Appendix A

習題解答

Chapter 1 Java的第一步

1. ① ×　② ○　③ ×　④ ×　⑤ ×

Chapter 2 Java的基本知識

1. 這一段程式碼在語法上並沒有錯，編譯、執行時也不會發生問題，但是程式碼沒有層級，很難讓人看懂。建議您適度加以換行和縮排，就像下面這樣。

```
// 輸出文字到螢幕的程式碼
class SampleP1
{
   public static void main(String[] args)
   {
      System.out.println("歡迎來到Java?!");
      System.out.println("Java語言真好用!");
   }
}
```

2.

```
// 輸入文字與數值
Class Sample2
{
   Public static void main(String[] args)
   {
      System.out.println('A')
      System.out.println("歡迎來到Java!")
      System.out.printin(123)
   }
}
```

3.

```
class SampleP3
{
   public static void main (String[] args)
   {
      System.out.println(123);
      System.out.println("\\100");
      System.out.println("明天見！");
   }
}
```

4.

```
class Sample4
{
   public static void main(String[] args)
   {
      System.out.println("1\t2\t3\t");
   }
}
```

5.

● 八進制

```
class SampleP5_1
{
   public static void main (String[] args)
   {
      System.out.println(06);
      System.out.println(024);
      System.out.println(015);
   }
}
```

● 十六進制

```
class SampleP5_2
{
   public static void main (String[] args)
   {
      System.out.println(0x6);
      System.out.println(0x14);
      System.out.println(0xD);
   }
}
```

Chapter 3 變數

1. ① ×　② ×　③ ○

2. 3.14這個數字無法直接指定給char型態的變數。

3.

```
import java.io.*;

class SampleP3
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("你今年幾歲？");

      BufferedReader br =
       new BufferedReader(new InputStreamReader(System.in));

      String str = br.readLine();

      int num = Integer.parseInt(str);

      System.out.println("你今年" + num + "歲。");
   }
}
```

4.

```
import java.io.*;

class SampleP4
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入圓周率的值：");

      BufferedReader br =
        new BufferedReader(new InputStreamReader(System.in));

      String str = br.readLine();

      double pi = Double.parseDouble(str);

      System.out.println("圓周率的值是" + pi + "。");
   }
}
```

5.

```
import java.io.*;

class SampleP5
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入身高和體重：");

      BufferedReader br =
        new BufferedReader(new InputStreamReader(System.in));

      String str1 = br.readLine();
      String str2 = br.readLine();
```

```
        double num1 = Double.parseDouble(str1);
        double num2 = Double.parseDouble(str2);

        System.out.println("身高是" + num1 + "。");
        System.out.println("體重是" + num2 + "。");
    }
}
```

Chapter 4 運算式和運算子

1. ① ×　　② ○　　③ ×

2.

```
class SampleP2
{
    public static void main(String[] args)
    {
        int ans1 = 0-4;
        double ans2 = 3.14*2;
        double ans3 =(double)5/3;
        int ans4 = 30%7;
        double ans5 = (7+32)/(double)5;

        System.out.println("0-4等於" + ans1);
        System.out.println("3.14×2等於" + ans2);
        System.out.println("5÷3等於" + ans3);
        System.out.println("30÷7的餘數為" + ans4);
        System.out.println("(7+32)÷5等於" + ans5);
    }
}
```

3.

```
import java.io.*;

class SampleP3
{
    public static void main(String[] args) throws IOException
    {
        System.out.println("請輸入正方形的邊長：");

        BufferedReader br =
         new BufferedReader(new InputStreamReader(System.in));

        String str = br.readLine();

        int width = Integer.parseInt(str);
        System.out.println("正方形的面積是" + (width*width) +"。");
    }
}
```

4.

```
import java.io.*;

class SampleP4
{
    public static void main(String[] args) throws IOException
    {
        System.out.println("請輸入三角形的底和高：");

        BufferedReader br =
          new BufferedReader(new InputStreamReader(System.in));

        String str1 = br.readLine();
        String str2 = br.readLine();

        double height = Integer.parseInt(str1);
        double width = Integer.parseInt(str2);

        System.out.println("三角形的面積是"
           + (height * width / (double) 2) + "。");
    }
}
```

5.

```
import java.io.*;

class SampleP5
{
    public static void main(String[] args) throws IOException
    {
        System.out.println("請輸入科目1~5的分數：");

        BufferedReader br =
          new BufferedReader(new InputStreamReader(System.in));

        String str1 = br.readLine();
        String str2 = br.readLine();
        String str3 = br.readLine();
        String str4 = br.readLine();
        String str5 = br.readLine();

        int sum = 0;
        sum += Integer.parseInt(str1);
```

```
      sum += Integer.parseInt(str2);
      sum += Integer.parseInt(str3);
      sum += Integer.parseInt(str4);
      sum += Integer.parseInt(str5);

      System.out.println("5個科目的總分為" + sum + "。");
      System.out.println("5個科目的平均為" +
         (sum /(double) 5) + "。");
   }
}
```

另外，只要使用第6章介紹的迴圈敘述，就能把這個程式寫得更簡潔。

Chapter 5 條件處理

1. ① a >= 0 && a < 10

   ② !(a==0)

   ③ a >=10 || a == 0

2.

```
import java.io.*;

class SampleP2
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入整數。");

      BufferedReader br =
        new BufferedReader(new InputStreamReader(System.in));

      String str = br.readLine();
      int res = Integer.parseInt(str);

      if((res % 2) == 0)
         System.out.println(res + "是偶數。");
      else
         System.out.println(res + "是奇數。");
   }
}
```

3.

```
import java.io.*;

class SampleP3
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入兩個整數。");

      BufferedReader br =
        new BufferedReader(new InputStreamReader(System.in));

      String str1 = br.readLine();
      String str2 = br.readLine();

      int num1 = Integer.parseInt(str1);
      int num2 = Integer.parseInt(str2);

      if(num1 < num2){
         System.out.println(num1 + "比" + num2 + "來得小。");
      }
      else if(num1 > num2){
         System.out.println(num1 + "比" + num2 + "來得大。");
      }
      else{
         System.out.println("兩個數值是相同的。");
      }
   }
}
```

4.

```
import java.io.*;

class SampleP4
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入0到10之間的整數。");

      BufferedReader br =
       new BufferedReader(new InputStreamReader(System.in));

      String str = br.readLine();
      int res = Integer.parseInt(str);

      if(res >= 0 && res <= 10){
         System.out.println("正確答案。");
```

```
        }
        else{
            System.out.println("答錯了。");
        }
    }
}
```

5.

```
import java.io.*;

class SampleP5
{
    public static void main(String[] args) throws IOException
    {
        System.out.println("請輸入成績。");

        BufferedReader br =
          new BufferedReader(new InputStreamReader(System.in));

        String str = br.readLine();
        int res = Integer.parseInt(str);

        switch(res){
            case 1:
                System.out.println("請拼命加油吧！");
                break;
            case 2:
                System.out.println("請再加油一點！");
                break;
            case 3:
                System.out.println("您還可以作得更好！");
                break;
            case 4:
                System.out.println("幹得好！");
                break;
            case 5:
                System.out.println("您真是太優秀了！");
                break;
        }
    }
}
```

Chapter 6 迴圈敘述

1.

```
class SampleP1
{
   public static void main(String[] args)
   {
      System.out.println("輸出1～10的偶數。");

      for(int i=1; i<=10; i++){
         if((i % 2) == 0)
            System.out.println(i);
      }
   }
}
```

2.

```
import java.io.*;

class SampleP2
{
   public static void main(String[] args)throws IOException
   {
      System.out.println("請輸入考試的成績。（輸入0就結束）");

      BufferedReader br =
        new BufferedReader(new InputStreamReader(System.in));

      int num = 0;
      int sum = 0;

      do{
         String str = br.readLine();
         num = Integer.parseInt(str);
         sum += num;
      }while(num != 0);

      System.out.println("總分為" + sum + "分。");
   }
}
```

3.

```
class SampleP3
{
   public static void main(String[] args)
   {
      for(int i=1; i<=9; i++){
         for(int j=1; j<=9; j++){
            System.out.print(i*j + "\t");
         }
         System.out.print("\n");
      }
   }
}
```

4.

```
class SampleP4
{
   public static void main(String[] args)
   {
      for(int i=1; i<=5; i++){
         for(int j=0; j<i; j++){
            System.out.print("*");
         }
         System.out.print("\n");
      }
   }
}
```

5.

```
import java.io.*;

class SampleP5
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入大於2的整數。");

      BufferedReader br =
       new BufferedReader(new InputStreamReader(System.in));

      String str = br.readLine();
      int num = Integer.parseInt(str);

       for(int i=2; i<=num; i++){
          if(i == num){
             System.out.println(num + "是質數。");
           }
           else if(num % i == 0){
             System.out.println(num + "不是質數。");
             break;
           }
       }
   }
}
```

Chapter 7 陣列

1. ① × ② × ③ ○

2. 陣列的索引(test[5])超過了陣列的範圍。

3. ① A ② D ③ F ④ E

4.

```
import java.io.*;

class SampleP4
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入五個人的分數。");

      BufferedReader br =
        new BufferedReader(new InputStreamReader(System.in));

      int[] test = new int[5];

      for(int i=0; i<test.length; i++){
         String str = br.readLine();
         int tmp = Integer.parseInt(str);
         test[i] = tmp;
      }

      int max = 0;

      for(int i=0; i<test.length; i++){
          if(max < test[i]){
               max = test[i];
          }
      }

      for(int i=0; i<test.length; i++){
         System.out.println("第" + (i+1) + "個人的分數是" +
            test[i] + "分。");
      }

      System.out.println("最高分是" + max + "分。");
   }
}
```

Chapter 8 類別的基礎知識

1. ① ○　② ○　③ ×

2. 呼叫setNumGas()方法時沒有加上指向物件的變數名稱，應該改成car1.setNumGas()才對。

3. 2

4. ① ○　② ×　③ ○

5.

```
class MyPoint
{
   int x;
   int y;

   void setX(int px)
   {
      x = px;
   }
   void setY(int py)
   {
      y = py;
   }
   int getX()
   {
      return x;
   }
   int getY()
   {
      return y;
   }
}

class SampleP5
{
   public static void main(String[] args)
   {
      MyPoint p1;
      p1 = new MyPoint();
      p1.setX(10);
      p1.setY(5);

      int px = p1.getX();
      int py = p1.getY();

      System.out.println("X座標是" + px + "，Y座標是" + py + "。");
   }
}
```

Chapter 9 類別的功能

1. ① ○　② ○　③ ×　④ ×

2. 在類別方法中呼叫出實體方法 show()。

3. ① ×　② ○　③ ○

4. 請注意 this(); 的部分。在建立第一個物件的時候會輸出①的結果，建立第二個物件的時候才會輸出②、③的結果。

   ① 沒有　② 沒有　③ 一個

5.

```
class MyPoint
{
   private int x;
   private int y;

   public MyPoint()
   {
      x = 0;
      y = 0;
   }
   public MyPoint(int px, int py)
   {
      x = px;
      y = py;
   }
   public void setX(int px)
   {
      if(px >= 0 && px <=100)
         x = px;
   }
   public void setY(int py)
   {
      if(py >= 0 && py <=100)
         y = py;
   }
   public int getX()
   {
      return x;
   }
   public int getY()
   {
      return y;
   }
}

class SampleP5
{
   public static void main(String[] args)
   {
      MyPoint p1;
      p1 = new MyPoint();
      p1.setX(10);
      p1.setY(5);

      int px1 = p1.getX();
      int py1 = p1.getY();

      System.out.println("p1的X座標是" + px1 + "，Y座標是" +
         py1 + "。");

      MyPoint p2;
      p2 = new MyPoint(20,10);

      int px2 = p2.getX();
      int py2 = p2.getY();

      System.out.println("p2的X座標是" + px2 + "，Y座標是" +
         py2 + "。");
   }
}
```

Chapter 10 類別的應用方式

1. ① ×　　② ○

2.

```
import java.io.*;

class SampleP2
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入字串。");

      BufferedReader br =
        new BufferedReader(new InputStreamReader(System.in));

      String str1 = br.readLine();
      StringBuffer str2 = new StringBuffer(str1);
      str2.reverse();

      System.out.println("把" + str1 + "反過來寫就會變成" + str2 +
         "。");
   }
}
```

3.

```
import java.io.*;

class SampleP3
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入字串。");

      BufferedReader br =
       new BufferedReader(new InputStreamReader(System.in));

      String str1 = br.readLine();

      System.out.println("請以整數來輸入a的插入位置。");

      String str2 = br.readLine();
      int num = Integer.parseInt(str2);

      StringBuffer str3 = new StringBuffer(str1);
      str3.insert(num, 'a');

      System.out.println("字串變成" + str3 + "了。");
   }
}
```

4.

```
import java.io.*;

class SampleP4
{
   public static void main(String[] args) throws IOException
   {
      System.out.println("請輸入兩個整數。");

      BufferedReader br =
        new BufferedReader(new InputStreamReader(System.in));

      String str1 = br.readLine();
      String str2 = br.readLine();

      int num1 = Integer.parseInt(str1);
      int num2 = Integer.parseInt(str2);

      int ans = Math.min(num1, num2);

      System.out.println(num1 + "和" + num2 +
         "之中較小的一方是" + ans + "。");
   }
}
```

Chapter 11 建立新類別

1. ① ×　② ×　③ ○　④ ○

2. ① ×　② ×　③ ○

3. ① A　② 無　③ B　④ 無　⑤ A　⑥ 單　⑦ B　⑧ 單

建立第一個物件之後，父類別的無參數建構式會先被呼叫。在建立第二個物件的時候，則會透過呼叫 super(b); 而先呼叫單參數的建構式。

4.

```
class Car
{
   protected int num;
   protected double gas;

   public Car()
   {
      num = 0;
      gas = 0.0;
      System.out.println("生產了車子。");
   }
   public void setCar(int n, double g)
   {
      num = n;
      gas = g;
      System.out.println("將車號設為" + num + "，汽油量設為" +
         gas + "。");
   }
   public String toString()
   {
      String str = "車號為" + num + "，汽油量為" + gas + "的車子";
      return str;
   }
}

class SampleP4
{
   public static void main(String[] args)
   {
      Car car1 = new Car();
      car1.setCar(1234, 20.5);

      System.out.println("這是" + car1 + "。");
   }
}
```

Chapter 12 介面

1. ① ×　② ○　③ ○　④ ×

2. ① ○　② ○　③ ×（D是介面）

3. a

4. 無法建立抽象類別的物件。

Chapter 13 設計大型程式－入門篇

1. ① ○　② ×　③ ○　④ ×

2. ① ○　② ×　③ ○

3. ① A　② A、B　③ C、D

4. 指定的套件名稱有誤。

Chapter 14 例外處理與輸入 / 輸出處理

1. ① ×　② ○　③ ○

2.

```
import java.io.*;

class SampleP2
{
   public static void main(String[] args)
   {
      try{
         PrintWriter pw = new PrintWriter
         (new BufferedWriter(new FileWriter("test1.txt")));

         pw.println("A long time ago,");
         pw.println("There was a little girl.");

         pw.close();
      }
      catch(IOException e){
         System.out.println("輸出入錯誤。");
      }
   }
}
```

3.

```
import java.io.*;

class SampleP3
{
   public static void main(String[] args)
   {
      if(args.length != 1){
         System.out.println("請指定正確的檔案名稱。");
         System.exit(1);
      }

      try{
         PrintWriter pw = new PrintWriter
         (new BufferedWriter(new FileWriter(args[0])));

         pw.println("A long time ago,");
         pw.println("There was a little girl.");

         pw.close();
      }
      catch(IOException e){
         System.out.println("輸出入錯誤。");
      }
   }
}
```

Chapter 15 執行緒

1. ① ○　② ○　③ ×

2.

```
class Car implements Runnable
{
   private String name;

   public Car(String nm)
   {
      name = nm;
   }
   public void run()
   {
      for(int i=0; i<5; i++){
         System.out.println("正在進行" + name + "的處理工作。");
      }
   }
}
```

```
class SampleP2
{
   public static void main(String[] args)
   {
      Car car1 = new Car("1號車");

      Thread th1 = new Thread(car1);
      th1.start();

      Car car2 = new Car("2號車");

      Thread th2 = new Thread(car2);
      th2.start();

      for(int i=0; i<5; i++){
         System.out.println("正在進行main()的處理工作。");
      }
   }
}
```

3.

```
class Car implements Runnable
{
   private String name;

   public Car(String nm)
   {
      name = nm;
   }
   public void run()
   {
      for(int i=0; i<5; i++){
         try{
            Thread.sleep(1000);
            System.out.println("正在進行" + name + "的處理工作。");
         }
         catch(InterruptedException e){}
      }
   }
}

class SampleP3
{
   public static void main(String[] args)
   {
      Car car1 = new Car("1號車");

      Thread th1 = new Thread(car1);
      th1.start();

      for(int i=0; i<5; i++){
         System.out.println("正在進行main()的處理工作。");
      }
   }
}
```

Chapter 16 Applet

1.

```
import java.applet.Applet;
import java.awt.Graphics;
import java.awt.Color;
import java.awt.Font;

public class SampleP1 extends Applet
{
   public void paint(Graphics g)
   {
      g.setColor(Color.blue);
      g.setFont(new Font("Serif", Font.BOLD, 20));
      g.drawString("Hello", 20, 20);
   }
}
```

2.

```
import java.applet.Applet;
import java.awt.Graphics;

public class SampleP2 extends Applet
{
   public void paint(Graphics g)
   {
      g.fillRect(10, 10, 100, 100);
   }
}
```

3.

```
import java.applet.Applet;
import java.awt.Graphics;
import java.awt.Image;
import java.awt.event.MouseListener;
import java.awt.event.MouseEvent;
```

```
public class SampleP3 extends Applet implements MouseListener
{
    Image img;
    int x = 10;
    int y = 10;

    public void init()
    {
        addMouseListener(this);
        img = getImage(getDocumentBase(),"Image.gif");
    }
    public void mouseClicked(MouseEvent e)
    {
    }
    public void mouseEntered(MouseEvent e)
    {
    }
    public void mouseExited(MouseEvent e)
    {
    }
    public void mousePressed(MouseEvent e)
    {
        x = e.getX();
        y = e.getY();
        repaint();
    }
    public void mouseReleased(MouseEvent e)
    {
    }
    public void paint(Graphics g)
    {
        g.drawImage(img, x, y, this);
    }
}
```

4.

```
import java.applet.Applet;
import java.awt.Graphics;
import java.awt.event.MouseListener;
import java.awt.event.MouseEvent;

public class SampleP4 extends Applet implements MouseListener
{
    boolean bl = true;

    public void init()
    {
        addMouseListener(this);
    }
    public void mouseClicked(MouseEvent e)
    {
    }
    public void mouseEntered(MouseEvent e)
    {
        bl = true;
        repaint();
    }
    public void mouseExited(MouseEvent e)
    {
        bl = false;
        repaint();
    }
    public void mousePressed(MouseEvent e)
    {
    }
    public void mouseReleased(MouseEvent e)
    {
    }
    public void paint(Graphics g)
    {
        if(bl == true){
            g.drawString("您好", 10, 10);
        }
        else{
            g.drawString("再見", 10, 10);
        }
    }
}
```

5.

```
import java.applet.Applet;
import java.awt.Graphics;

public class SampleP5 extends Applet implements Runnable
{
    int num;
    int x;

    public void init()
    {
        Thread th;
        th = new Thread(this);
        th.start();
    }
    public void run()
    {
        try{
         for(int i=0; i<10; i++){
            num = i;
            x = i*10;
            repaint();
            Thread.sleep(1000);
         }
      }catch(InterruptedException e){}
   }
   public void paint(Graphics g)
   {
      String str = "這是數字" + num + "喔。";
      g.drawString(str, x, 10);
   }
}
```

Chapter 17 複習重要基礎概念

1.

```
public class SampleP1
{
   public static void main(String[] args)
   {
      System.out.println("早安");
      System.out.println("再見");
   }
}
```

2.

SampleP2.html

```
<html>
<head>
<title>範例</title>
</head>
<body>
<center>
<applet code="SampleP2.class" width="300" height="300">
</applet>
</center>
</body>
</html>
```

SampleP2.java

```
import javax.swing.*;

public class SampleP2 extends JApplet
{
   private JLabel lb;

   public void init()
   {
      lb = new JLabel();

      lb.setText("歡迎使用Applet！");

      add(lb);
   }
}
```

Chapter 18 類別庫

省略

Chapter 19 Swing的基礎知識

1. （省略HTML文件，以下亦然）

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

public class SampleP1 extends JApplet
{
   private JLabel lb;
   private JButton bt;

   public void init()
   {
      // 建立元件
      lb = new JLabel("歡迎光臨。");
      bt = new JButton("購買");

      // 新增到容器中
      add(lb, BorderLayout.NORTH);
      add(bt, BorderLayout.SOUTH);

      // 登錄傾聽者
      bt.addActionListener(new SampleActionListener());
```

```
    }

    // 傾聽者類別
    class SampleActionListener implements ActionListener
    {
        public void actionPerformed(ActionEvent e)
        {
            bt.setText("Thanks!");
        }
    }

}
```

2.

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

public class SampleP2 extends JApplet
{
    private JButton bt;

    public void init()
    {
        // 建立元件
        bt = new JButton("歡迎。");

        // 新增到容器中
        add(bt, BorderLayout.NORTH);

        // 登錄傾聽者
        bt.addMouseListener(new SampleMouseListener());
    }

    // 傾聽者類別
    class SampleMouseListener extends MouseAdapter
    {
        public void mouseEntered(MouseEvent e)
        {
            bt.setText("歡迎光臨。");
        }
        public void mouseExited(MouseEvent e)
        {
            bt.setText("歡迎。");
        }
    }
}
```

3.

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

public class SampleP3 extends JApplet
{
   private JLabel lb1, lb2;
   private String str;

   public void init()
   {
      // 建立元件
      lb1 = new JLabel("請按下鍵盤按鍵。");
      lb2 = new JLabel();

      // 新增到容器中
      add(lb1, BorderLayout.NORTH);
      add(lb2, BorderLayout.CENTER);

      // 登錄傾聽者
      addKeyListener(new SampleKeyListener());
   }

   // 傾聽者類別
   class SampleKeyListener extends KeyAdapter
   {
      public void keyPressed(KeyEvent e)
      {
         char c = e.getKeyChar();
         lb2.setText("是" + c + "吧。");
      }
   }
}
```

Chapter 20 Swing的進階應用

1.

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

public class SampleP1 extends JApplet
{
   private JLabel lb;
   private JPanel pn;
   private JRadioButton rb1, rb2, rb3, tmp;
   private ButtonGroup bg;

   public void init()
   {
      // 建立元件
      lb  = new JLabel("歡迎光臨。");
      pn = new JPanel();
      rb1 = new JRadioButton("黃", true);
      rb2 = new JRadioButton("紅", false);
      rb3 = new JRadioButton("藍", false);
      bg  = new ButtonGroup();

      // 設定元件
      lb.setOpaque(true);
      lb.setBackground(Color.yellow);

      // 新增到按鈕群組中
      bg.add(rb1);
      bg.add(rb2);
      bg.add(rb3);

      // 新增到容器中
      pn.add(rb1);
      pn.add(rb2);
      pn.add(rb3);
      add(lb, BorderLayout.NORTH);
      add(pn, BorderLayout.SOUTH);

      // 登錄傾聽者
      rb1.addActionListener(new SampleActionListener());
      rb2.addActionListener(new SampleActionListener());
      rb3.addActionListener(new SampleActionListener());
   }

   // 傾聽者類別
   class SampleActionListener implements ActionListener
   {
      public void actionPerformed(ActionEvent e)
      {
          tmp = (JRadioButton) e.getSource();
          if(tmp == rb1)
             lb.setBackground(Color.yellow);
          else if(tmp == rb2)
             lb.setBackground(Color.red);
          else if(tmp == rb3)
             lb.setBackground(Color.blue);
      }
   }
}
```

2.

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

public class SampleP2 extends JApplet
{
   private JLabel lb;
   private JCheckBox cb;
   private Icon ic;

   public void init()
   {
      // 建立元件
      lb = new JLabel("這是車子。");
      cb = new JCheckBox("顯示影像");
      ic = new ImageIcon(getImage(getDocumentBase(), "car.gif"));

      // 設定元件
      lb.setBorder(new LineBorder(Color.blue, 10));

      // 新增到容器中
      add(lb, BorderLayout.CENTER);
      add(cb, BorderLayout.SOUTH);

      // 登錄傾聽者
      cb.addItemListener(new SampleItemListener());
   }
   // 傾聽者類別
   class SampleItemListener implements ItemListener
   {
      public void itemStateChanged(ItemEvent e)
      {
         if(e.getStateChange() == ItemEvent.SELECTED)
         {
            lb.setIcon(ic);
         }
         else if(e.getStateChange() == ItemEvent.DESELECTED)
         {
            lb.setIcon(null);
         }
      }
   }
}
```

3.

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

public class SampleP3 extends JApplet
{
   private JLabel lb;
   private JPanel pn;
   private JRadioButton rb1, rb2, rb3, tmp;
   private ButtonGroup bg;

   public void init()
   {
      // 建立元件
      lb = new JLabel("Hello!");
      pn = new JPanel();
      rb1 = new JRadioButton("普通", true);
      rb2 = new JRadioButton("粗體", false);
      rb3 = new JRadioButton("斜體", false);
      bg  = new ButtonGroup();

      // 設定元件
      lb.setFont(new Font("Serif", Font.PLAIN, 20));
      lb.setHorizontalAlignment(JLabel.CENTER);

      // 新增到按鈕群組中
      bg.add(rb1);
      bg.add(rb2);
      bg.add(rb3);

      // 新增到容器中
      pn.add(rb1);
      pn.add(rb2);
      pn.add(rb3);
      add(lb, BorderLayout.NORTH);
      add(pn, BorderLayout.SOUTH);

      // 登錄傾聽者
      rb1.addActionListener(new SampleActionListener());
      rb2.addActionListener(new SampleActionListener());
      rb3.addActionListener(new SampleActionListener());
   }

   // 傾聽者類別
   class SampleActionListener implements ActionListener
   {
      public void actionPerformed(ActionEvent e)
      {
         tmp = (JRadioButton) e.getSource();
```

```
          if(tmp == rb1)
             lb.setFont(new Font("Serif", Font.PLAIN, 20));
          else if(tmp == rb2)
             lb.setFont(new Font("Serif", Font.BOLD, 20));
          else if(tmp == rb3)
             lb.setFont(new Font("Serif", Font.ITALIC, 20));
      }
   }
}
```

Chapter 21 Swing的活用技巧

1.

```
import java.util.*;
import java.text.*;
import java.awt.*;
import javax.swing.*;

public class SampleP1 extends JApplet
{
   private JLabel lb;
   private JList lst;
   private JScrollPane sp;

   public void init()
   {
      // 建立元件
      lb = new JLabel("歡迎光臨。");
      lst = new JList(new MyListModel());
      sp = new JScrollPane(lst);

      // 新增到容器中
      add(lb, BorderLayout.NORTH);
      add(sp, BorderLayout.SOUTH);
    }

   // 模型類別
   class MyListModel extends AbstractListModel
   {
       DateFormat df;

       public MyListModel()
       {
           df = new SimpleDateFormat("yyyy/MM/dd");
       }
       public int getSize()
       {
          return 50;
       }
       public Object getElementAt(int index)
       {
           Calendar cl = Calendar.getInstance();
           cl.setTime(new Date());
           cl.add(Calendar.DATE, index);

           String str = df.format(cl.getTime());
           return str;
       }
   }
}
```

2.

```
import java.util.*;
import java.text.*;
import java.awt.*;
import javax.swing.*;
import javax.swing.event.*;

public class SampleP2 extends JApplet
{
   private JLabel lb;
   private JList lst;
   private JScrollPane sp;

   public void init()
   {
      // 建立元件
      lb = new JLabel("顯示日期。");
      lst = new JList(new MyListModel());
      sp = new JScrollPane(lst);

      // 新增到容器中
      add(lb, BorderLayout.NORTH);
      add(sp, BorderLayout.SOUTH);

      // 登錄傾聽者
      lst.addListSelectionListener
         (new SampleListSelectionListener());
   }

   // 傾聽者類別
   class SampleListSelectionListener implements
      ListSelectionListener
   {
      public void valueChanged(ListSelectionEvent e)
      {
         JList tmp = (JList) e.getSource();
         String str = (String) tmp.getSelectedValue();
         lb.setText("是" + str + "吧。");
      }
   }

   // 模型類別
   class MyListModel extends AbstractListModel
   {
       DateFormat df;

       public MyListModel()
       {
           df = new SimpleDateFormat("yyyy/MM/dd");
       }
       public int getSize()
       {
          return 50;
       }
       public Object getElementAt(int index)
       {
           Calendar cl = Calendar.getInstance();
           cl.setTime(new Date());
           cl.add(Calendar.DATE, index);

           String str = df.format(cl.getTime());
           return str;
       }
   }
}
```

3.

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

public class SampleP3 extends JFrame
{
   private JLabel lb;
   private JPanel pn;
   private JButton bt;

   public static void main(String[] args)
   {
      SampleP3 sm = new SampleP3();
   }
   public SampleP3()
   {
      // 設定標題
      super("範例");

      // 建立元件
      lb = new JLabel("歡迎光臨。");
      pn = new JPanel();
      bt = new JButton("購買");

      // 新增到容器中
      pn.add(bt);
      add(lb, BorderLayout.NORTH);
      add(pn, BorderLayout.SOUTH);

      // 登錄傾聽者
      addWindowListener(new SampleWindowListener());
      bt.addActionListener(new SampleActionListener());

      // 設定框架
      setSize(200, 200);
      setVisible(true);
   }

   // 傾聽者類別
   class SampleWindowListener extends WindowAdapter
   {
      public void windowClosing(WindowEvent e)
      {
         System.exit(0);
      }
   }
   class SampleActionListener implements ActionListener
   {
      public void actionPerformed(ActionEvent e)
      {
         lb.setText("謝謝惠顧。");
      }
   }
}
```

4.

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

public class SampleP4 extends JFrame
{
   JLabel lb;
   JMenuBar mb;
   JMenu mn[] = new JMenu[4];
   JMenuItem mi[] = new JMenuItem[6];
```

```
public static void main(String[] args)
{
     SampleP4 sm = new SampleP4();
}
public SampleP4()
{
   // 設定標題
   super("範例");

   // 建立元件
   lb = new JLabel("歡迎光臨。");
   mb = new JMenuBar();

   mn[0] = new JMenu("主選單1");
   mn[1] = new JMenu("主選單2");
   mn[2] = new JMenu("子選單1");
   mn[3] = new JMenu("子選單2");

   mi[0] = new JMenuItem("汽車");
   mi[1] = new JMenuItem("卡車");
   mi[2] = new JMenuItem("戰車");
   mi[3] = new JMenuItem("計程車");
   mi[4] = new JMenuItem("跑車");
   mi[5] = new JMenuItem("迷你車");

   mn[0].add(mi[0]);
   mn[0].add(mi[1]);

   mn[2].add(mi[2]);
   mn[2].add(mi[3]);

   mn[3].add(mi[4]);
   mn[3].add(mi[5]);

   mn[1].add(mn[2]);
   mn[1].addSeparator();
   mn[1].add(mn[3]);

   mb.add(mn[0]);
   mb.add(mn[1]);

   // 新增到容器中
   add(mb, BorderLayout.NORTH);
   add(lb, BorderLayout.CENTER);

   // 登錄傾聽者
   for(int i=0; i<mi.length; i++){
      mi[i].addActionListener(new SampleActionListener());
   }
   addWindowListener(new SampleWindowListener());
```

```
      // 設定框架
      setSize(200, 200);
      setVisible(true);
   }

   // 傾聽者類別
   class SampleActionListener implements ActionListener
   {
      public void actionPerformed(ActionEvent e)
      {
         JMenuItem tmp =(JMenuItem) e.getSource();
         String str = tmp.getText();
         lb.setText("是" + str + "吧。");
      }
   }
   class SampleWindowListener extends WindowAdapter
   {
      public void windowClosing(WindowEvent e)
      {
         System.exit(0);
      }
   }
}
```

5.

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

public class SampleP5 extends JApplet
{
   private JLabel lb;
   private JPanel pn;
   private JButton bt;

   public void init()
   {
      // 建立元件
      lb= new JLabel("歡迎光臨。");
      pn = new JPanel();
      bt = new JButton("購買");

      // 新增到容器中
      pn.add(bt);
      add(lb, BorderLayout.NORTH);
      add(pn, BorderLayout.SOUTH);

      // 登錄傾聽者
```

```
        bt.addActionListener(new SampleActionListener());
     }

     // 傾聽者類別
     class SampleActionListener implements ActionListener
     {
        public void actionPerformed(ActionEvent e)
        {
           String title = "感謝卡";
           String msg = "非常感謝您的購買。";
           int type = JOptionPane.INFORMATION_MESSAGE;

           JOptionPane.showMessageDialog
              (getContentPane(), msg, title, type);
        }
     }
  }
```

Chapter 22 Servlet

1.

SampleP1.html

```
<html>
<head><title>範例</title></head>
<body><center>
<h2>歡迎</h2>
<hr/>
請輸入您的大名。<br/>
<br/>
<form action="http://localhost:8080/YJKSample06/servlet/SampleP1"
method="GET">
<input type="text" name="name"/>
<input type="submit" value="傳送"/>
</form>
</center></body>
</html>
```

SampleP1.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class SampleP1 extends HttpServlet
{
   public void doGet(HttpServletRequest request,
                     HttpServletResponse response)
   throws ServletException
   {
      try{
         // 取得表單資料
         String tmp = request.getParameter("name");
         String name = new String(tmp.getBytes("8859_1"),
                        "Big5");

         // 設定內容類型
         response.setContentType("text/html; charset=Big5");

         // 輸出HTML文件
         PrintWriter pw = response.getWriter();
         if(name.length() != 0){
            pw.println("<html>\n" +
                       "<head><title>\n" + name +
                       "</title></head>\n" +
                       "<body><center>\n" +
                       "<h2>歡迎</h2>\n" +
                       "歡迎光臨，" + name + "先生。<br/>\n" +
                       "</center></body>\n" +
                       "</html>\n");
          }
          else{
             pw.println("<html>\n" +
                        "<head><title></title></head>\n" +
                        "<body><center>\n" +
                        "<h2>錯誤</h2>\n" +
                        "請輸入您的名字。<br/>\n" +
                        "</center></body>\n" +
                        "</html>\n");
          }
       }
       catch(Exception e){
          e.printStackTrace();
       }
   }
}
```

2.

SampleP2.html

```
<html>
<head><title>範例</title></head>
<body><center>
<img src="car.gif"/><br/>
```

```
<h2>歡迎</h2>
<hr/>
請選擇一項商品。<br/>
<br/>
<form action="http://localhost:8080/YJKSample06/servlet/SampleP2"
method="GET">
<input type="text" name="cars"/>
<input type="submit" value="傳送"/>
</form>
</center></body>
</html>
```

SampleP2.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class SampleP2 extends HttpServlet
{
   public void doGet(HttpServletRequest request,
                     HttpServletResponse response)
   throws ServletException
   {
      try{
         // 取得表單資料
         String tmp = request.getParameter("cars");
         String carname = new String(tmp.getBytes("8859_1"),
                           "Big5");

         // 設定內容類型
         response.setContentType("text/html; charset=Big5");

         // 輸出HTML文件
         PrintWriter pw = response.getWriter();
         if(carname.length() == 0){
             pw.println("<html>\n" +
                        "<head><title>錯誤</title></head>\n" +
                        "<body><center>\n" +
                        "<h2>錯誤</h2>\n" +
                        "請選擇一項產品。<br/>\n" +
                        "</center></body>\n" +
                        "</html>\n");
          }
         else if(carname.equals("計程車")){
            pw.println("<html>\n" +
                       "<head><title>\n" + carname +
                       "</title></head>\n" +
                       "<body><center>\n" +
                       "<h2>\n" + carname + "</h2>\n" +
                       "無法購買" + carname + "。<br/>\n" +
                       "</center></body>\n" +
                       "</html>\n");
          }
          else {
            pw.println("<html>\n" +
                       "<head><title>\n" + carname +
                       "</title></head>\n" +
                       "<body><center>\n" +
                       "<h2>\n" + carname + "</h2>\n" +
                       "感謝您購買" + carname + "。<br/>\n" +
                       "</center></body>\n" +
                       "</html>\n");
          }
       }
       catch(Exception e){
          e.printStackTrace();
       }
   }
}
```

Chapter 23 JSP

1.

SampleP1.html

```
<html>
<head><title>範例</title></head>
<body><center>
<h2>歡迎</h2>
<hr/>
請輸入您的大名。<br/>
<br/>
<form action="http://localhost:8080/YJKSample07/SampleP1.jsp"
method="GET">
<input type="text" name="name"/>
<input type="submit" value="傳送"/>
</form>
</center></body>
</html>
```

SampleP1.jsp

```
<%@ page contentType="text/html; charset=Big5" %>
<%
    String tmp = request.getParameter("name");
    String name = new String(tmp.getBytes("8859_1"),
                   "Big5");
%>

<html>
<head>
<title><%= name %></title>
</head>
<body>
<center>
<h2>歡迎</h2>
歡迎光臨，
<%= name%>
先生。<br/>
</center>
</body>
</html>
```

2.

```
<%@ page contentType="text/html; charset=Big5" %>
<%@ page import="java.util.*" %>

<%!
   HttpSession hs;
   Integer cn;
   Date dt;
   String str1, str2;
%>
<%
   // 取得session
   hs = request.getSession(true);
   cn = (Integer) hs.getAttribute("count");
   dt = (Date) hs.getAttribute("date");

   // 設定次數
   if(cn == null){
   cn = new Integer(1);
      dt = new Date();
      str1 = "這是您的初次造訪。";
      str2 = "";
   }
   else{
      cn = new Integer(cn.intValue() + 1);
      dt = new Date();
```

```
        str1 = "這是您的第" + cn + "次造訪。";
        str2 = "(上次是在：" + dt + ")";
    }

    // 設定Session
    hs.setAttribute("count", cn);
    hs.setAttribute("date", dt);
%>
<html>
<head>
<title>範例</title>
</head>
<body>
<center>

<h2>歡迎</h2>
<%= str1 %><br/>
<%= str2 %><br/>
請選擇一件商品。<br/>
<a href="car1.html">汽車</a><br/>
<a href="car2.html">卡車</a><br/>
<a href="car3.html">戰車</a><br/>
</center>
</body>
</html>
```

3.

SampleP3.html

```
<html>
<head><title>範例</title></head>
<body><center>
<img src="car.gif"/><br/>
<h2>歡迎</h2>
<hr/>
請選擇一件商品。<br/>
<br/>
<form action="http://localhost:8080/YJKSample07/servlet/SampleP3"
method="GET">
<input type="text" name="cars"/>
<input type="submit" value="傳送"/>
</form>
</center></body>
</html>
```

CarBean.java

```
package mybeans;
import java.io.*;

public class CarBean implements Serializable
{
   private String carname;
   private String cardata;

   public CarBean()
   {
      carname = null;
      cardata = null;
   }
   public void setCarname(String cn)
   {
      carname = cn;
   }
   public String getCardata()
   {
      return cardata;
   }
   public void makeCardata()
   {
      cardata = "車種:" + carname;
   }
}
```

SampleP3.java

```
import mybeans.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class SampleP3 extends HttpServlet
{
   public void doGet(HttpServletRequest request,
                     HttpServletResponse response)
   throws ServletException
   {
      try{
         // 取得表單資料
         String tmp = request.getParameter("cars");
         String carname = new String(tmp.getBytes("8859_1"),
                                    "Big5");

         // 建立Bean
         CarBean cb = new CarBean();
         cb.setCarname(carname);
         cb.makeCardata();

         // 設定請求
         request.setAttribute("cb", cb);

         // 取得Servlet Context
         ServletContext sc = getServletContext();

         // 轉交請求
         if(carname.length() == 0){
            sc.getRequestDispatcher("/error.html")
              .forward(request, response);
         }
         else if(carname.equals("計程車")){
            sc.getRequestDispatcher("/SampleP3T.jsp")
              .forward(request, response);
         }
         else{
            sc.getRequestDispatcher("/SampleP3.jsp")
              .forward(request, response);
         }
      }
      catch(Exception e){
         e.printStackTrace();
      }
   }
}
```

SampleP3.jsp

```
<%@ page contentType="text/html; charset=Big5" %>
<jsp:useBean id="cb" class="mybeans.CarBean" scope="request"/>

<html>
<head>
<title>範例</title>
</head>
<body>
<center>
<h2><jsp:getProperty name="cb" property="cardata"/>
</h2>
感謝您購買本公司的
<jsp:getProperty name="cb" property="cardata"/>
。<br/>
</center>
</body>
</html>
```

SampleP3T.jsp

```
<%@ page contentType="text/html; charset=Big5" %>
<jsp:useBean id="cb" class="mybeans.CarBean" scope="request"/>

<html>
<head>
<title>範例</title>
</head>
<body>
<center>
<h2>抱歉</h2>
您無法購買
<jsp:getProperty name="cb" property="cardata"/>
。<br/>
</center>
</body>
</html>
```

Chapter 24 JDBC

1.

```
import java.sql.*;

public class SampleP1
{
   public static void main(String[] args)
   {
      try{
         // 準備連線
         String url = "jdbc:derby:fooddb;create=true";
         String usr = "";
         String pw = "";

         // 連接到資料庫
         Connection cn
            = DriverManager.getConnection(url, usr, pw);

         // 準備查詢
         DatabaseMetaData dm = cn.getMetaData();
         ResultSet tb
            = dm.getTables(null, null, "水果資料表", null);

         Statement st = cn.createStatement();

         String qry1 = "CREATE TABLE 水果資料表(編號 int,
            名稱 varchar(50), 販賣店 varchar(50))";
         String[] qry2 = {
            "INSERT INTO 水果資料表 VALUES (1,'橘子','青山商店')",
            "INSERT INTO 水果資料表 VALUES (2,'蘋果','東京市場')",
            "INSERT INTO 水果資料表 VALUES (3,'香蕉','鈴木商店')",
            "INSERT INTO 水果資料表 VALUES (4,'草莓','東京市場')",
            "INSERT INTO 水果資料表 VALUES (5,'梨子','青山商店')",
            "INSERT INTO 水果資料表 VALUES (6,'栗子','橫濱百貨')",
            "INSERT INTO 水果資料表 VALUES (7,'桃子','橫濱百貨')",
            "INSERT INTO 水果資料表 VALUES (8,'鳳梨','佐藤商店')",
            "INSERT INTO 水果資料表 VALUES (9,'柿子','青山商店')",
            "INSERT INTO 水果資料表 VALUES (10,'西瓜','東京市場')"};
         String qry3 = "SELECT * FROM 水果資料表";

         if(!tb.next()){
            st.executeUpdate(qry1);
            for(int i=0; i<qry2.length; i++){
               st.executeUpdate(qry2[i]);
            }
         }

         // 進行查詢
         ResultSet rs = st.executeQuery(qry3);

         // 取得資料
         ResultSetMetaData rm = rs.getMetaData();
         int cnum = rm.getColumnCount();
         while(rs.next()){
            for(int i=1; i<=cnum; i++){
                System.out.print(rm.getColumnName(i) + ":"+
                                 rs.getObject(i) + "  ");
            }
            System.out.println("");
         }

         // 關閉連線
         rs.close();
         st.close();
         cn.close();
      }
      catch(Exception e){
         e.printStackTrace();
      }
   }
}
```

2.

```
import java.sql.*;

public class SampleP2
{
   public static void main(String[] args)
   {
      if(args.length != 3){
         System.out.println("參數的數量不對。");
         System.exit(1);
      }

      try{
         // 準備連線
         String url = "jdbc:derby:fooddb;create=true";
         String usr = "";
         String pw = "";

         // 連接到資料庫
         Connection cn
            = DriverManager.getConnection(url, usr, pw);

         // 準備查詢
         Statement st = cn.createStatement();
         String qry1 = "INSERT INTO 水果資料表 VALUES ("
            + args[0] + ", '"
            + args[1] + "','"
            + args[2] + "')";
         String qry2 = "SELECT * FROM 水果資料表";

         // 進行查詢
         st.executeUpdate(qry1);
         ResultSet rs = st.executeQuery(qry2);

         // 取得資料
         ResultSetMetaData rm = rs.getMetaData();
         int cnum = rm.getColumnCount();

         while(rs.next()){
            for(int i=1; i<=cnum; i++){
               System.out.print(rm.getColumnName(i) + ":" +
                                rs.getObject(i) + "  ");
            }
            System.out.println("");
         }

         // 關閉連線
         rs.close();
         st.close();
         cn.close();
      }
      catch(Exception e){
         e.printStackTrace();
      }
   }
}
```

3.

```
import java.util.*;
import java.sql.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.*;

public class SampleP3 extends JFrame
{
   private JTable tb;
   private JScrollPane sp;

   public static void main(String[] args)
   {
      SampleP3 sm = new SampleP3();
   }
   public SampleP3()
   {
      // 設定標題
      super("範例");

      // 建立元件
      tb = new JTable();
      sp = new JScrollPane(tb);

      // 新增到容器中
      add(sp);

      // 登錄傾聽者
      addWindowListener(new SampleWindowListener());

      // 設定框架
      setSize(300, 300);
      setVisible(true);

      try{
         // 準備連線
         String url = "jdbc:derby:fooddb;create=true";
         String usr = "";
         String pw = "";
         // 連接到資料庫
         Connection cn
            = DriverManager.getConnection(url, usr, pw);

         // 準備查詢
         Statement st = cn.createStatement();
         String qry = "SELECT * FROM 水果資料表";

         // 進行查詢
         ResultSet rs = st.executeQuery(qry);
         tb.setModel(new MyTableModel(rs));

         // 關閉連線
         rs.close();
         st.close();
         cn.close();
      }
      catch(Exception e){
         e.printStackTrace();
      }
   }

   // 傾聽者類別
   class SampleWindowListener extends WindowAdapter
   {
      public void windowClosing(WindowEvent e)
      {
         System.exit(0);
      }
   }

   // 模型類別
   class MyTableModel extends AbstractTableModel
   {
      private ArrayList<String> colname;
      private ArrayList<ArrayList> data;

      public MyTableModel(ResultSet rs)
      {
         try{

            // 取得欄位數量
            ResultSetMetaData rm = rs.getMetaData();
            int cnum = rm.getColumnCount();
            colname = new ArrayList<String>(cnum);

            // 取得欄位名稱
            for(int i=1; i<=cnum; i++){
               colname.add(rm.getColumnName(i));
            }
```

```
            // 取得列
            data = new ArrayList<ArrayList>();
            while(rs.next()){
               ArrayList<String> rowdata
                  = new ArrayList<String>();
               for(int i=1; i<=cnum; i++){
                  rowdata.add(rs.getObject(i).toString());
               }
               data.add(rowdata);
            }
         }
         catch(Exception e){
            e.printStackTrace();
         }
      }
      public int getRowCount()
      {
         return data.size();
      }
      public int getColumnCount()
      {
         return colname.size();
      }
      public Object getValueAt(int row, int column)
      {
         ArrayList rowdata = (ArrayList) (data.get(row));
         return rowdata.get(column);
      }
      public String getColumnName(int column)
      {
         return (String) colname.get(column);
      }
   }
}
```

Chapter 25 檔案操作

1. ① ×　　② ○　　③ ×

2.

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

public class SampleP2 extends JFrame
{
   private JPanel pn;
   private JLabel lb;
   private JTextArea ta;
   private JScrollPane sp;
   private JButton bt1, bt2;

   public static void main(String[] args)
   {
      SampleP2 sm = new SampleP2();
   }
   public SampleP2()
   {
      // 設定標題
      super("範例");

      // 建立元件
      lb = new JLabel("請選擇檔案。");
      ta = new JTextArea();
      sp = new JScrollPane(ta);

      pn = new JPanel();
      bt1 = new JButton("讀取");
      bt2 = new JButton("儲存");

      // 新增到容器中
      pn.add(bt1);
      pn.add(bt2);

      add(lb, BorderLayout.NORTH);
      add(sp, BorderLayout.CENTER);
      add(pn, BorderLayout.SOUTH);

      // 登錄傾聽者
      bt1.addActionListener(new SampleActionListener());
      bt2.addActionListener(new SampleActionListener());
      addWindowListener(new SampleWindowListener());

      // 設定框架
      setSize(600, 400);
      setVisible(true);
  }

   // 傾聽者類別
   class SampleActionListener implements ActionListener
   {
      public void actionPerformed(ActionEvent e)
      {
         Container cnt = getContentPane();
         JFileChooser fc = new JFileChooser();
         fc.setFileFilter(new MyFileFilter());
```

```
      try{
         if(e.getSource() == bt1){
            int res = fc.showOpenDialog(cnt);
            if(res == JFileChooser.APPROVE_OPTION){
               File fl = fc.getSelectedFile();
               BufferedReader br =
                  new BufferedReader(new FileReader(fl));
               ta.read(br, null);
               br.close();
            }
         }
         else if(e.getSource() == bt2){
            int res = fc.showSaveDialog(cnt);
            if(res == JFileChooser.APPROVE_OPTION){
               File fl = fc.getSelectedFile();
               BufferedWriter bw =
                  new BufferedWriter(new FileWriter(fl));
               ta.write(bw);
               bw.close();
           }
         }
      }
      catch(Exception ex){
         ex.printStackTrace();
      }
   }
   // 過濾器類別
   class MyFileFilter extends
      javax.swing.filechooser.FileFilter
   {
      public boolean accept(File f)
      {
         if(f.isDirectory()){
            return true;
         }

         String fn = f.getName();
         if(fn.toLowerCase().endsWith(".java")){
            return true;
         }
         return false;
      }
      public String getDescription()
      {
         return "java檔案";
      }
   }
}
class SampleWindowListener extends WindowAdapter
{
      public void windowClosing(WindowEvent e)
      {
         System.exit(0);
      }
   }
}
```

Chapter 26 XML

1.

```
import java.io.*;
import javax.xml.parsers.*;
import javax.xml.transform.*;
import javax.xml.transform.stream.*;
import javax.xml.transform.dom.*;
import org.w3c.dom.*;

public class SampleP1
{
   public static void main(String[] args) throws Exception
   {
      // 進行DOM的準備
      DocumentBuilderFactory dbf
         = DocumentBuilderFactory.newInstance();
      DocumentBuilder db
         = dbf.newDocumentBuilder();

      // 讀入文件
      Document doc
         = db.parse(new FileInputStream("Sample.xml"));

      // 建立新文件
      Document doc2 = db.newDocument();

      // 新增根元素
      Element root = doc2.createElement("車子");
      doc2.appendChild(root);

      // 取出元素
      NodeList lst = doc.getElementsByTagName("price");

      for(int i=0; i<lst.getLength(); i++){
         Node n = lst.item(i);
         for(Node ch = n.getFirstChild();
                  ch != null;
                  ch = ch.getNextSibling()){
```

```
            Element elm = doc2.createElement("價格");
            Text txt = doc2.createTextNode(ch.getNodeValue());
            elm.appendChild(txt);
            root.appendChild(elm);
         }
      }

      // 輸出文件
      TransformerFactory tff
         = TransformerFactory.newInstance();
      Transformer tf
         = tff.newTransformer();
      tf.setOutputProperty(OutputKeys.ENCODING, "UTF-8");
      tf.transform(new DOMSource(doc2),
                   new StreamResult("result.xml"));
      System.out.println("輸出到result.xml了。");
   }
}
```

2.

```
import java.io.*;
import javax.xml.parsers.*;
import javax.xml.transform.*;
import javax.xml.transform.stream.*;
import javax.xml.transform.dom.*;
import org.w3c.dom.*;

public class SampleP2
{
   public static void main(String[] args) throws Exception
   {
      // 進行DOM的準備
      DocumentBuilderFactory dbf
         = DocumentBuilderFactory.newInstance();
      DocumentBuilder db
         = dbf.newDocumentBuilder();

      // 建立新文件
      Document doc = db.newDocument();

      // 新增根元素
      Element root = doc.createElement("水果列表");
      doc.appendChild(root);

      // 新增元素
      Element fruit = doc.createElement("水果");
      root.appendChild(fruit);
      Element elm1 = doc.createElement("名稱");
      Text txt1 = doc.createTextNode("橘子");
      elm1.appendChild(txt1);
      fruit.appendChild(elm1);

      Element elm2 = doc.createElement("進貨店家");
      Text txt2 = doc.createTextNode("青山商店");
      elm2.appendChild(txt2);
      fruit.appendChild(elm2);

      // 輸出文件
      TransformerFactory tff
         = TransformerFactory.newInstance();
      Transformer tf
         = tff.newTransformer();
      tf.setOutputProperty(OutputKeys.ENCODING, "UTF-8");
      tf.transform(new DOMSource(doc),
                   new StreamResult("result.xml"));
      System.out.println("輸出到result.xml了。");
   }
}
```

Chapter 27 網路

1. ① ×　② ○　③ ×

2.

```
import java.io.*;
import java.net.*;

public class SamplePS
{
   public static void main(String[] args)
   {
      SamplePS sm = new SamplePS();

      if(args.length != 1){
         System.out.println("參數的數量不對。");
         System.exit(1);
      }

      try{
         ServerSocket ss =
            new ServerSocket(Integer.parseInt(args[0]));

         System.out.println("正在等待。");
         while(true){
            Socket sc = ss.accept();
            System.out.println("歡迎。");

            PrintWriter pw = new PrintWriter
               (new BufferedWriter
               (new OutputStreamWriter(sc.getOutputStream())));
            pw.println("這裡是伺服器。");
            pw.flush();
            pw.close();

            sc.close();
         }
      }
      catch(Exception e){
         e.printStackTrace();
      }
   }
}
```

3.

```
import java.io.*;
import java.awt.*;
import java.awt.event.*;
import java.net.*;
import javax.swing.*;

public class SamplePC extends JFrame
{
   private JLabel lb1, lb2;
   private JTextField tf1, tf2;
   private JTextArea ta;
   private JPanel pn1, pn2;
   private JButton bt;

   public static void main(String[] args)
   {
      SamplePC sm = new SamplePC();
   }
   public SamplePC()
   {
      // 設定標題
      super("範例");

      // 建立元件
      lb1 = new JLabel("主機");
      lb2 = new JLabel("埠");
      tf1 = new JTextField();
      tf2 = new JTextField();
      ta = new JTextArea();
      pn1 = new JPanel();
      pn2 = new JPanel();
      bt = new JButton("連線");

      // 設定容器
      pn1.setLayout(new GridLayout(2,2));

      // 新增到容器中
      pn1.add(lb1);
      pn1.add(tf1);
      pn1.add(lb2);
      pn1.add(tf2);
      pn2.add(bt);

      add(pn1, BorderLayout.NORTH);
      add(ta, BorderLayout.CENTER);
      add(pn2, BorderLayout.SOUTH);

      // 登錄傾聽者
      bt.addActionListener(new SampleActionListener());
      addWindowListener(new SampleWindowListener());

      // 設定框架
      setSize(300, 200);
      setVisible(true);

   }

   // 傾聽者類別
   class SampleActionListener implements ActionListener
   {
      public void actionPerformed(ActionEvent e)
      {
         try{
            InetAddress ia
               = InetAddress.getByName(tf1.getText());
            String host = ia.getHostName();
            int port = Integer.parseInt(tf2.getText());

            Socket sc = new Socket(host, port);
            BufferedReader  br = new BufferedReader
                 (new InputStreamReader(sc.getInputStream()));
            String str = br.readLine();
            ta.setText(str);
            br.close();
            sc.close();
         }
         catch(Exception ex){
            ex.printStackTrace();
         }
      }
   }
   class SampleWindowListener extends WindowAdapter
   {
      public void windowClosing(WindowEvent e)
      {
          System.exit(0);
      }
   }
}
```