

方法庫-範例研討：時間/日期工具套件

(A) 程式功能：Ex8_3.java

請製作一套時間與日期計算工具的工具包，再編寫一主程式導入該套件，並測試套件下類別方法與物件方法引用情形；期望操作結果如下：(如示範)

(B) 製作技巧研討：

吾人可以稍加修改 Ex8_1 的時間計算工具 (timeTool)，以及 Ex8_2 的日期工具 (dateTool)，使其合乎套件語法再重新編譯即可。又 timeTool 類別是產生物件方法，而 dateTool 是類別方法，也剛好能滿足系統的要求。系統僅要求輸入字串格式輸入，轉換再輸出日期與時間格式，因此僅使用到 dateTool 類別中 getData() 方法，與 timeTool 類別中 getTime() 方法。

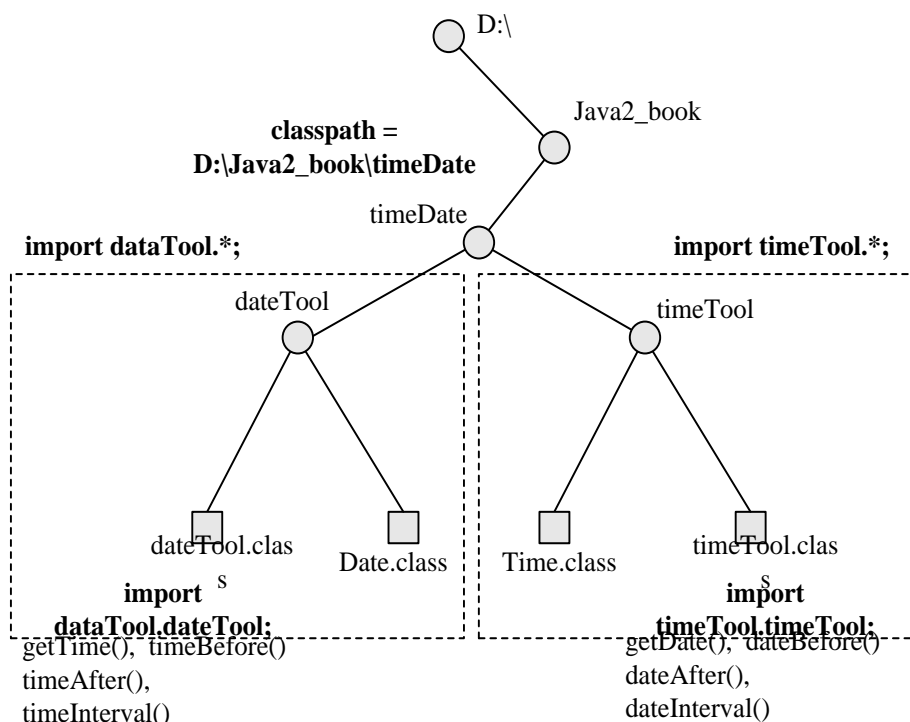


圖 8-6 timeDate 方法庫架構

吾人將 timeTool 物件方法 (Ex8_1 範例) 與 dateTool 類別方法 (Ex8_2 範例)，整合成 timeDate 方法庫，如圖 8-6 所示。首先我們利用 classpath 環境變數指定方法庫儲存目錄位置 (G:\Examples\timeDate)，接下來再修改 Ex8_1 與 Ex8_2 的程式範例，各個

程式說明如下：

(C) 程式範例：Date.java

此程式原來是包含於 dateTool.java 檔案內，為了讓其他檔案可以直接引用，因此必須將它製作成獨立檔案，並宣告成 public 屬性。此程式經編譯後會產生 Date.class 類別，功能是產生儲存日期資料的物件，原始檔案如下：

```
01 // Date.java, 產生方法套件語法
02 // 儲存於 \Examples\timeDate 目錄下(假設於 g 磁碟機)
03 // 設定環境變數 > set classpath=D:\Java2_book\timeDate;.
04 // 編譯語法: > javac -d \Java2_book\timeDate Date.java
05
06
07
08 package dateTool;
09 public class Date {
10     public int year;
11     public int month;
12     public int day;
13
14 }
15
```

- 第 1~4 行：說明使用。

(D) 程式範例：Time.java

原來此程式包含於 timeTool.java 內，我們將他分離成獨立程式。此程式將產生 Time.class 類別，功能是產生儲存時間資料的物件，程式範例如下：

```
01 // Time.java
02 package timeTool;
03
04
05 /* 時間類別，產生時間物件的格式 */
06 public class Time {
07     public int hour;
08     public int minute;
09     public int second;
10
11 }
12
13
```

(F) 程式範例：timeTool.java

吾人還是利用 Ex8_1 範例的 timrTool.java 範例（包含 4 個物件方法），將他複製到目前工作目錄下（chap8\Ex8_3），並稍加修改使成為套件的原始程式，修改地方以『粗黑』表示，顯示部分檔案如下：

```
01 // timeTool.java
02 // 儲存於 \Examples\timeDate 目錄下(假設於 g 磁碟機)
03 // 設定環境變數 > set classpath= D:\Java2_book\timeDate;.
04 // 編譯語法: > javac -d \Java2_book\timeDate timeTool.java
05
06 package timeTool;
07 import java.util.*;
08
09 public class timeTool {
10
11     /* 轉換字串成為時間物件格式 */
12     public Time getTime(String time_S){
13         .....
14         .....
15     }
16
17     /* 之前時間計算器 */
18     public Time timeBefore(Time now, Time value) {
19         .....
20         ....
21     }
22
23     /* 之後時間計算器 */
24     public Time timeAfter(Time now, Time value) {
25         .....
26         .....
27     }
28
29     /* 兩點時間之間距離計算器 */
30     public Time timeInterval(Time start, Time end) {
31         .....
32         .....
33     }
34 }
```

(F) 程式範例：dateTool.java

將原來 dateTool.java（Ex8_2 範例，包含 4 個類別方法）稍加修改使成為套件的原始程式，修改地方以『粗黑』表示，顯示部分檔案如下：

```
01 // dateTool.java, 產生方法套件語法
02 // 儲存於 \Examples\timeDate 目錄下(假設於 g 磁碟機)
03 // 設定環境變數 > set classpath= D:\Java2_book\timeDate;.
04 // 編譯語法: > javac -d \Java2_book\timeDate dateTool.java
```

```

05
06 package dateTool;
07 import java.util.*;
08
09 public class dateTool {
10
11     /* 轉換字串成為日期物件格式 */
12     public static Date getDate(String date_S){
13         Date date = new Date();
14         Scanner s = new Scanner(date_S).useDelimiter("/");
15         date.year = s.nextInt();
16         date.month = s.nextInt();
17         date.day = s.nextInt();
18         return date;
19     }
20
21     /* 之後天數的日期計算器 */
22     public static Date dateAfter(Date now, int value) {
23         ....
24         ....
25     }
26
27     /* 之前天數的日期計算器 */
28     public static Date dateBefore(Date now, int value) {
29         .....
30         .....
31     }
32
33     /* 計算兩個日期之間天數的計算器 */
34     public static int dateInterval(Date start, Date end) {
35         .....
36         .....
37     }
38 }
39
    
```

(G) 程式範例：Ex8_3.java

吾人依照系統要求，製作一個簡單程式來測試引用套件內方法是否可以成功。圖 8-7 為 Ex8_3 類別內引用方法庫的運用情形，程式範例如下：

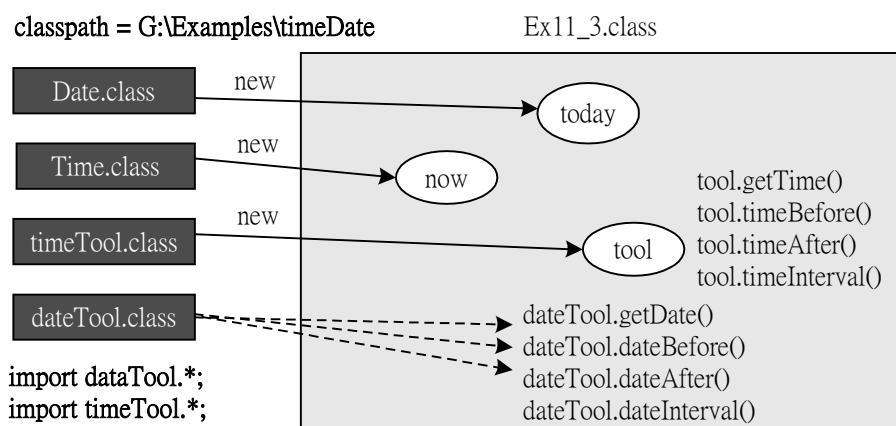


圖 8-7 Ex8_3 引用方法庫範例

```
01 // Ex8_3.java
02 // 庫存方法儲存於 D:\Java2_book\timeDate 目錄下
03 // 設定環境變數 > set classpath= D:\Java2_book\timeDate;.
04
05 import dateTool.dateTool;
06 import dateTool.Date;
07 import timeTool.timeTool;
08 import timeTool.Time;
09 import java.io.*;
10 import java.util.*;
11 public class Ex8_3 {
12     public static void main(String args[]) throws IOException {
13         BufferedReader keyin = new BufferedReader(new
14             InputStreamReader(System.in));
15         /* 時間工具是物件方法 */
16         timeTool tool = new timeTool(); // 產生時間工具
17         Time now = new Time(); // 儲存目前時間
18         String time_S; // 讀入時間字串
19
20
21         /* 日期工具是類別方法 */
22         Date today = new Date(); // 儲存今天日期
23         String date_S; // 讀入日期字串
24
25         /* 設定今天日期 */
26         System.out.printf("設定目前日期(年/月/日) =>");
27         date_S = keyin.readLine();
28         today = dateTool.getDate(date_S); // 引用類別方法
29
30
31         /* 設定目前時間 */
32         System.out.printf("設定目前時間(時/分/秒) =>");
33         time_S = keyin.readLine();
34         now = tool.getTime(time_S); // 引用物件方法
35
36         /* 輸出現在時刻 */
37         System.out.printf("\n現在是 %d 年 %d 月 %d 日",
38             today.year, today.month, today.day);
39         System.out.printf(" %d 時 %d 分 %d 秒\n",
40             now.hour, now.minute, now.second);
41         System.out.printf("== 現在時刻輸出完畢 ==\n");
42     }
43 }
44
```

(H) 編譯原始程式

上述總共製作了 5 個原始檔案，其中 4 個檔案是套件程式，另一個是主程式，編譯如下：



```
D:\Java2_book\Ex8_3>javac -d \Java2_book\timeDate dateTool.java
D:\Java2_book\Ex8_3>javac -d \Examples\timeDate timeTool.java
D:\Java2_book\Ex8_3>javac -d \Examples\timeDate Time.java
D:\Java2_book\Ex8_3>javac -d \Examples\timeDate Date.java
D:\Java2_book\Ex8_3>javac Ex8_3.java
D:\Java2_book\Ex8_3>dir/b
dateTool.java
timeTool.java
Ex8_3.java
Date.java
Time.java
Ex8_3.class
```

我們可以發現，編譯後僅多出來 Ex8_3.class 一個檔案，其他檔案將被放置於所指定的套件目錄下（G:\Examples\timeDate\），並且以套件名稱為目錄，儲存該套件下的類別檔案，可觀察如下：（可查閱到 4 個類別檔案儲存位置）

```
D:\Java2_book\chap8\Ex8_3>cd \Java2_book\timeDate
D:\Java2_book\timeDate>dir/b/s
D:\Java2_book\timeDate\dateTool
D:\Java2_book\timeDate\timeTool
D:\Java2_book\timeDate\dateTool\Date.class
D:\Java2_book\timeDate\dateTool\dateTool.class
D:\Java2_book\timeDate\timeTool\Time.class
D:\Java2_book\timeDate\timeTool\timeTool.class
```