```
 1  /*
 2      Chapter 5:  Reserve a Party Room
 3      Programmer: Mike Brown
 4      Date:       March 30, 2009
 5      Filename:   Reservations.java
 6      Purpose:    This Program creates a windowed application to reserve a
        party room.
 7              It calls an external class named Rooms.
 8  */
 9
10  import javax.swing.JOptionPane;
11  import java.awt.*;
12  import java.awt.event.*;
13
14  public class Reservations extends Frame implements ActionListener
15  {
16      Color lightRed = new Color(255,90,90);
17      Color lightGreen = new Color(140,215,40);
18
19      Rooms room = new Rooms(5,3);
20
21      Panel roomPanel = new Panel();
22          TextArea roomDisplay[] = new TextArea[9];
23
24      Panel buttonPanel = new Panel();
25          Button bookButton = new Button("Book Room");
26
27      Panel inputPanel = new Panel();
28          Label custNameLabel = new Label ("Name");
29          TextField nameField = new TextField(15);
30          Label custPhoneLabel = new Label("Phone Number");
31          TextField phoneField = new TextField(15);
32          Label numLabel = new Label("Number in party:");
33          Choice numberOfGuests = new Choice();
34          CheckboxGroup options = new CheckboxGroup();
35              Checkbox nonSmoking = new Checkbox("Nonsmoking", false,
                options);
36              Checkbox smoking = new Checkbox("Smoking", false, options);
37              Checkbox hidden = new Checkbox("", true, options);
38
39      public Reservations()
40      {
41          //set Layouts for frame and three panels
42          this.setLayout(new BorderLayout());
43              roomPanel.setLayout(new GridLayout(2,4,10,10));
44              buttonPanel.setLayout(new FlowLayout());
45              inputPanel.setLayout(new FlowLayout());
46
47          //add components to room panel
48          for (int i=1; i<9; i++)
49          {
50              roomDisplay[i] = new TextArea(null,3,5,3);
51              if(i<6)
52                  roomDisplay[i].setText("Room " + i + " Nonsmoking");
53              else
54                  roomDisplay[i].setText("Room " + i + " Smoking");
55              roomDisplay[i].setEditable(false);
56              roomDisplay[i].setBackground(lightGreen);
57              roomPanel.add(roomDisplay[i]);
58          }
59
60          //add components to button panel
61          buttonPanel.add(bookButton);
62
63          //add components to input panel
64          inputPanel.add(custNameLabel);
65          inputPanel.add(nameField);
```

```
 66          inputPanel.add(custPhoneLabel);
 67          inputPanel.add(phoneField);
 68          inputPanel.add(numLabel);
 69          inputPanel.add(numberOfGuests);
 70              for(int i=8; i<=20; i++)
 71                  numberOfGuests.add(String.valueOf(i));
 72          inputPanel.add(nonSmoking);
 73          inputPanel.add(smoking);
 74
 75          //add panels to frame
 76          add(buttonPanel, BorderLayout.SOUTH);
 77          add(inputPanel, BorderLayout.CENTER);
 78          add(roomPanel, BorderLayout.NORTH);
 79
 80          bookButton.addActionListener(this);
 81
 82          //overriding the windowClosing() method will allow the user to
             click the Close button
 83          addWindowListener(
 84              new WindowAdapter()
 85              {
 86                  public void windowClosing(WindowEvent e)
 87                  {
 88                      System.exit(0);
 89                  }
 90              }
 91          );
 92      }//end of constructor method
 93
 94      public static void main(String[] args)
 95      {
 96          Reservations f = new Reservations();
 97          f.setBounds(200,200,600,300);
 98          f.setTitle("Reserve a Party Room");
 99          f.setVisible(true);
100      }//end of main
101
102      public void actionPerformed(ActionEvent e)
103      {
104          if (hidden.getState())
105          {
106              JOptionPane.showMessageDialog(null, "You must select
                 Nonsmoking or Smoking.", "Error", JOptionPane.ERROR_MESSAGE);
107          }
108          else
109          {
110              int available = room.bookRoom(smoking.getState());
111
112              if(available > 0) //room is available
113              {
114                  roomDisplay[available].setBackground(lightRed); //display
                     room as occupied
115                  roomDisplay[available].setText(
116                                                      roomDisplay[available].
                                                        getText() +
117                                                      "\n" +
118                                                      nameField.getText() +
119                                                      " " +
120                                                      phoneField.getText() +
121                                                      "\nparty of " +
122                                                      numberOfGuests.
                                                        getSelectedItem()
123                                                  ); //display info in room
124                  clearFields();
125              }
126              else //room is not available
127              {
```

```
128                 if(smoking.getState())
129                     JOptionPane.showMessageDialog(null, "Smoking is
                        full.", "Error", JOptionPane.INFORMATION_MESSAGE);
130                 else
131                     JOptionPane.showMessageDialog(null, "NonSmoking is
                        full.", "Error", JOptionPane.INFORMATION_MESSAGE);
132                 hidden.setState(true);
133             }//end of else block that checks the available room number
134         }//end of else block that checks the state of the hidden option
            button
135     }//end of the actionPerformed() method
136
137     //reset the text fields and choice component
138     void clearFields()
139     {
140         nameField.setText("");
141         phoneField.setText("");
142         numberOfGuests.select(0);
143         nameField.requestFocus();
144         hidden.setState(true);
145     }//end of clearFields() method
146
147 }//end of Reservation class
```