



# 程式語言 (Programming Language)

## Lec. 08 記事本程式整合說明

---



楊 吳 泉

義守大學資訊工程學系

web: <http://elearning.isu.edu.tw>

<http://audtm.net/~wcyang>

mail: [wcyang@isu.edu.tw](mailto:wcyang@isu.edu.tw)



# 綱要 (Outline)

---

- 程式規劃
- 程式架構
- 介面安排
- 檔案讀寫
- 功能擴充





# 程式規劃

## ➤ (觀念)

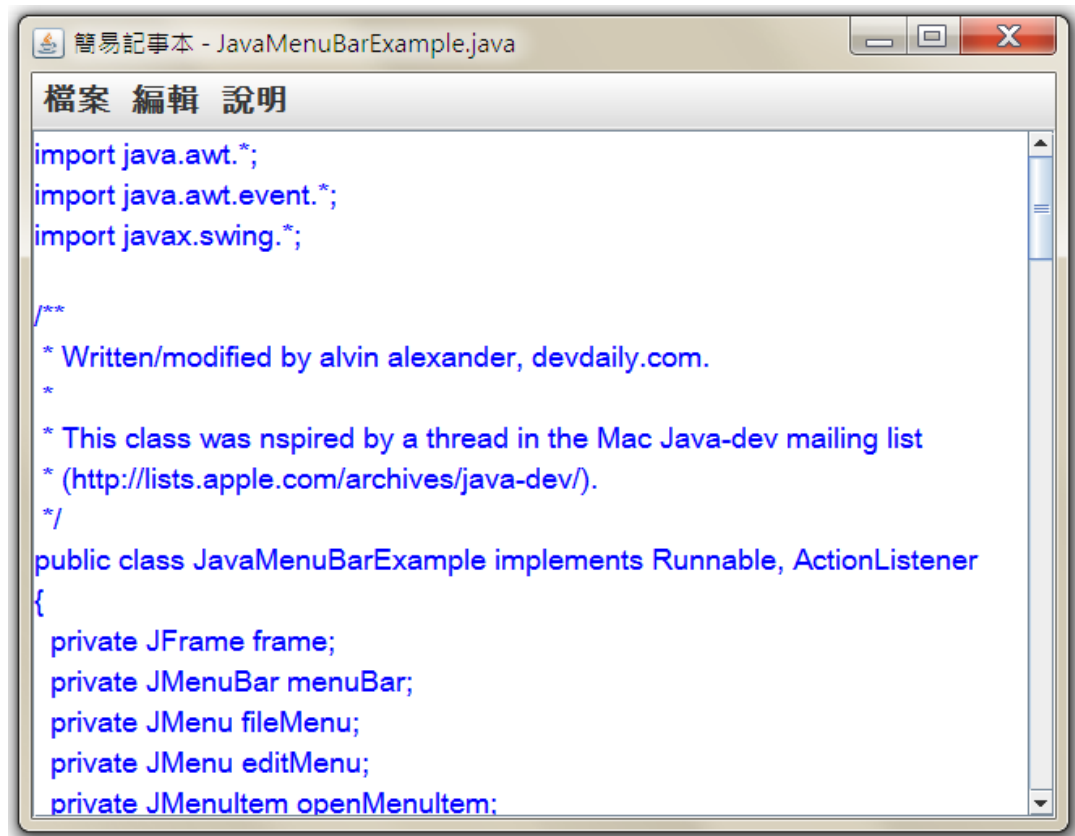
- ✓ 程式架構
- ✓ 物件觀念

## ➤ 功能

- ✓ 檔案
  - 開檔(新、舊)、存檔
- ✓ 編輯
  - 搜尋、取代
- ✓ 說明
  - 程式說明

## ➤ 外觀

- ✓ Menu bar規劃
- ✓ 文字區(需可規劃)



```
檔案 編輯 說明

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

/**
 * Written/modified by alvin alexander, devdaily.com.
 *
 * This class was inspired by a thread in the Mac Java-dev mailing list
 * (http://lists.apple.com/archives/java-dev/).
 */
public class JavaMenuBarExample implements Runnable, ActionListener
{
    private JFrame frame;
    private JMenuBar menuBar;
    private JMenu fileMenu;
    private JMenu editMenu;
    private JMenuItem openMenuItem;
```



# 程式架構

```
1 import java.awt.*;
2 import java.awt.event.*;
3 import javax.swing.*;
4 import java.io.*;
5
6 public class SimpleNotepad implements ActionListener {
7     JFrame fr = new JFrame("簡易記事本");
8     JMenuBar mb = new JMenuBar();
9     JMenu m1 = new JMenu("檔案");
10    JMenu m2 = new JMenu("編輯");
11    JMenu m3 = new JMenu("說明");
12    JMenuItem m11 = new JMenuItem("開新檔案");
13    JMenuItem m12 = new JMenuItem("開舊檔案");
14    JMenuItem m13 = new JMenuItem("儲存檔案");
15    JMenuItem m21 = new JMenuItem("搜尋...");
16    JMenuItem m22 = new JMenuItem("取代...");
17    JMenuItem m31 = new JMenuItem("關於本程式");
18    JTextArea ta = new JTextArea();
19    JScrollPane sp = new JScrollPane(ta);
20    FileDialog fd;
21    String fileDir, fileName; // =====
22    Font f = new Font("Arial Unicode MS", Font.BOLD, 20);
23    Font f1 = new Font("Arial Unicode MS", Font.PLAIN, 18);
24
25    public SimpleNotepad() {
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49    public void actionPerformed(ActionEvent e) {
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113    public static void main(String[] args) {
114
115
116
117
118 }
```

路徑宣告區  
(相關類別所在路徑)

使用者介面物件  
宣告及產生

程式建構子(Constructor)  
設定程式各物件初始狀態

按鍵動作設定  
對應menu item等介面  
元件之動作設定

程式啟動



# 程式觀念說明 I

---

## ➤ 圖形使用者介面(Graphic User Interface, GUI)

### ✓ Frame選擇

- `java.awt.Frame`
- `javax.swing.JFrame`

### ✓ JFrame使用方式

- 將程式宣告為JFrame之子類別，亦即程式本身即為frame
- 程式中使用JFrame

### ✓ 動作管控

- 滑鼠按下對等元件
  - 需實現ActionListener (內含一抽象方法: `actionPerformed`)
- 熱鍵(Hot key)



# 程式觀念說明 II

---

## ➤ 檔案存取技巧

### ✓ 檔案

- 路徑(目錄)與檔名是開啟檔案的要件
- 檔案本身也可是一個物件

### ✓ 開啟檔案

- java.awt.FileDialog
- javax.swing.JFileChooser

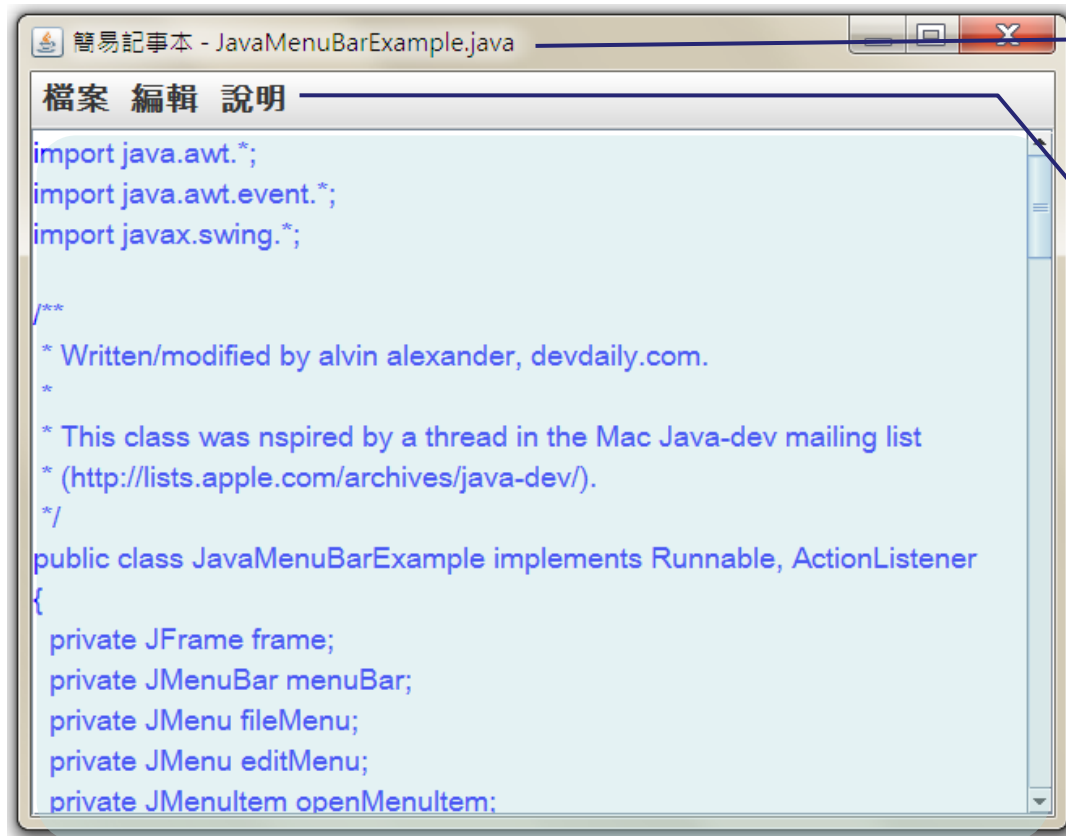
### ✓ 檔案讀取

- java.io.FileInputStream (一般檔案)
- java.io.FileReader (文字檔)

### ✓ 檔案寫入

- java.io.FileOutputStream (一般檔案)
- java.io.FileWriter (文字檔)

# 介面安排



Frame Title

Frame Menu Bar

Frame內容  
Layout預設值為  
BorderLayout  
TextArea預計擺中間



# 元件宣告

```
7      JFrame fr = new JFrame("簡易記事本");
8      JMenuBar mb = new JMenuBar();
9      JMenu m1 = new JMenu("檔案");
10     JMenu m2 = new JMenu("編輯");
11     JMenu m3 = new JMenu("說明");
12     JMenuItem m11 = new JMenuItem("開新檔案");
13     JMenuItem m12 = new JMenuItem("開舊檔案");
14     JMenuItem m13 = new JMenuItem("儲存檔案");
15     JMenuItem m21 = new JMenuItem("搜尋...");
16     JMenuItem m22 = new JMenuItem("取代...");
17     JMenuItem m31 = new JMenuItem("關於本程式");
18     JTextArea ta = new JTextArea();
19     JScrollPane sp = new JScrollPane(ta);
20     FileDialog fd;
21     String fileDir, fileName; // =====
22     Font f = new Font("Arial Unicode MS", Font.BOLD, 20);
23     Font f1 = new Font("Arial Unicode MS", Font.PLAIN, 18);
```

Menu bar 宣告

TextArea 宣告

File process 宣告，可改在  
actionPerformed處宣告

使用字型宣告





# 元件設定

```
25 public SimpleNotepad() {  
26     m1.add(m11); m1.add(m12); m1.add(m13); m1.setFont(f);  
27     m11.addActionListener(this); m11.setFont(f1);  
28     m12.addActionListener(this); m12.setFont(f1);  
29     m13.addActionListener(this); m13.setFont(f1);  
30  
31     m2.add(m21); m2.add(m22); m2.setFont(f);  
32     m21.addActionListener(this); m21.setFont(f1);  
33     m22.addActionListener(this); m22.setFont(f1);  
34  
35     m3.add(m31); m3.setFont(f);  
36     m31.addActionListener(this); m31.setFont(f1);  
37  
38     mb.add(m1); mb.add(m2); mb.add(m3);  
39     ta.setFont(f1);  
40     ta.setEditable(false);  
41     fr.setLocation(100, 100);  
42     fr.setSize(800,600);  
43     fr.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
44     fr.setJMenuBar(mb);  
45     fr.add(BorderLayout.CENTER,sp);  
46     fr.setVisible(true);  
47 }
```

Menu 設定

TextArea設定

Frame設定



# 檔案讀寫

```
57     if(e.getSource()==m12) {
58         fd = new FileDialog(fr,"開舊檔案",FileDialog.LOAD);
59         fd.setVisible(true);
60         ta.setText("");
61         ta.setEditable(true);
62         ta.setForeground(Color.blue);
63         fileDir = fd.getDirectory();
64         fileName = fd.getFile();
65         fr.setTitle("簡易記事本 - "+fileName);
66         try {
67             FileReader fin = new FileReader(fileDir+fileName);
68             BufferedReader in = new BufferedReader(fin);
69             String str;
70             while((str=in.readLine())!=null) ta.append(str+"\n");
71             fin.close();    in.close();
72         }
73         catch(IOException e1) {
74             ta.append("檔案處理錯誤!\n");
75         }
76     }
```

```
49     public void actionPerformed(ActionEvent e) {
50         if(e.getSource()==m11) {
51             ta.setText("");
52             ta.setEditable(true);
53             ta.setForeground(Color.blue);
54             fileName = new String("new");
55             fr.setTitle("簡易記事本 - "+fileName);
56         }
```

```
77     if(e.getSource()==m13) {
78         fd = new FileDialog(fr,"儲存檔案",FileDialog.SAVE);
79         fd.setVisible(true);
80         fileDir=fd.getDirectory();
81         fileName=fd.getFile();
82         try {
83             FileWriter fout = new FileWriter(fileDir+fileName);
84             String str = ta.getText();
85             fout.write(str);
86             fout.close();
87         }
88         catch(IOException e1) {
89             ta.append("檔案處理錯誤!\n");
90         }
91         fr.setTitle("簡易記事本 - "+fileName);
92     }
```

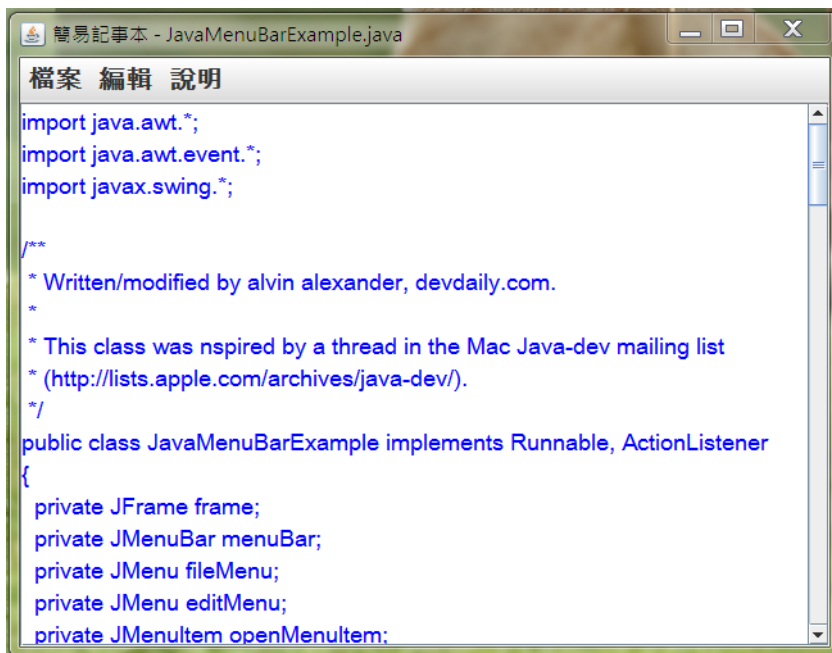


# 功能擴充 I

```
93         if(e.getSource()==m21) {
94             ta.setText("按下搜尋");
95         }
96         if(e.getSource()==m22) {
97             ta.setText("按下取代");
98         }
99         if(e.getSource()==m31) {
100             JDialog dl = new JDialog(fr,"關於簡單記事本");
101             JTextArea dlta = new JTextArea();
102             dlta.setEditable(false);
103             dlta.setFont(f);
104             dlta.setText("作者:義守大學\n日期:2013.05.14");
105             dlta.setBackground(Color.lightGray);
106             dl.setLocation(200, 200);
107             dl.setSize(200,100);
108             dl.add(dlta);
109             dl.setVisible(true);
110         }
111     }
112
113     public static void main(String[] args) {
114         SimpleNotepad demo = new SimpleNotepad();
115         demo.ta.append("開始執行...\n");
116     }
```

} 未完成

# 功能擴充 II



```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

/**
 * Written/modified by alvin alexander, devdaily.com.
 *
 * This class was inspired by a thread in the Mac Java-dev mailing list
 * (http://lists.apple.com/archives/java-dev/).
 */
public class JavaMenuBarExample implements Runnable, ActionListener
{
    private JFrame frame;
    private JMenuBar menuBar;
    private JMenu fileMenu;
    private JMenu editMenu;
    private JMenuItem openMenuItem;

```

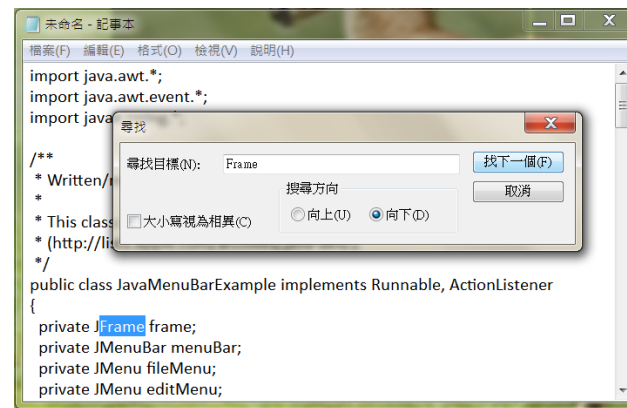


```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

/**
 * Written/modified by alvin alexander, devdaily.com.
 *
 * This class was inspired by a thread in the Mac Java-dev mailing list
 * (http://lists.apple.com/archives/java-dev/).
 */
public class JavaMenuBarExample implements Runnable,
ActionListener
{
    private JFrame frame;
    private JMenuBar menuBar;
    private JMenu fileMenu;

```



```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

/**
 * Written/modified by alvin alexander, devdaily.com.
 *
 * This class was inspired by a thread in the Mac Java-dev mailing list
 * (http://lists.apple.com/archives/java-dev/).
 */
public class JavaMenuBarExample implements Runnable, ActionListener
{
    private JFrame frame;
    private JMenuBar menuBar;
    private JMenu fileMenu;
    private JMenu editMenu;

```