



習題

11.1 抽象類別

1. 試修改 app11_1，在抽象類別 CShape 裡加入 CShape(String str) 建構元，用來設定資料成員 color 的值，並測試之。
2. 下面的程式碼是一個簡單數學四則運算的範例。我們在抽象類別 Math 裡已定義好一個 show()，以及 4 個 abstract 函數。請在 Compute 類別裡撰寫 add()、sub()、mul() 與 div() 這 4 個函數的定義，使得我們可以利用 Compute 類別來做兩個整數的四則運算。例如，在第 22 行建立 Compute 類別的物件 cmp 後，便可利用它來進行第 23~24 行的運算。

```
01 // hw11_2, 抽象類別的範例
02 abstract class Math
03 {
04     protected int ans;
05     public void show()
06     {
07         System.out.println("ans="+ans);
08     }
09     public abstract void add(int a, int b);    // 計算 a+b
10     public abstract void sub(int a, int b);   // 計算 a-b
11     public abstract void mul(int a, int b);   // 計算 a*b
12     public abstract void div(int a, int b);   // 計算 a/b
13 }
14 class Compute extends Math
15 {
16     // 請完成這個部分的程式碼
17 }
18 public class hw11_2
19 {
20     public static void main(String args[])
21     {
22         Compute cmp=new Compute();
23         cmp.mul(3,5); // 計算 3*5
24         cmp.show();   // 此行會回應 "ans=15" 字串
25     }
26 }
```



3. 在習題 2 中，如果在 `main()` 函數裡加入這麼一行敘述：

```
Math mth=new Math();
```

編譯時您將得到下面的錯誤訊息：

```
Math is abstract; cannot be instantiated
```

請解釋這行錯誤訊息的大意，並指明是錯在哪兒。

4. 在習題 2 中，抽象類別 `Math` 裡的 `show()` 函數是否可以宣告成 `static`？

11.2 介面的使用

5. 下面的程式碼是修改自習題 2，其中把抽象類別 `Math` 改以介面來宣告。請在 `Compute` 類別裡撰寫 `add()`、`sub()`、`mul()`、`div()` 與 `show()` 這五個函數的定義，使得我們可以利用 `Compute` 類別來做兩個整數的四則運算：

```
01 // hw11_5, 介面實作的範例
02 interface Math
03 {
04     public void show();
05     public void add(int a, int b);    // 計算 a+b
06     public void sub(int a, int b);    // 計算 a-b
07     public void mul(int a, int b);    // 計算 a*b
08     public void div(int a, int b);    // 計算 a/b
09 }
10 class Compute implements Math
11 {
12     // 請完成這個部分的程式碼
13 }
14
15 public class hw11_5
16 {
17     public static void main(String args[])
18     {
19         Compute cmp=new Compute();
20         cmp.mul(3,5); // 計算 3*5
21         cmp.show();   // 此行會回應 "ans=15" 字串
22     }
23 }
```



6. 於習題 2 中，show() 函數是詳細定義在 Math 類別裡，但於習題 5 中，show() 的定義則是放在 Compute 裡。這兩者有何差別？我們可以把習題 5 中 show() 的詳細定義放在 Math 介面裡嗎？
7. 於習題 5 中，我們是否可以在第 4~8 行的函數之前加上 abstract 關鍵字？加不加這個關鍵字有差別嗎？
8. 如果某個類別 C 同時繼承類別 B，又實作介面 A，則我們必須以下面的語法來撰寫類別 A：

```
class C extends B implements A
{
    // 類別 C 裡的程式碼
}
```

下面的程式碼定義介面 iAaa 與類別 Cbbb，試撰寫類別 Cccc，使其繼承自類別 Cbbb，時也實作介面 iAaa，並於程式執行時，會於第 25 行印出 "num=5" 字串。

```
01 // hw11_8
02 interface iAaa
03 {
04     public void show();
05 }
06
07 class Cbbb
08 {
09     public int num=10;
10
11     public void set(int n)
12     {
13         num=n;
14     }
15 }
16
17 // 請於此處定義 Cccc 類別
18
19 public class hw11_8
20 {
21     public static void main(String args[])
```



```
22     {
23         Cccc cc=new Cccc();
24         cc.set(5);
25         cc.show();    // 印出 num=5
26     }
27 }
```

11.3 多重繼承

9. 下面的程式碼是修改自習題 5，其中增加一個 AdvancedMath 介面。AdvancedMath 介面裡定義 3 個函數：

```
(1) public void mod(int a, int b)    // 計算 a%b
(2) public void fac(int a);          // 計算 a!
(3) public void pow(int a, int b);   // 計算 ab
```

請在 Compute 類別裡撰寫所有函數的定義，使得我們可以利用 Compute 類別來做加減乘除與 mod()、fac()、pow() 等運算：

```
01 // hw11_9, 多重繼承的練習
02 interface Math
03 {
04     void show();
05     public void add(int a, int b);
06     public void sub(int a, int b);
07     public void mul(int a, int b);
08     public void div(int a, int b);
09 }
10 interface AdvancedMath
11 {
12     public void mod(int a, int b);    // 計算 a%b
13     public void fac(int a);          // 計算 a!
14     public void pow(int a, int b);    // 計算 ab
15 }
16 class Compute implements Math,AdvancedMath
17 {
18     // 請完成這個部分的程式碼
19 }
20
21 public class hw11_9
```



```

22  {
23      public static void main(String args[])
24      {
25          Compute cmp=new Compute();
26          cmp.mul(3,5);
27          cmp.show();           // 此行會回應 "ans=15" 字串
28          cmp.mod(14,5);
29          cmp.show();           // 此行會回應 "ans=4" 字串
30          cmp.fac(5);
31          cmp.show();           // 此行會回應 "ans=120" 字串
32      }
33  }

```

10. 在習題 9 中,如果把第 16 行改寫成 Compute 類別只實作 Math 介面,而非同時實作 Math 與 AdvancedMath 介面,亦即把 16 行改寫成:

```
16  class Compute implements Math
```

那麼 mod()、fac() 與 pow() 這些函數是否還是可以正確執行?為什麼?

11. 假設有個 CStu 類別,其資料成員如下:

```

class CStu
{
    protected String id;           // 學號
    protected String name;         // 姓名
    protected int mid;              // 期中考成績
    protected int finl;             // 期末考成績
    protected int common;           // 平時成績
}

```

下面的程式中,在介面 Data 裡已定義好一個 showData(),用來顯示學生的學號及姓名。介面 Test 裡已定義好 showScore(),用來顯示學生的各項成績;calcu() 則是將學期成績以期中、期末考佔 30%,平時成績佔 40%的方式計算。試完成下面的程式,使得輸出的項目,除了該生的資料之外,還要顯示學期成績。

```

01  // hw11_11, 多重繼承的練習
02  interface Data
03  {
04      public void showData();
05  }

```



```
06 interface Test
07 {
08     public void showScore();
09     public double calcu();
10 }
11
12 // 請完成這個部分的程式
13
14 public class hw11_11
15 {
16     public static void main(String args[])
17     {
18         CStu stu=new CStu("940001","Fiona",90,92,85);
19         stu.show();
20     }
21 }
```

11.4 介面的延伸

12. 下面的程式中，我們先宣告一個介面 Data，再宣告另一個介面 Test 繼承它。

```
01 // hw11_12,
02 interface Data
03 {
04     public void best();           // 判斷那一科成績較高
05     public void failed();        // 判斷那一科成績低於 60 分
06 }
07 interface Test extends Data
08 {
09     public void showData();       // 顯示學生的資料及平均成績
10     public double average();      // 計算數學和英文的平均成績
11 }
12 class CStu implements Test
13 {
14     protected String name;       // 姓名
15     protected int math;          // 數學成績
16     protected int english;       // 英文成績
17
18     // 請完成這個部分的程式
19 }
20
```



```
21 public class hw11_12
22 {
23     public static void main(String args[])
24     {
25         CStu stu=new CStu("Judy",58,91);
26         stu.show();
27     }
28 }
```

請在 CStu 類別裡撰寫所有函數的定義，再於 CStu 類別中加入一個 show()，用來呼叫 best()、failed()、showData()與 average()等函數。使得程式執行後的結果如下：

```
/* output-----
姓名:Judy
數學成績:58
英文成績:91
平均成績:74.5
Judy 的英文比數學好
Judy 的數學當掉了
-----*/
```

13. 下面的範例修改自習題 5，我們先宣告一個 Show_ans 介面，再宣告另一個介面 Math 繼承它。請在 Compute 類別裡撰寫所有函數的定義，使得我們可以利用 Compute 類別的物件來呼叫 show()、add()、sub()、mul() 與 div() 等運算。

```
01 // hw11_13,介面的延伸
02 interface Show_ans
03 {
04     public void show();
05 }
06 interface Math extends Show_ans
07 {
08     public void add(int a, int b);
09     public void sub(int a, int b);
10     public void mul(int a, int b);
11     public void div(int a, int b);
12 }
13
14 class Compute implements Math
15 {
```



```
16      // 請完成這個部分的程式碼
17  }
18
19  public class hw11_13
20  {
21      public static void main(String args[])
22      {
23          Compute cmp=new Compute();
24          cmp.mul(3,5);
25          cmp.show();
26      }
27  }
```

14. 下面的程式中，我們先宣告一個介面 iVolume，再宣告一個抽象函數 CSphere 實作它。

```
01  // hw11_14,
02  interface iVolume
03  {
04      public void showData();           // 顯示球體的資料
05      public double vol();             // 計算球體積
06  }
07  abstract class CSphere implements iVolume
08  {
09      final double PI=3.14;
10      protected int x;
11      protected int y;
12  }
13  class CCircle extends CSphere
14  {
15      // 請完成這個部分的程式碼
16  }
17
18  public class hw11_14
19  {
20      public static void main(String args[])
21      {
22          CCircle cir=new CCircle(8,6,2);
23          cir.showData();
24      }
25  }
```

- (a) 請在 CCircle 類別裡撰寫適當的程式，使得程式執行後的結果如下：



```

/* output-----
球心: (8, 6)
半徑: 2
球體積: 33.49333333333333
-----*/

```

（註：CSphere 物件的體積為 $\frac{4}{3}\pi r^3$ ）。

- (b) 抽象函數 CSphere 雖然實作介面 iVolume，可是在抽象函數 CSphere 裡卻沒有看到關於介面 iVolume 所宣告的 showData() 與 vol() 函數，為什麼可以編譯無誤？
- (c) 完成(a)之後，試將 CCircle 類別裡的 showData() 與 vol() 移到抽象函數 CSphere 裡，並請稍做修改，使得程式可以執行。

11.5 類別關係的判別—instanceof

15. 請依下面的步驟逐步完成程式的需求：
 - (a) 分別建立 CShape、CCircle、CTriangle、CCoin 類別，類別裡的定義可以不寫。
 - (b) 設定 CCircle、CTriangle 為 CShape 的子類別，CCoin 為 CCircle 的子類別。
 - (c) 於主程式中分別宣告 CCircle、CTriangle、CCoin 物件 cicrle、triangle、coin。
 - (d) 請判斷 cicrle 是否為 CCoin 的子類別物件、triangle 是否為 CShape 的子類別物件、coin 是否為 Object 的子類別物件。
16. 請於程式中測試 null 是否繼承 Object 類別。