2 = new JButton("<");

btn2.addActionListener(this);

// 註冊事件處理

JPanel jpane = new JPanel();

jpane.add(tLbl);

jpane.add(Centigrade);

jpane.add(btn1);

jpane.add(btn2);

jpane.add(tLb2);

jpane.add(Fahrenheit);

jpane.add(lb1);

jpane.add(lb2);

c.add(jpane); }

// 實作事件處理方法

public void actionPerformed(ActionEvent evt) {

if(evt.getSource() == btn1){

double intValue1 = Double.valueOf(Centigrade.getText());

Fahrenheit.setText("" + (intValue1\*9/5+32));}

if(evt.getSource() == btn2){

double intValue2 = Double.valueOf(Fahrenheit.getText());

Centigrade.setText("" + (intValue2-32)\*5/9);}}

// 主程式

public static void main(String[] args) {

Example2 app = new Example2();

app.setDefaultCloseOperation(DISPOSE\_ON\_CLOSE);

app.setSize(600,100); // 設定視窗尺寸

app.setVisible(true); // 顯示視窗}}

JLabel—簡易文字標籤

JTextField—輸入文字元件

JPasswordField—輸入後像密碼一樣

JTextField.setColumns(10)—10表欄位長度

JComboBox—下拉式選單

//JFrame jframe = new JFrame("視窗程式");  
jframe.setVisible(true);  
String[] option = new String[]{"選項一","選項二"};  
JComboBox jComboBox = new JComboBox(option);  
jframe.add(jComboBox);

宣告JComboBox需帶入一個String陣列做為選項的項目。取得使用者選取的選項方法如下：

int selected = jComboBox.getSelectedIndex();

JButton—按鈕

JPanel—群組的概念

JCheckBox—核取方塊

JComboBox下拉式清單元件是繼承自JComponent，它是一種選擇元件，不過，此元件只會顯示一個項目（目前選擇的選項），需按下旁邊的向下箭頭，才會拉出整張選單的選項

String[] items = { "程式語言", "計算機概論", "資料庫系統", "網頁設計"};

JComboBox list = new JComboBox(items);

對話框

showMessageDialog() 可顯示【確定】按鈕

JOptionPane.showMessageDialog(jpane, "一個測試的訊息視窗!");

showConfirmDialog() 可顯示詢問問題

int n = JOptionPane.showConfirmDialog(jpane,"您是否已經按下showMessageDialog按鈕? ","操作問題“,JOptionPane.YES\_NO\_OPTION);

showOptionDialog() 可顯示指定標題文字、圖示、訊息和按鈕

Object[] options = {"showMessageDialog按鈕", "showConfirmDialog按鈕"};

int m = JOptionPane.showOptionDialog(jpane,"哪一個按鈕顯示警告訊息?", "操作問題", JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE, null, options, options[0]);























