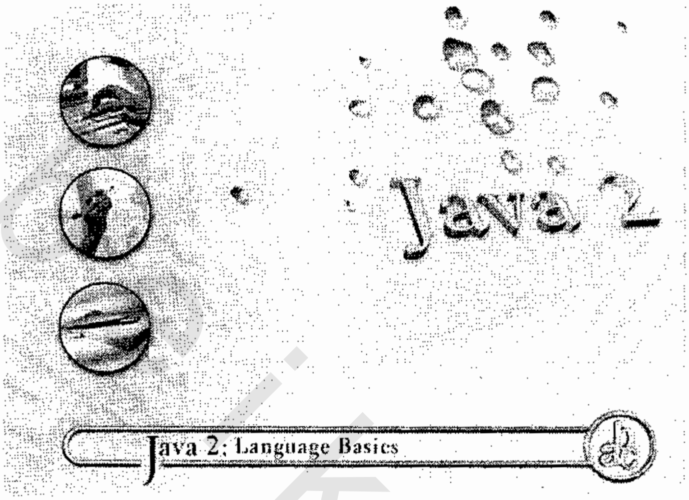
الفصل العشرون



تطبيقات النصوص والتواريخ والمجموعات

String, Date, and Collections Applications

في هذا الفصل نتعرف علي العديد من التطبيقات الهامة للغة Java حيث نتعرف علي العديد من تطبيقات النصوص والتواريخ والمجموعات Collections.

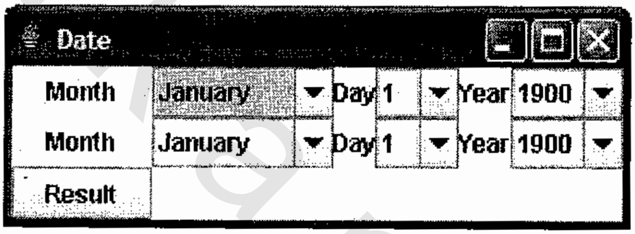
مقدمة:

نتعلم في هذا الفصل مشاريع النصوص والتواريخ والمجموعات Collections.

مثال (1): التعامل مع التواريخ:

أولاً: هدف المثال:

اكتب برنامجاً يقوم بإدخال تاريخين في أداة إطار JFrame كما هو واضح في الشكل (20-1) ثم يقوم بطباعة التاريخ الأحدث ، كما يجب أن يتعامل البرنامج مع السنوات الكبيسة والأشهر ذات الثلاثين يوماً وذات الحادي والثلاثين يوماً.



Month January Day 1 Year 1900

Month January Day 1 Year 1900

Result

الشكل (20-1) تنفيذ البرنامج

ثانياً: كود البرمجة:

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

public class MyDate extends JFrame
{
 public static final String []month28 = {"1","2","3","4","5","6","7","8","9","10","11","12","13","14","15","16",
  "17","18","19","20","21","22","23","24","25","26","27","28"};
```

```

public static final String []month29 = {"1","2","3","4","5","6","7","8","9","10","11","12","13","14","15","16",
 "17","18","19","20","21","22","23","24","25","26","27","28","29"};

public static final String []month30 = {"1","2","3","4","5","6","7","8","9","10","11","12","13","14","15","16",
 "17","18","19","20","21","22","23","24","25","26","27","28","29","30"};

public static final String []month31 = {"1","2","3","4","5","6","7","8","9","10","11","12","13","14","15","16",
 "17","18","19","20","21","22","23","24","25","26","27","28","29","30","31"};

public static final String []monthNames = {"January","February","March","April","May","June",
 "July","August","September","October","November","December"};

 public static final int startYear = 1900;
 public static final int endYear = 3000;

 public static String[] yearValues;

 JLabel labelMonth1,labelDay1,labelYear1,labelMonth2,labelDay2,labelYear2;
 JComboBox comboMonth1,comboDay1,comboYear1,comboMonth
```

```
  2,comboDay2,comboYear2;
JButton result;
GridBagLayout grid;
GridBagConstraints gbc;

MyDate()
{
     addWindowListener(new WindowAdapter()
          {
               public void windowClosing(WindowEvent
  we)
               {
                    dispose();
                    System.exit(0);
               }
          }
     );

     yearValues = new String[endYear - startYear + 1];
     for ( int i = 0 ; i <= endYear - startYear ; i++ )
     {
          yearValues[i] = Integer.toString(startYear + i);
     }

     Container c = getContentPane();

     grid = new GridBagLayout();
     gbc = new GridBagConstraints();
     c.setLayout(grid);

     labelMonth1 = new JLabel("Month");
     labelDay1 = new JLabel("Day");
     labelYear1 = new JLabel("Year");
     labelMonth2 = new JLabel("Month");
```

```
        labelDay2 = new JLabel("Day");
        labelYear2 = new JLabel("Year");

        MyComboListener myComboListener = new MyComboListener();

        comboMonth1 = new JComboBox(monthNames);
        comboMonth1.addItemListener(myComboListener);
        comboDay1 = new JComboBox(month31);
        comboDay1.addItemListener(myComboListener);
        comboYear1 = new JComboBox(yearValues);
        comboYear1.addItemListener(myComboListener);
        comboMonth2 = new JComboBox(monthNames);
        comboMonth2.addItemListener(myComboListener);
        comboDay2 = new JComboBox(month31);
        comboDay2.addItemListener(myComboListener);
        comboYear2 = new JComboBox(yearValues);
        comboYear2.addItemListener(myComboListener);

        ButtonListener b = new ButtonListener();

        result = new JButton("Result");
        result.addActionListener(b);

        gbc.gridx = 1;
        gbc.gridy = 1;
        grid.setConstraints(labelMonth1,gbc);
        c.add(labelMonth1);

        gbc.gridx = 2;
        gbc.gridy = 1;
        grid.setConstraints(comboMonth1,gbc);
```

```
c.add(comboMonth1);

gbc.gridx = 3;
gbc.gridy = 1;
grid.setConstraints(labelDay1,gbc);
c.add(labelDay1);

gbc.gridx = 4;
    gbc.gridy = 1;
    grid.setConstraints(comboDay1,gbc);
    c.add(comboDay1);

    gbc.gridx = 5;
    gbc.gridy = 1;
    grid.setConstraints(labelYear1,gbc);
    c.add(labelYear1);

    gbc.gridx = 6;
    gbc.gridy = 1;
    grid.setConstraints(comboYear1,gbc);
    c.add(comboYear1);

    gbc.gridx = 1;
    gbc.gridy = 2;
    grid.setConstraints(labelMonth2,gbc);
    c.add(labelMonth2);

    gbc.gridx = 2;
    gbc.gridy = 2;
    grid.setConstraints(comboMonth2,gbc);
    c.add(comboMonth2);

    gbc.gridx = 3;
    gbc.gridy = 2;
```

```
        grid.setConstraints(labelDay2,gbc);
        c.add(labelDay2);

        gbc.gridx = 4;
        gbc.gridy = 2;
        grid.setConstraints(comboDay2,gbc);
        c.add(comboDay2);

        gbc.gridx = 5;
        gbc.gridy = 2;
        grid.setConstraints(labelYear2,gbc);
        c.add(labelYear2);

        gbc.gridx = 6;
        gbc.gridy = 2;
        grid.setConstraints(comboYear2,gbc);
        c.add(comboYear2);

        gbc.gridx = 1;
        gbc.gridy = 3;
        grid.setConstraints(result,gbc);
        c.add(result);

        setTitle("Date");
        pack();
        setVisible(true);
    }

    public static void main(String args[])
    {
        MyDate myDate = new MyDate();
    }

    class ButtonListener implements ActionListener
    {
        public void actionPerformed(ActionEvent e)
        {
            Object obj = e.getSource();

            String strmonth1 = "";
            int month1 = 0;
            int day1 = 0;
            int year1 = 0;
            String strmonth2 = "";
            int month2 = 0;
            int day2 = 0;
            int year2 = 0;

                try
                {
                    strmonth1 = (String)comboMonth1.getSelectedItem();
                    month1 = getMonthOrder(strmonth1);
                    day1 = Integer.parseInt((String)comboDay1.getSelectedItem());
                    year1 = Integer.parseInt((String)comboYear1.getSelectedItem());
                    strmonth2 = (String)comboMonth2.getSelectedItem();
                    month2 = getMonthOrder(strmonth2);
                    day2 = Integer.parseInt((String)comboDay2.getSelectedItem());
                    year2 =
Integer.parseInt((String)comboYear2.getSelectedItem()
);
                }

                catch(Exception ex)
                {
JOptionPane.showMessageDialog(MyDate.this,"Wron
g Date Values");
                    return;
                }

            if ( obj == result )
            {
                int newMonth = 0;
                int newDay = 0;
                int newYear = 0;

                if ( year1 > year2 )
                {
                    newMonth = month1;
                    newDay = day1;
                    newYear = year1;
                }

                else if ( year1 < year2 )
                {
                    newMonth = month2;
                    newDay = day2;
                    newYear = year2;
                }

                else if ( year1 == year2 )
                {
```

```
if ( month1 > month2 )
{
    newMonth = month1;
    newDay = day1;
    newYear = year1;
}

else if ( month1 <month2 )
{
    newMonth = month2;
    newDay = day2;
    newYear = year2;
}

else if ( month1 == month2 )
{
    if ( day1 > day2 )
    {
        newMonth = month1;
        newDay = day1;
        newYear = year1;
    }

    else if ( day1 < day2 )
    {
        newMonth = month2;
        newDay = day2;
        newYear = year2;
    }

    else if ( day1 == day2 )
    {
        newMonth = month1;
        newDay = day1;
```

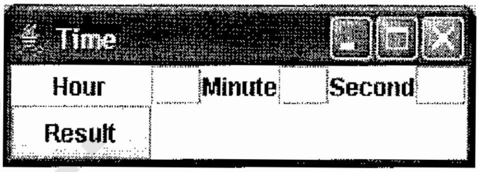
```
                            newYear = year1;
                        }
                    }
                }

                String message = "New Date is " + newDay + "/" + newMonth + "/" + newYear;
                JOptionPane.showMessageDialog(MyDate.this,message);
            }

        }
    }

    public int getMonthOrder(String monthName)
    {
        int order = 1;

        for ( int i = 0 ; i < monthNames.length ; i++ )
        {
            if ( monthNames[i].equals(monthName) )
            {
                order = i + 1;
            }
        }

        return order;
    }

    class MyComboListener implements ItemListener
    {
        public void itemStateChanged(ItemEvent e)
        {
            Object source = e.getSource();

            if ( source == comboDay1 || source == comboDay2 )
            {
                return;
            }

            else if ( source == comboMonth1 || source == comboYear1 )
            {
                updateComboValues(comboDay1,comboMonth1,comboYear1);
            }

            else if ( source == comboMonth2 || source == comboYear2 )
            {
                updateComboValues(comboDay2,comboMonth2,comboYear2);
            }
        }
    }

    public void updateComboValues(JComboBox comboDay,JComboBox comboMonth,JComboBox comboYear)
    {
        try
        {
            String strmonth = (String)comboMonth.getSelectedItem();
        int month = getMonthOrder(strmonth);

        if ( month == 1 || month == 3 || month == 5 || month == 7 || month == 8 || month == 10 || month == 12 )
        {
            comboDay.removeAllItems();

            for ( int i = 0 ; i < month31.length ; i++ )
            {
                comboDay.addItem(month31[i]);
            }
        }

        else if ( month == 4 || month == 6 || month == 9 || month == 11 )
        {
            comboDay.removeAllItems();

            for ( int i = 0 ; i < month30.length ; i++ )
            {
                comboDay.addItem(month30[i]);
            }
        }

        else if ( month == 2 )
        {
            comboDay.removeAllItems();

            int year = Integer.parseInt((String)comboYear.getSelectedItem());

```

```
                if ( year % 4 == 0 )
                {
                    for ( int i = 0 ; i < month29.length ; i++ )
                    {
                        comboDay.addItem(month29[i]);
                    }
                }

                else
                {
                    for ( int i = 0 ; i < month28.length ; i++ )
                    {
                        comboDay.addItem(month28[i]);
                    }
                }
            }
        }

        catch(Exception e)
        {
            return;
        }
    }
}
```

مثال (2): التعامل مع الوقت:

أولاً: هدف المثال:

اكتب برنامجاً يقوم بإدخال الزمن بالساعة والدقيقة والثانية في أداة إطار JFrame كما هو واضح في الشكل (20-2) ثم يقوم بإضافة ثانية واحدة علي هذا الزمن ثم طباعة الزمن الجديد.



الشكل (20-2) تنفيذ البرنامج

ثانياً: كود البرمجة:

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

public class MyTime extends JFrame
{
  JLabel labelHour,labelMinute,labelSecond;
  JTextField textHour,textMinute,textSecond;
  JButton result;
  GridBagLayout grid;
  GridBagConstraints gbc;

  MyTime()
  {
       addWindowListener(new WindowAdapter()
            {
                 public void windowClosing(WindowEvent we)
                 {
                    dispose();
                    System.exit(0);
                }
            }
        );

        Container c = getContentPane();

        grid = new GridBagLayout();
        gbc = new GridBagConstraints();
        c.setLayout(grid);

        labelHour = new JLabel("Hour");
        labelMinute = new JLabel("Minute");
        labelSecond = new JLabel("Second");

        textHour = new JTextField(2);
        textMinute = new JTextField(2);
        textSecond = new JTextField(2);

        ButtonListener b = new ButtonListener();

        result = new JButton("Result");
        result.addActionListener(b);

        gbc.gridx = 1;
        gbc.gridy = 1;
        grid.setConstraints(labelHour,gbc);
        c.add(labelHour);

        gbc.gridx = 2;
        gbc.gridy = 1;
        grid.setConstraints(textHour,gbc);
        c.add(textHour);

```

```
    gbc.gridx = 3;
    gbc.gridy = 1;
    grid.setConstraints(labelMinute,gbc);
    c.add(labelMinute);

    gbc.gridx = 4;
    gbc.gridy = 1;
    grid.setConstraints(textMinute,gbc);
    c.add(textMinute);

    gbc.gridx = 5;
    gbc.gridy = 1;
    grid.setConstraints(labelSecond,gbc);
    c.add(labelSecond);

    gbc.gridx = 6;
    gbc.gridy = 1;
    grid.setConstraints(textSecond,gbc);
    c.add(textSecond);

    gbc.gridx = 1;
    gbc.gridy = 2;
    grid.setConstraints(result,gbc);
    c.add(result);

    setTitle("Time");
    pack();
    setVisible(true);
}

public static void main(String args[])
{
    MyTime myTime = new MyTime();
}

```

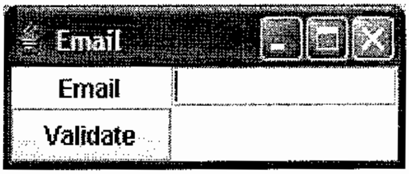
```
class ButtonListener implements ActionListener
{
    public void actionPerformed(ActionEvent e)
    {
        Object obj = e.getSource();

        int hour = 0;
        int minute = 0;
        int second = 0;

                try
                {
                    hour = Integer.parseInt(textHour.getText());
                    minute = Integer.parseInt(textMinute.getText());
                    second = Integer.parseInt(textSecond.getText());
                }

                catch(Exception ex)
                {
                    JOptionPane.showMessageDialog(MyTime.this,"Wrong Time Values");
                    return;
                }

            if ( obj == result )
            {
                Calendar time = Calendar.getInstance();
                time.set(2006,8,30,hour,minute,second);
                time.add(Calendar.SECOND,1);
```

```

                    int newHour = time.get(Calendar.HOUR);
                    int newMinute = time.get(Calendar.MINUTE);
                    int newSecond = time.get(Calendar.SECOND);

                    String message = "New Time is " + newHour + ":" + newMinute + ":" + newSecond;
                    JOptionPane.showMessageDialog(MyTime.this,message);
                }
            }
    }
}
```

مثال (3): التعامل مع النصوص:

أولاً: هدف المثال:

في هذا المثال نريد إنشاء برنـامج يحتـوي علـي أداة إطـار JFrame كمـا هـو واضـح في الشكل (20-3) ، وعلي المستخدم إدخال بريده الإليكتروني E-mail ، ونريد التأكد من أن بريده الإليكتروني E-mail يحتوي علي الحرف @ وأنه ينتهـي بـالحروف (.com) ، وبعد التأكد مـن صـحة إدخـال المستخدم ، فإننـا نظهـر رسـالة لنـبين اسـم المسـتخدم User Name واسم الشركة Company Name.



الشكل (20-3) تنفيذ البرنامج

ثانياً: كود البرمجة:

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

public class MyEmail extends JFrame
{
 JLabel labelEmail;
 JTextField textEmail;
 JButton validate;
 GridBagLayout grid;
 GridBagConstraints gbc;

 MyEmail()
 {
     addWindowListener(new WindowAdapter()
         {
             public void windowClosing(WindowEvent we)
             {
                 dispose();
                 System.exit(0);
             }
         }
     );

     Container c = getContentPane();

     grid = new GridBagLayout();
     gbc = new GridBagConstraints();
     c.setLayout(grid);

     labelEmail = new JLabel("Email");

```

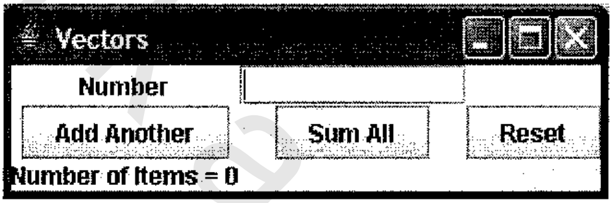
```
textEmail = new JTextField(10);

textEmail.addKeyListener(new KeyAdapter()
    {
        public void keyReleased(KeyEvent e)
        {
            char c = e.getKeyChar();

            if ( c == '\n' )
            {
                validate.doClick();
            }
        }
    }
);

ButtonListener b = new ButtonListener();

validate = new JButton("Validate");
validate.addActionListener(b);

gbc.gridx = 1;
gbc.gridy = 1;
grid.setConstraints(labelEmail,gbc);
c.add(labelEmail);

gbc.gridx = 2;
gbc.gridy = 1;
grid.setConstraints(textEmail,gbc);
c.add(textEmail);

gbc.gridx = 1;
gbc.gridy = 2;
grid.setConstraints(validate,gbc);
```

```
        c.add(validate);

        setTitle("Email");
        pack();
        setVisible(true);
    }

    public static void main(String args[])
    {
        MyEmail myEmail = new MyEmail();
    }

    class ButtonListener implements ActionListener
    {
        public void actionPerformed(ActionEvent e)
        {
            Object obj = e.getSource();

            if ( obj == validate )
            {
                String email = textEmail.getText();

                if ( email == null || email.trim().equals("") )
                {
                    JOptionPane.showMessageDialog(MyEmail.this,"Please enter your email");
                    textEmail.selectAll();
                    textEmail.requestFocus();
                    return;
                }

                if ( email.indexOf("@",0) < 0 )
            {
                JOptionPane.showMessageDialog(MyEmail.this,"The email should contain '@' character");
                    textEmail.selectAll();
                    textEmail.requestFocus();
                    return;
                }

                else if ( email.lastIndexOf(".com",20) < 0 )
                {
                JOptionPane.showMessageDialog(MyEmail.this,"The email should end with .com");
                    textEmail.selectAll();
                    textEmail.requestFocus();
                    return;
                }

                else
                {
                    String username = email.substring(0,email.indexOf("@",0));
                    String companyname = email.substring(email.indexOf("@",0) + 1,email.length());
                JOptionPane.showMessageDialog(MyEmail.this,"User Name: " + username + "\n" + "Company Name: " + companyname);
                }
            }
        }
    }
}
```

مثال (4): التعامل مع المجموعات Collections:

أولاً: هدف المثال:

في هذا المثال نريد إنشاء برنامج يحتوي علي أداة إطار JFrame كما هو واضح في الشكل (20-4) ، بحيث يقوم المستخدم بإدخال رقم ثم يضغط علي الزر Add Another ليقوم البرنامج بحفظ الرقم في الذاكرة ، ويظل المستخدم يدخل الأرقام ويقوم البرنامج بحفظ جميع الأرقام الجديدة ويوضح عدد الأرقام الموجود في الذاكرة ، فإذا رغب المستخدم في جمع الأرقام الموجودة في الذاكرة فإنه يضغط علي الزر Sum All ، أما إذا رغب في مسح الموجود والبدء من جديد فإنه يضغط علي الزر Reset.



الشكل (20-4) تنفيذ البرنامج

ثانياً: كود البرمجة:

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

public class MyNumbers extends JFrame
{
  JLabel labelNum,labelNumberOfNumbers;
  JTextField textNum;
  JButton addAnother,sumAll,reset;
  Vector vectorNumbers = new Vector();
  GridBagLayout grid;
  GridBagConstraints gbc;

```

```
MyNumbers()
{
    addWindowListener(new WindowAdapter()
        {
            public void windowClosing(WindowEvent we)
            {
                dispose();
                System.exit(0);
            }
        }
    );

    Container c = getContentPane();

    grid = new GridBagLayout();
    gbc = new GridBagConstraints();
    c.setLayout(grid);

    labelNum = new JLabel("Number");
    labelNumberOfNumbers = new JLabel("Number of Items = 0");

    textNum = new JTextField(10);

    ButtonListener b = new ButtonListener();

    addAnother = new JButton("Add Another");
    addAnother.addActionListener(b);
    sumAll = new JButton("Sum All");
    sumAll.addActionListener(b);
    reset = new JButton("Reset");
    reset.addActionListener(b);

    gbc.gridx = 1;
    gbc.gridy = 1;
    grid.setConstraints(labelNum,gbc);
    c.add(labelNum);

    gbc.gridx = 2;
    gbc.gridy = 1;
    grid.setConstraints(textNum,gbc);
    c.add(textNum);

    gbc.gridx = 1;
    gbc.gridy = 2;
    grid.setConstraints(addAnother,gbc);
    c.add(addAnother);

    gbc.gridx = 2;
    gbc.gridy = 2;
    grid.setConstraints(sumAll,gbc);
    c.add(sumAll);

    gbc.gridx = 3;
    gbc.gridy = 2;
    grid.setConstraints(reset,gbc);
    c.add(reset);

    gbc.gridx = 1;
    gbc.gridy = 3;
    grid.setConstraints(labelNumberOfNumbers,gbc);
    c.add(labelNumberOfNumbers);

    setTitle("Vectors");
    pack();
    setVisible(true);
}
```

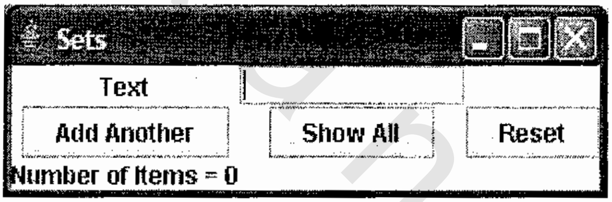
```

public static void main(String args[])
{
    MyNumbers myNumbers = new MyNumbers();
}

class ButtonListener implements ActionListener
{
    public void actionPerformed(ActionEvent e)
    {
        Object obj = e.getSource();

        if ( obj == reset )
        {
            vectorNumbers.clear();
            labelNumberOfNumbers.setText("Number of Items = " + vectorNumbers.size());
        }

        else if ( obj == sumAll )
        {
            double sum = 0;
            for ( int i = 0 ; i < vectorNumbers.size() ; i++ )
            {
                double n = ((Double)vectorNumbers.get(i)).doubleValue();
                sum += n;
            }

            JOptionPane.showMessageDialog(MyNumbers.this,"Sum = " + sum);
        }
```

```

            if ( obj == addAnother )
            {
                double num = 0;

                try
                {
                    num = Double.parseDouble(textNum.getText());
                }

                catch(Exception ex)
                {
                    JOptionPane.showMessageDialog(MyNumbers.this,"Please Enter a Number");
                    textNum.selectAll();
                    textNum.requestFocus();
                    return;
                }

                Double d = new Double(num);
                vectorNumbers.add(d);
                labelNumberOfNumbers.setText("Number of Items = " + vectorNumbers.size());
                textNum.setText("");
                textNum.requestFocus();
            }
        }
    }
}
```

مثال (5): طباعة القيم الفريدة Unique:

أولاً: هدف المثال:

في هذا المثال نريد إنشاء برنامج يحتوي علي أداة إطار JFrame كما هو واضح في الشكل (20-5)، بحيث يقوم المستخدم بإدخال نص ثم يضغط علي الزر Add Another ليقوم البرنامج بحفظ النص في الذاكرة بشرط ألا يكون قد سبق للمستخدم إدخال هذا النص من قبل، وفي هذه الحالة يطبع البرنامج رسالة خطأ للمستخدم، ويظل المستخدم يدخل النصوص ويقوم البرنامج بحفظ جميع النصوص الجديدة ويوضح عدد النصوص الموجود في الذاكرة، فإذا رغب المستخدم في عرض النصوص الموجودة في الذاكرة فإنه يضغط علي الزر Show All، أما إذا رغب في مسح الموجود والبدء من جديد فإنه يضغط علي الزر Reset.



الشكل (20-5) تنفيذ البرنامج

ثانياً: كود البرمجة:

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

public class MyStrings extends JFrame
{
  JLabel labelString,labelNumberOfNumbers;
  JTextField text;
  JButton addAnother,showAll,reset;
  TreeSet set = new TreeSet();
```

```
GridBagLayout grid;
GridBagConstraints gbc;

MyStrings()
{
    addWindowListener(new WindowAdapter()
        {
            public void windowClosing(WindowEvent we)
            {
                dispose();
                System.exit(0);
            }
        }
    );

    Container c = getContentPane();

    grid = new GridBagLayout();
    gbc = new GridBagConstraints();
    c.setLayout(grid);

    labelString = new JLabel("Text");
    labelNumberOfNumbers = new JLabel("Number of Items = 0");

    text = new JTextField(10);

    ButtonListener b = new ButtonListener();

    addAnother = new JButton("Add Another");
    addAnother.addActionListener(b);
    showAll = new JButton("Show All");
    showAll.addActionListener(b);
    reset = new JButton("Reset");
```

```
reset.addActionListener(b);

gbc.gridx = 1;
gbc.gridy = 1;
grid.setConstraints(labelString,gbc);
c.add(labelString);

gbc.gridx = 2;
gbc.gridy = 1;
grid.setConstraints(text,gbc);
c.add(text);

gbc.gridx = 1;
gbc.gridy = 2;
grid.setConstraints(addAnother,gbc);
c.add(addAnother);

gbc.gridx = 2;
gbc.gridy = 2;
grid.setConstraints(showAll,gbc);
c.add(showAll);

gbc.gridx = 3;
gbc.gridy = 2;
grid.setConstraints(reset,gbc);
c.add(reset);

gbc.gridx = 1;
gbc.gridy = 3;
grid.setConstraints(labelNumberOfNumbers,gbc);
c.add(labelNumberOfNumbers);

setTitle("Sets");
pack();
```

```
    setVisible(true);
}

public static void main(String args[])
{
    MyStrings myStrings = new MyStrings();
}

class ButtonListener implements ActionListener
{
    public void actionPerformed(ActionEvent e)
    {
        Object obj = e.getSource();

        if ( obj == reset )
        {
            set.clear();
            labelNumberOfNumbers.setText("Number
of Items = " + set.size());
        }

        else if ( obj == showAll )
            {
                String message = "";

                Iterator iterator = set.iterator();

                while ( iterator.hasNext() )
                {
                    message += iterator.next() + "\n";
                }


JOptionPane.showMessageDialog(MyStrings.this,mes
sage);
            }

            if ( obj == addAnother )
            {
                String inputText = text.getText();

                if ( set.contains(inputText) )
                {
                    JOptionPane.showMessageDialog(MyStrings.this,"This text was entered before\nand will not be added");
                    return;
                }

                set.add(inputText);
                labelNumberOfNumbers.setText("Number of Items = " + set.size());
                text.setText("");
                text.requestFocus();
            }
        }
    }
}
```

ملخص الفصل:

- تعلمنا في هذا الفصل العديد من تطبيقات النصوص والتواريخ والمجموعات Collections.

- في الفصل القادم نتعلم – بإذن الله – العديد من التطبيقات الأخري ، فتابع معنا الفصل القادم.