

APPENDIX

F

Java Applet

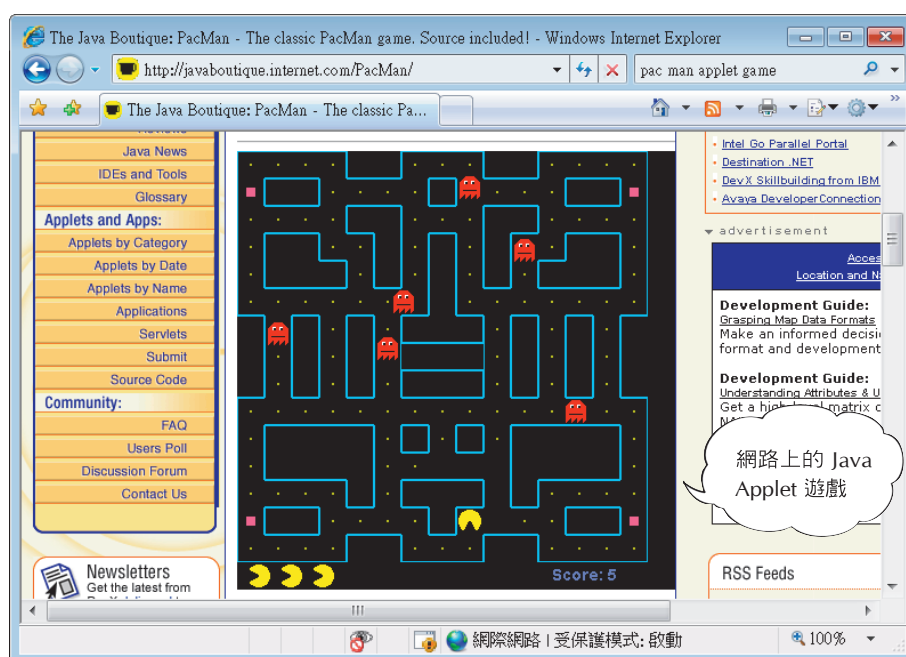
學習目標

- 認識 Java Applet
- 學習撰寫 Java Applet
- 學習 Java Applet 相關 HTML 語法
- 認識 Java Applet 的測試工具

大多數的電腦使用者，第一次接觸到 Java，通常是在瀏覽網頁時看到網頁內嵌的 Java Applet，甚至也有人因為看到有趣的 Java Applet 程式，而興起學習 Java 的念頭。本章就要來介紹什麼是 Java Applet、如何設計 Java Applet、如何將它們放到網頁之中。

F-1 甚麼是 Java Applet？

簡單的說，Java Applet 就是專門放在 HTML 網頁中的 Java 程式。Java Applet 雖然也是 Java 程式，但卻不能像本書前面所看到的 Java 程式一樣，可以從命令列啟動、執行，Java Applet 一定要放在 HTML 網頁中，並經由瀏覽器顯示、執行。



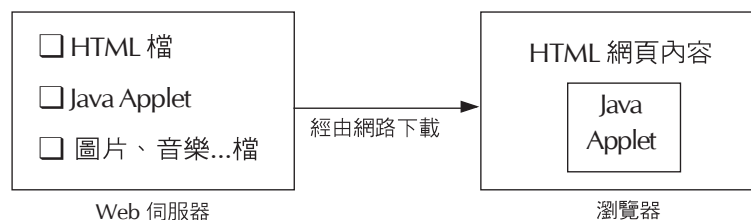
TIP 可在命令列執行的 Java 程式稱為 Java Application (Java 應用程式)。

F-1-1 Java Applet 架構

Java Applet 其實是具有 GUI 的特殊 Java 程式, 它和前面學過的 Java 應用程式有幾點明顯不同：

- Java Applet 一定是 GUI 程式, 而撰寫此種程式時, 需使用 AWT 的 Applet 或 Swing 的 JApplet 類別來設計。
- Java Applet 無法單獨執行, 必須用特定的 HTML 標籤將 Applet 嵌入 HTML 網頁之中。之後用瀏覽器載入此網頁時, 只要電腦上有 Java Runtime, 該 Applet 的 GUI 介面就會出現在網頁之中並執行。

由於 Java Applet 是放在網頁中執行的小程式, 所以其應用方式是將 Java Applet 及網頁放在 Web 伺服器上, 用戶端的 IE、FireFox 瀏覽器經由網路存取網頁時, 瀏覽器就會將內嵌於網頁的 Java Applet 也一併下載回來, 然後顯示在網頁中的指定位置。



由於 Java Applet 是應用於網路的環境, 而且是從伺服器下載到用戶端來執行, 因此基於安全考量, Java Applet 不能存取用戶端電腦上的資源, 最多只能取得一些如作業系統版本、瀏覽器版本等基本的資訊。這些限制都是由 JVM 來控制的, 當 JVM 查覺目前執行的是 Java Applet, 自然會加上各種存取限制, 就算 Java Applet 想讀取用戶端檔案, 也無法順利讀到, 並會引發例外。

TIP 應用於內部網站的 Java Applet, 可能需讓 Java Applet 具有存取用戶端檔案的功能, 此時可建立所謂的 Trusted Applet (受信任的 Java Applet), 即可順利存取用戶端的檔案。有興趣者可參閱 Java SDK 文件中的 keytool、jarsigner 工具的說明文件。

如前所述, 撰寫 Java Applet 需使用 AWT 的 Applet 或 Swing 的 JApplet 類別, 在此我們就沿續第 18 章的主題, 使用 Swing 的 JApplet 來開發 Java Applet。

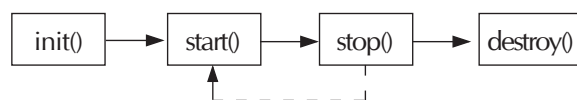
Java Applet 的四個特殊方法

從寫程式的角度看, 除了前述 Java Applet 一定要用 Applet/JApplet 類別來開發外, Java Applet 與一般 Java 應用程式還有一個很大的差異: Java Applet 沒有 main() 方法。

由於 Java Applet 無法單獨執行, 而是由瀏覽器載入執行的, 更詳細一點說: 瀏覽器載入 Java Applet 後, 會自行呼叫其建構方法來產生程式類別物件, 或者說是由瀏覽器呼叫 Java Applet。所以 Java Applet 不需像普通程式一樣有 main() 方法來擔任程式的啟動工作, 而是由 Applet/JApplet 類別提供下列 4 個方法, 供瀏覽器在不同的時期呼叫:

- **init() 方法**: 當 Java Applet 剛被載入時, 瀏覽器即會呼叫此方法。所以我們可以把它看成相當於 Java Applet 的 main() 方法, 有關 Java Applet 的初始化工作, 都可放在這個方法中。
- **start() 方法**: 當 Java Applet 所在的網頁被瀏覽器載入時, 就會呼叫此方法。
- **stop() 方法**: 當瀏覽器離開 Java Applet 所在的網頁時, 就會呼叫此方法。
- **destroy() 方法**: 當 Java Applet 要被結束時, 即呼叫此方法。

在整個 Java Applet 的執行生命週期中, 這 4 個方法被呼叫的順序如下:



其中 `init()`、`destroy()` 在整個生命週期中都只會被呼叫一次，但 `start()` 和 `stop()` 則可能會被呼叫多次。例如當使用者從含有 Java Applet 的網頁進入另一個網頁時，瀏覽器就會呼叫 `stop()` 方法請 Java Applet 停止運作；稍後使用者若又回到含有 Java Applet 的網頁（例如在瀏覽器按上一頁鈕），則瀏覽器就會呼叫 `start()` 方法請程式繼續動作。例如我們若是用 Java Applet 製作動畫，就可在 `stop()` 方法中暫停動畫，而在 `start()` 方法中開始動畫。

雖然 Applet/JApplet 類別有上述這 4 個特別的方法，但也不見得所有的 Java Applet 程式都需要實作這 4 個方法。例如只是單純繪圖的程式，也可以只有改寫 `paint()` 或 `paintComponent()` 方法。以下的範例程式也不會實作全部的 4 個方法。

F-1-2 測試 Java Applet 的工具

由於 Java Applet 是放在網頁中，並由瀏覽器瀏覽網頁時才會執行，所以在開發的過程中，也需在相同的環境下進行測試。因此我們可將 Java Applet 連同網頁放在 Web 伺服器的網站目錄中，然後用瀏覽器瀏覽該網頁，才會看到 Java Applet 的執行畫面。

不過每次測試時都要使用瀏覽器實在有些不便，因此 JDK 提供了一個簡單的 Java Applet 測試工具：`appletviewer`，其用法相當簡單，只要以內嵌 Java Applet 的網頁檔案名稱為參數，執行 `appletviewer` 即可。例如：

```
appletviewer MyApplet.html
```

此時就會出現一個視窗，顯示網頁的內容，只要網頁中的 HTML 碼也都正確，就會看到 Java Applet 的執行情形：



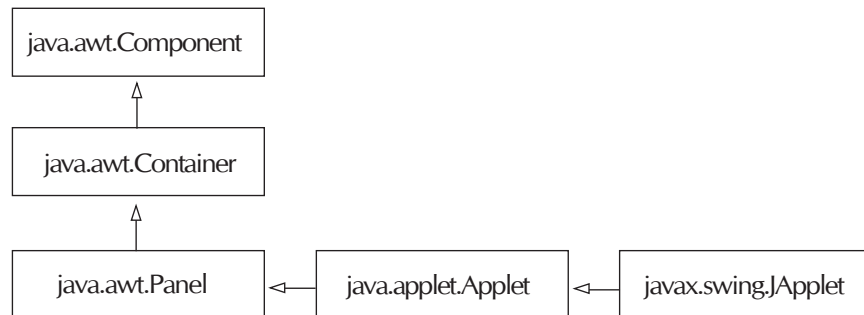
稍後我們就會以 `appletviewer` 來測試撰寫好的 Java Applet 及網頁。

F-2 最簡單的 Java Applet 程式

要撰寫 Java Applet 其實不困難, 首先是 GUI 的部份, 在第 18 章我們已學過 Swing 的容器類別 `JFrame`, 而寫 Java Applet 時, 只要把它換成 `JApplet` 即可, 也就是把 `JApplet` 當成一個上層的容器, 只不過這個容器沒有獨立的視窗, 而是內嵌於瀏覽器視窗中執行。

F-2-1 撰寫 Java Applet 程式

`JApplet` 雖然也是 Swing 類別, 但其繼承關係和 `JFrame` 等類別不同, 它是 AWT 中 `Applet` 類別的子類別, 如下圖所示：



TIP 雖然一般也將 Applet 類別視為 AWT 中的成員，不過 Applet 類別是歸於 java.applet 套件，而非 java.awt 套件。

JApplet 和 JFrame 在應用上也有個明顯的差異：由於 Applet 並無獨立的應用程式視窗，而且 Java Applet 顯示在網頁中的大小是由 HTML 標籤控制，所以我們不必再於 JApplet 中去設定視窗大小、或設定視窗關閉行為、顯示視窗等等。

除了換用 JApplet 類別外，其它有關 GUI 的部份，都仍可繼續使用上一章所用的技巧和方法，例如 JApplet 也和 JFrame 一樣有 Content Pane 的觀念、一樣預設使用 BorderLayout，其它像是 Swing 元件、事件處理、版面配置、甚至於 2D 繪圖等，也都依前一章的方式來應用即可。

以下我們就將前一章使用 JFrame 所設計的程式 SimpleListener.java，做局部的修改，就能讓它變成 Java Applet 程式。如前所述，Java Applet 不需 main() 方法、不需理睬視窗，所以我們將程式原先繼承的 JFrame 類別改成 JApplet 類別，並刪除相關的視窗處理敘述：

程式 ButtonApplet.java 計算按鈕次數的 Java Applet

```
01 import javax.swing.*;
02 import java.awt.event.*;
03
04 public class ButtonApplet extends JApplet
05     implements ActionListener {
06     int act = 0;        // 用來記錄按鈕被次數的變數
07     JButton mybutton = new JButton("換個文字");
08
09     // 用建構方法來建立元件、將元件加入視窗、顯示視窗
10     public ButtonApplet() {
11         mybutton.addActionListener(this);
12         getContentPane().add(mybutton);
13     }
14
15     public void actionPerformed(ActionEvent e) {
16         act++;        // 將按鈕次數加 1
17         mybutton.setText("發生 " + act + " 次按鈕事件");
18     }
19 }
```

1. 第 4 行宣告的 ButtonApplet 為 JApplet 的子類別。
2. 第 12 行將按鈕元件加到 Content Pane 中。
3. 第 15 ~ 18 行的按鈕事件處理方法和前一章的範例程式略有不同，因為 Applet 沒有視窗，因此也沒有視窗標題欄，所以我們將按鈕次數的文字改成顯示在按鈕上面。

請先以平常編譯程式的方式來編譯此 .java 檔，編譯出來的 .class 檔無法單獨執行，所以我們要另外寫一個網頁來放置這個 Java Applet，下面就來看要如何將 Java Applet 放到網頁之中。

F-2-2 撰寫載入 Java Applet 程式的網頁

要將 Java Applet 內嵌至 HTML 網頁中, 可使用以下 3 種標籤:

- **<object> 標籤**: 目前 HTML 標準建議使用的萬用型標籤, 除可用於嵌入 Java Applet 外, 其它如圖形、音效等物件也都可利用此標籤來嵌入網頁中。
- **<applet> 標籤**: 最早被提出來用於嵌入 Java Applet 的標籤, 但在最新的 HTML 標準中已建議不要使用, 不過目前的瀏覽器仍都有支援此標籤。
- **<embed> 標籤**: 雖然目前的瀏覽器多有支援, 但並非 HTML 標準, 所以建議不要使用。

雖然這 3 個標籤在 HTML 標準中的地位不同, 但在現實世界的使用卻無太大的分別, 因為主流的瀏覽器都支援這 3 種標籤的使用。為方便說明, 本章以介紹 <applet> 為主。

TIP 本附錄只說明與 Applet 有關的 HTML 標籤, 至於詳細的 HTML 網頁編輯和語法, 請參見其它相關書籍。

<applet> 標籤的用法很簡單, 基本上只需在標籤中用其屬性設定 Java Applet 的相關資訊, 接著再將標籤放在網頁中的適當位置。與 Java Applet 相關的屬性包括:

- **code**: 指定 Java Applet 類別檔 (原始碼編譯後產生的 .class 檔) 的檔案名稱。
- **width**: 指定 Java Applet 顯示於網頁中的寬度 (像素)。
- **height**: 指定 Java Applet 顯示於網頁中的高度 (像素)。
- **align**: 指定 Java Applet 在網頁中与其它文字或物件的對齊方式。

- **alt**：指定替代性的顯示文字, 當瀏覽器無法顯示 Java Applet 時 (例如電腦未安裝 JDK 或 JRE), 就會在應顯示 Java Applet 的地方顯示此段訊息。
- **codebase**：指定 Java Applet 類別檔的存放路徑 (URL), 若未註明此項, 表示類別檔與 HTML 檔是放在同一個位置。
- **name**：指定一個名稱給 Java Applet。
- **hspace**：指定 Java Applet 與左、右文字 (或其它物件) 的間距。
- **vspace**：指定 Java Applet 與上、下文字 (或其它物件) 的間距。

雖然可用的屬性很多, 但必要的只有 3 個, 只需指定類別檔名、以及 Java Applet 在網頁中的寬度與高度這三項資訊即可, 例如：

```
<applet code="AppletButton.class"
        width="200"
        height="160">
此處可放置一些說明文字, 供不支援 applet 標籤的瀏覽器顯示文字訊息
</applet>
```

另外, 在 `<applet>` 和 `</applet>` 標籤之間, 可以用 `<param>` 標籤傳遞參數給 Java Applet 使用, 在稍後 F-2-5 節, 就會示範此種用法。

以下就是一個陽春的網頁, 其中的 `<applet>` 標籤只用最基本的 3 個屬性來指定顯示 `ButtonApplet.class` 的方式：

程式 ButtonApplet.html 計算按鈕次數的 Java Applet

```

01 <html>
02 <head><title>只有一個按鈕的Applet</title>
03 <meta http-equiv="Content-Type" content="text/html; charset=Big5">
    </head>
04 <body>
05 使用 applet 標籤嵌入 Java Applet
06 <applet code="ButtonApplet.class"
07     width="200" height="160">
08 您的瀏覽器不支援 applet 標籤!
09 </applet>
10 </body></html>

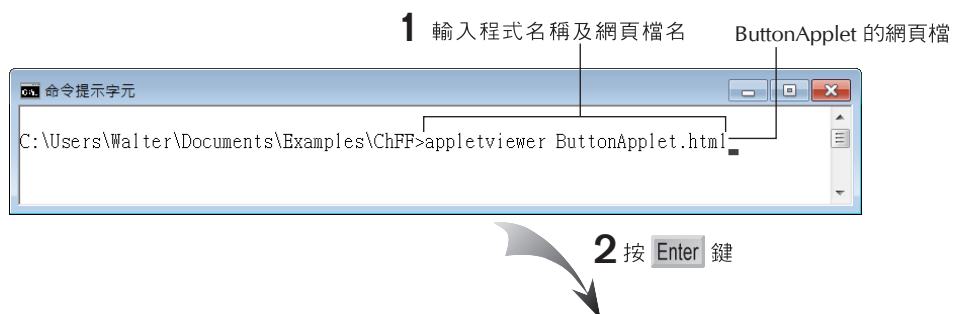
```

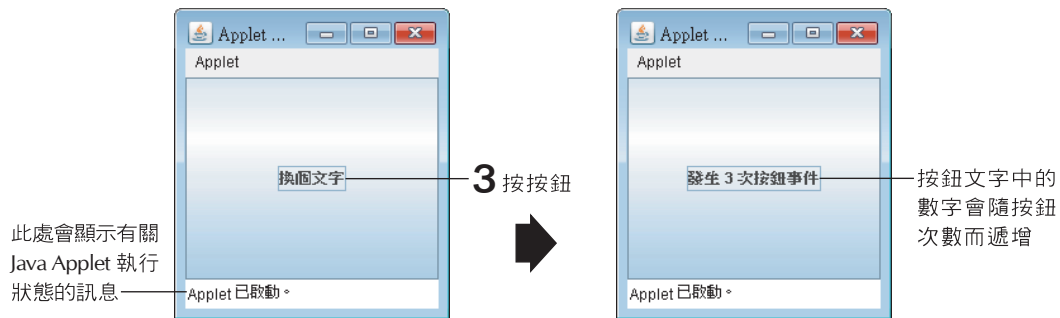
因為 `<applet>` 標籤的應用很簡單，所以稍後就不列出各範例程式所用的 HTML 檔，只會註明該程式所用的 HTML 檔名，檔案內容請直接參考書附光碟中的檔案。

TIP `<object>` 標籤的語法和 `</applet>` 有很多相似之處，例如將上列程式中的 `<applet>`、`</applet>` 這一對標籤直接換成 `<object>`、`</object>` 也仍可正常運作。

F-2-3 以 JDK 測試 Java Applet

編譯好 Applet 及完成網頁後，即可開始測試 Java Applet 了。要用 `appletviewer` 來測試編譯好的 Java Applet 及撰寫好的網頁，只需以網頁為參數來執行 `appletviewer` 即可：

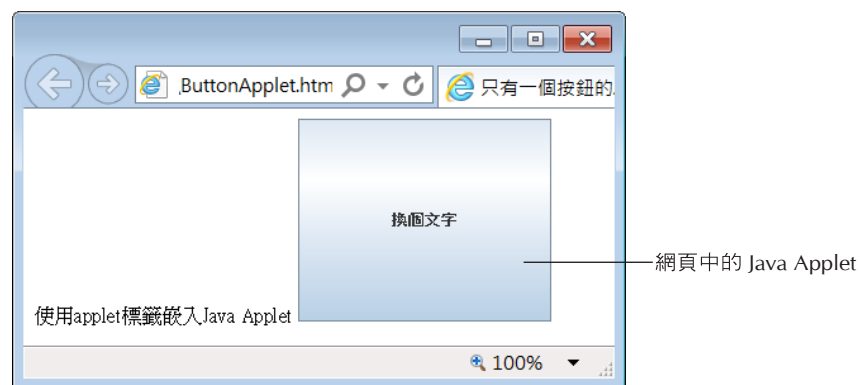




appletviewer 雖然也是載入網頁來執行 Java Applet, 但它並不會顯示網頁的其它內容, 只會顯示 Java Applet 的部份。而且若我們調整 appletviewer 顯示的視窗大小, 則 Java Applet 的大小也會隨之調整, 不會受 <applet> 標籤中指定寬高的限制。

F-2-4 以瀏覽器瀏覽 Java Applet

此外您也可打開瀏覽器, 直接瀏覽包含 Java Applet 的網頁, 實驗看看網頁的外觀, 及 Java Applet 在網頁中執行的狀態。以 Windows 各作業系統為例, 直接在**檔案總管**雙按網頁檔案, 就會以瀏覽器 (預設為 IE) 開啓這個 HTML 檔：



用瀏覽器顯示 Java Applet 時, 其大小就是固定為 <applet> 標籤中指定的大小, 不會隨瀏覽器視窗大小變化。

F-2-5 傳遞參數給 Java Applet

有時候我們可能會設計供多個網頁共用的 Java Applet，但每個網頁使用的方式會略有差異，此時就可利用在 `<applet>` 中以 `<param>` 設定參數的方式，讓 Java Applet 可取得參數，進而決定其執行方式。

在 `<param>` 標籤中，要用 `name` 屬性指出參數名稱，並用 `value` 屬性指定參數值；而在 Java Applet 程式中，則可用 `getParameter("name")` 的方式取得指定的參數，傳回值即為代表參數值的 `String` 物件。

```
HTML 檔案中： <applet ...>
                <param name="filename" value="cloud.jpg">
                ...
            </applet>

Java 程式中： String para = getParameter("filename")
                // para 的值為 "cloud.jpg"
```

為示範這種用法，我們仍是修改前一章的 `ShowImage.java` 範例程式，將它修改為 Java Applet 版本，並從 HTML 檔中的參數決定所要顯示的影像檔名稱。另外我們也改用 `Applet/JApplet` 類別內建的 `getImage()` 方法來取得影像：

```
Image getImage(URL url)                // 取得 url 所指的影像
Image getImage(URL url, String name)    // 取得 url 中檔名為 name 的影像
```

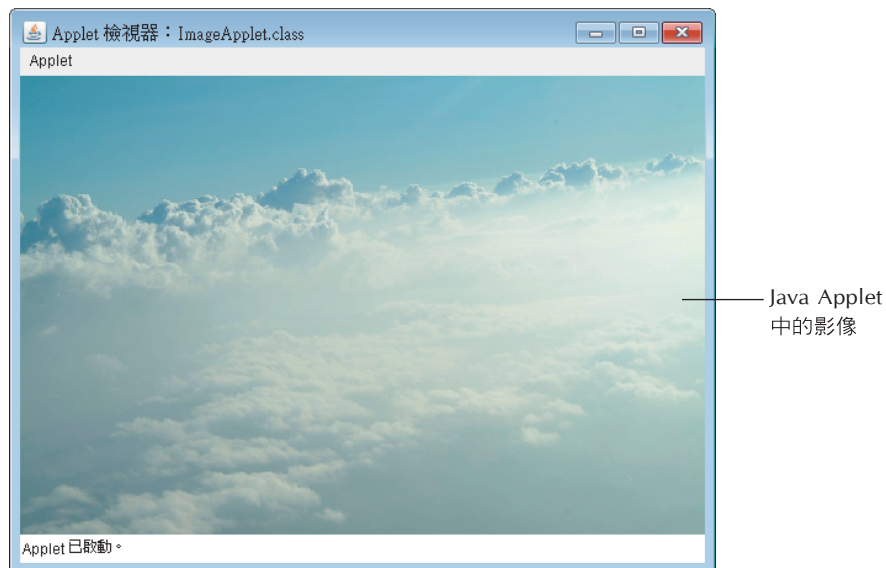
URL 是 `java.net` 套件中的類別，用來代表 URL (Uniform Resource Locator) 物件，只要以網址字串為參數，即可建構一個 URL 物件。或呼叫 `JApplet` 的 `getDocumentBase()` 方法，也會傳回 Java Applet 本身所在的 URL 物件。

以下就是使用 `Applet/JApplet` 的 `getImage()` 方法取得並顯示影像的範例：

程式 ImageApplet.java 顯示影像的 Java Applet

```
01 import java.awt.*;
02 import java.awt.geom.*;
03 import javax.swing.*;
04
05 class ShowImage extends JPanel {
06     Image img; // 代表影像的物件
07
08     public ShowImage(Image i) {
09         img = i;
10     }
11
12     public void paintComponent(Graphics g) {
13         super.paintComponent(g);
14         Graphics2D g2 = (Graphics2D) g;
15
16         AffineTransform at = new AffineTransform();
17         Dimension d = getSize();
18         double sc = Math.min(d.width/(double)img.getWidth(null),
19                             d.height/(double)img.getHeight(null));
20         at.scale(sc,sc);
21         g2.drawImage(img, at, this);
22     }
23 }
24
25 public class ImageApplet extends JApplet {
26     public void init () {
27         // 從網頁中取得影像檔名參數
28         Image i = getImage(getDocumentBase(), getParameter("fname"));
29
30         // 建立 Panel 物件, 並加入 Content Pane 中
31         ShowImage p = new ShowImage(i);
32         getContentPane().add(p);
33     }
34 }
```

執行結果



- 第 5 ~ 23 行的 ShowImage 類別, 其主要內容和前一章的範例 ShowImage.java 相同, 不過建構方法改成不是自己載入影像, 而是由呼叫者傳入影像物件。
- 第 28 行先用 getParameter() 取得 "fname" 參數的值, 也就是影像檔名稱, 然後再用 getImage() 方法取得影像。

F-3 綜合演練

F-3-1 由 Java Applet 開啓新視窗

前面的範例程式都是將舊的 GUI 程式移植成 Java Applet 放到網頁上使用, 採取的移植方式則是視原程式的內容, 將 JFrame 類別換成 JApplet、或是改將 JPanel 加到 JApplet 的 Content Pane 中。

此外將原先寫好的 Java GUI 程式移植到網路上還有一種特別的作法：就是讓 Java Applet 直接開啓新視窗，而此新視窗就是原先寫好的 GUI 視窗程式。

從 Java Applet 開啓 JFrame 或其它視窗，這些視窗仍受 Java Applet 本身的安全性限制，也就是不能存取用戶端的資源，另外 JFrame 有關整體程式運作的一些操作也需去除，例如上一章常用 setDefaultCloseOperation() 方法來設定關閉視窗時，即結束整個程式，此運作就不適用於 Java Applet 開啓的視窗，因為 Java Applet 的結束是由瀏覽器控制，而不能從我們的程式控制。

以下就是改寫第 18 章的溫度轉換程式，讓它變成是由 Java Applet 中開啓計算視窗的程式：

程式 TempApplet.java 開啓新視窗的 Java Applet

```
01 import java.awt.*;
02 import java.awt.event.*;
03 import javax.swing.*;
04
05 class TempConverter extends JFrame
06                     implements ActionListener {
07     // 轉換結果顯示區
08     JLabel result = new JLabel(" ", SwingConstants.CENTER);
09     JTextField degree = new JTextField(); // 輸入區
10     JButton f2c = new JButton("華氏轉攝氏");
11     JButton c2f = new JButton("攝氏轉華氏");
12
13     public TempConverter () {
14         Container contentPane = getContentPane();
15
16         // 將 5 個元件加到 BorderLayout 的五個位置
17         contentPane.add(new JLabel("請輸入溫度", SwingConstants.CENTER),
18                         BorderLayout.NORTH);
19         contentPane.add(f2c, BorderLayout.EAST);
20         contentPane.add(c2f, BorderLayout.WEST);
21         contentPane.add(degree, BorderLayout.CENTER);
22         contentPane.add(result, BorderLayout.SOUTH);
```



```

23
24     // 設定 this 物件為傾聽者
25     f2c.addActionListener(this);
26     c2f.addActionListener(this);
27
28     setTitle(" 華氏與攝氏溫度的轉換 ");
29     setSize(400,120);
30 }
31
32 public void actionPerformed(ActionEvent e) {
33     try {
34         // 取得輸入區的字串，並轉成浮點數
35         double value = Double.parseDouble(degree.getText());
36
37         String msg="";        // 顯示轉換結果的字串
38         if(e.getSource() == f2c) // 依按鈕決定轉換方式
39             msg= "華氏 " + value + " 度等於攝氏 " +
40                 ((value-32)*5/9) + " 度";
41         else
42             msg= "攝氏 " + value + " 度等於華氏 " +
43                 (value/5*9 + 32) + " 度";
44         // 並將結果寫到視窗最下方
45         result.setText(msg);
46     } catch (NumberFormatException ne) {
47         degree.setText("");    // 發生例外時清除輸入區內容
48     }
49 }
50 }
51
52 public class TempApplet extends JApplet
53     implements ActionListener {
54     TempConverter tc = new TempConverter();
55
56     public void init() {
57         JButton showButton = new JButton("溫度計算器");
58         getContentPane().add(showButton);
59         showButton.addActionListener(this);
60     }
61

```

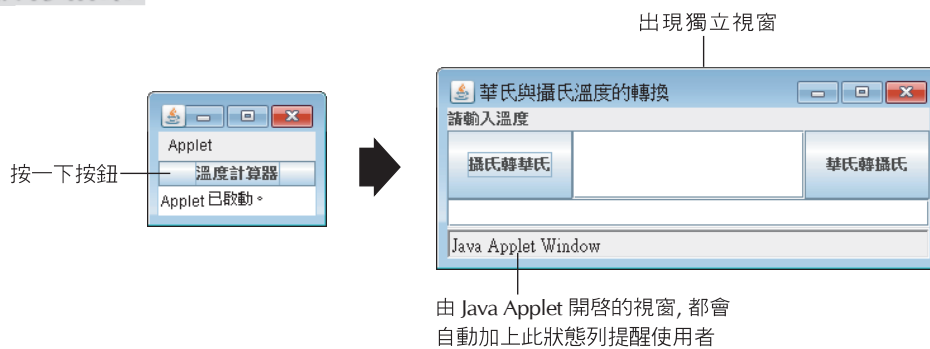
接下頁

```

62 public void actionPerformed(ActionEvent e) {
63     if (!tc.isVisible())
64         tc.setVisible(true);    //顯示視窗
65 }
66 }

```

執行結果



1. 第 5 ~ 50 行的 TempConverter 類別繼承 JFrame, 其內容和前一章的範例 TempConverter.java 大半相同, 不過去掉了呼叫 setDefaultCloseOperation()、setVisible() 方法的敘述。
2. 第 52 ~ 66 行為本程式的 Java Applet 類別定義, 此類別有個 TempConverter 成員。
3. 第 56 ~ 60 行的 init() 方法在 Java Applet 中加入一個用來顯示視窗的按鈕, 並設定按鈕事件傾聽者為 this 物件。
4. 第 62 ~ 65 行為按鈕事件處理方法, 此方法會檢查視窗是否已顯示, 尚未顯示就用 setVisible() 方法顯示視窗。

F-3-2 身兼 Java Application 與 Java Applet 的程式

雖然獨立執行的 Java Application 和放在網頁中的 Java Applet 本質上有很大的差異，但其實只要透過一些撰寫程式的技巧，我們就能讓自己寫的程式，不但可獨立執行，也可放在網頁中當成 Java Applet 使用。

這個技巧很簡單，就是讓繼承 JApplet 的類別也有個 main() 方法，並在其中建立 JFrame 物件，再將 GUI 的內容加到 JFrame。而在 Java Applet 應有的 init() 方法中，也是將相同的 GUI 內容加到 Java Applet 的 Content Pane 中。請參見以下這個簡單的例子：

程式 DrawShape.java 兩用的 Java 程式

```
01 import java.awt.*;
02 import java.awt.geom.*;
03 import javax.swing.*;
04
05 class MyPanel extends JPanel {
06     public void paintComponent(Graphics g) {
07         super.paintComponent(g);
08
09         double Width = getSize().width - 10;
10         double Height = getSize().height/2 - 10;
11
12         Graphics2D g2 = (Graphics2D) g;
13         g2.setPaint(Color.red);
14         g2.fill(new
15             Rectangle2D.Double(5,10,Width,Height));
16
17         g2.setPaint(new Color(64,128,64));
18         g2.fill(new
19             RoundRectangle2D.Double(5,15+Height,Width,Height,20,30));
20     }
21 }
22
23 public class DrawShape extends JApplet {
24     public void init() {
```

[接下頁](#)

```

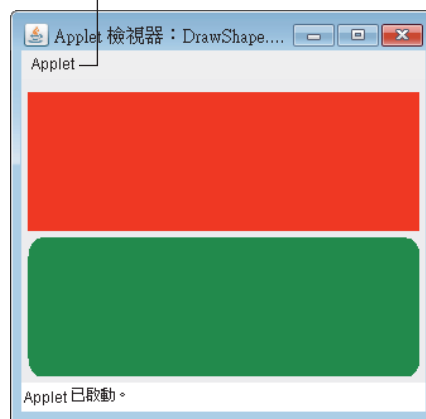
25     getContentPane().add(new MyPanel());
26 }
27
28 public static void main(String[] args) {
29     JFrame f = new JFrame("幾何圖案");
30     f.getContentPane().add(new MyPanel());
31     f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
32     f.setSize(320,240);
33     f.setVisible(true);
34 }
35 }

```

獨立執行的外觀



以 Java Applet 的形式放到網頁中的外觀



1. 第 5 ~ 21 行的 `MyPanel` 類別會在面板中畫兩種顏色的矩形及圓角矩形。
2. 第 24 ~ 26 行 Java Applet 的 `init()` 方法, 此方法直接將新建的 `MyPanel` 物件加到 Java Applet 中。
3. 第 28 ~ 34 行為單獨執行時所用的 `main()` 方法, 也就是建立一個 `JFrame` 物件, 並將自行建立的 `MyPanel` 物件加到其中。

利用這樣的技巧, 您就可將一些實用的應用程式, 改成可單獨執行或當成 Java Applet 用的多用途版本。

學習評量

1. Java 用來設計 Java Applet 的類別有兩個, 分別是 AWT 中的 _____ 類別及 Swing 中的 _____ 類別。
2. 下列何者不屬於 Applet/JApplet 在程式啟動 / 結束時期, 會執行到的四大方法 ?
 - (a) main()
 - (b) init()
 - (c) start()
 - (d) stop()
3. 下列關於使用網頁嵌入 Java Applet 的描述何者錯誤 ?
 - (a) 可用 <applet> 標籤嵌入 Java Applet 。
 - (b) 需指定 Java Applet 的原始檔檔名。
 - (c) 可用 <param> 標籤設定參數。
 - (d) 需在標籤中設定 Java Applet 的大小。
4. 要在網頁中嵌入 Java Applet, 可使用哪一個標籤 ?
 - (a) <applet>
 - (b) <embed>
 - (c)
 - (d) <object>

5. JApplet 預設使用何種配置管理員？
- (a) FlowLayout
 - (b) BorderLayout
 - (c) GridLayout
 - (d) CardLayout
6. 以下關於 Java Applet 安全性的描述何者錯誤？
- (a) Java Applet 可讀取伺服器端的檔案。
 - (b) Java Applet 可讀取用戶端的檔案。
 - (c) Java Applet 可寫入用戶端的檔案。
 - (d) Java Applet 可取得作業系統版本資訊。
7. 下列關於 JApplet 類別的描述何者錯誤？
- (a) JApplet 是容器類別。
 - (b) JApplet 有提供 getImage() 和 drawImage() 方法可載入和顯示影像。
 - (c) 程式可在 JApplet 的 init() 方法中做初始化的動作。
 - (d) JApplet 可用 setTitle() 方法設定瀏覽器標題欄。
8. 在整個 Java Applet 生命週期中, 只會被呼叫 1 次的 Applet/JApplet 內建方法是 _____ 及 _____。
9. 以下關於 JApplet 的敘述何者錯誤。
- (a) JApplet 的 init() 方法是在 Applet 載入網頁時被呼叫。
 - (b) JApplet 和 JFrame 一樣都有 Content Pane, 兩者都是 java.awt.Frame 的子類別。
 - (c) 在 JApplet 中也可加入 JPanel 物件。
 - (d) 從 JApplet 可開啓額外的 JFrame 視窗。

10. 請問下列程式片段有何問題？

```
public class TestApplet extends JApplet {  
    public void init () {  
        ...  
        JPanel p = new JPanel();  
        JButton b = new JButton("Testing");  
        p.add(b);  
        add(p);  
        ...  
    }  
}
```

程式練習

1. 請用第 18 章介紹的 Java 2D API 在 Java Applet 中繪製 2D 圖形。
2. 請練習設計一個簡單的 Java Applet, 每次用瀏覽器檢視時, 都會顯示不同顏色的圓形。
3. 承上題, 將程式改成每過一段時間, 就自動換色。(提示: 可使用 Swing 中的 Timer 類別建立計時器物件, 並用元件的 repaint() 方法進行重畫)
4. 請試改寫第 18 章的 TrigonoCalc.java 三角函數計算器, 使其變成 Java Applet 版本。
5. 請用 Java Applet 設計一個簡單的計算機程式, 可做基本的四則運算。
6. Applet/JApplet 除了有 getImage() 方法可載入影像外, 也有參數格式相同的 play() 方法可載入音樂檔 (只支援 .au 的格式) 並播放, 請利用此方法寫一個會播放音樂的 Java Applet。
7. Applet/JApplet 另外還有一組 getAudioClip() 方法可載入音樂檔, 其參數格式也與 getImage() 相同, 請利用此方法寫一個會依 HTML 檔中指定的音樂檔, 來載入及播放音樂的 Java Applet。(提示: getAudioClip() 的傳回值型別為 java applet 套件的 AudioClip 介面, 此介面有 play()、stop() 等方法可控制音樂的播放)
8. 承上題, 替程式加入具有控制音樂開始和停止播放的控制介面。
9. 請將 F-2-5 的 ImageApplet.java 改成可同時載入多張影像, 顯示時由使用者選擇要看哪一張影像。
10. 請將 F-2-5 的 ImageApplet.java 改成 Application/Applet 兩用版。