FIGURE 15.10

6.5 Checkbox

A checkbox allows user to check or uncheck an option. Users can check multiple checkboxes. Check boxes are user-interface components that have two states : on or off. A checkbox can be created with the help of one of the following constructors :

Constructor	Description
Checkbox( )	Creates a Checkbox without any label
Checkbox (String str)	Creates a Checkbox with the label str
Checkbox (String str, Boolean state)	Constructs a Checkbox with the label str and specified state. State may be true or false

Methods of Checkbox Class are :

Methods	Description
setLabel(String str)	To Set the Label or text to be displayed by the Checkbox
getLabel ( )	To get the label to be displayed by the Checkbox
setState(Boolean State)	To Set the State of Checkbox
getState( )	To get the State of Checkbox

Example

This program illustrates how to add check boxes in a frame.

```
import java.awt.*;
class demochkbox extends Frame
{
    public static void main(String args[ ])
    {
            Checkbox c1=new Checkbox("Hockey" , true);
            Checkbox c2=new Checkbox("Cricket");
            Checkbox c3=new Checkbox("Badminton");

            demochkbox f1 = new demochkbox();   //Create a frame
            f1.setTitle("Adding Check Boxes in a Frame"); //Set title of frame
            f1.setLayout(new FlowLayout());
            f1.add(c1);                          //adding a component to the frame
            f1.add(c2);
            f1.add(c3);
            f1.setSize(400, 100);                //set size of frame
            f1.setVisible(true);
    }
}
```

Output :

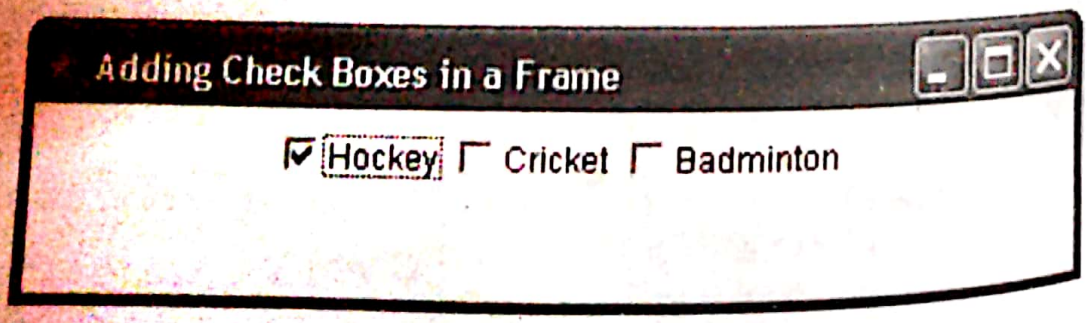

FIGURE 15.11

1.5.6.6 Radio Button (or Checkbox Group)

In Radio Buttons only one in a group can be selected at a time. **CheckboxGroup** class is used to create Radio Buttons. This class has only one constructor. The follwoing statement creates a checkbox group.

```
CheckboxGroup cbg = new CheckboxGroup( );
```

Now, we can create two checkboxes, belonging to CheckboxGroup, using the following statements.

```
Checkbox c1 = new Checkbox ("Yes", cbg, true);

Checkbox c2  = new Checkbox ("No", cbg, false);
```

Label — Checkboxgroup — State

Constructor	Description
CheckboxGroup( )	To Create a new CheckboxGroup
Checkbox(String, CheckboxGroup, boolean)	To Create a Checkbox with the specified label, set to the specified state, and in the specified check box group

Methods of CheckboxGroup Class are :

Methods	Description
getSelectedCheckbox( )	Gets the current choice
setSelectedCheckbox(Checkbox)	Sets the current choice to the specified Checkbox

Example

This program illustrates how to add radio buttons in a frame.

```
import java.awt.*;
class demoradiobutton extends Frame
{
    public static void main(String args[ ])
    {
        CheckboxGroup cg=new CheckboxGroup();

        Checkbox c1=new Checkbox("Cricket" , cg,true);
        Checkbox c2=new Checkbox("Badminton" , cg,false);
        Checkbox c3=new Checkbox("Hockey" , cg,false);

        demoradiobutton f1 = new demoradiobutton();   //Create a frame
        f1.setTitle("Adding Radio Buttons in a Frame"); //Set title of frame
        f1.setLayout(new FlowLayout());
        f1.add(c1);                                   //adding a component to the frame
        f1.add(c2);
        f1.add(c3);
        f1.setSize(400, 100);                         //set size of frame
        f1.setVisible(true);
    }
}
```

Output :

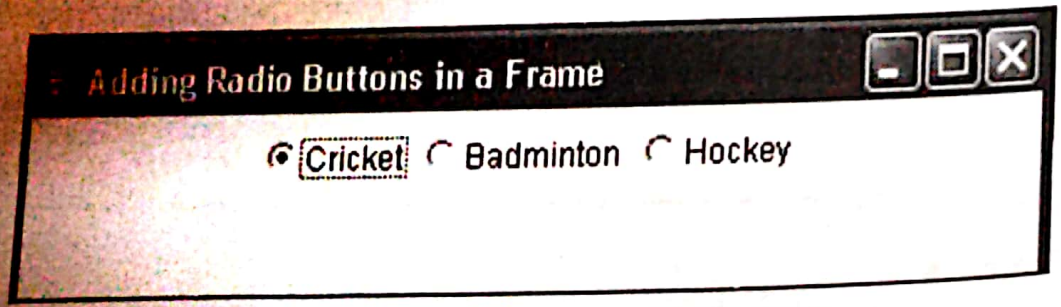

FIGURE 15.12

15.6.7 Choice

A Choice allows user to choose an item from a list of similar items. It is a drop down menu also called as a combo box (in swing) and only one item is visible before drop down.

Choice menu can have only one item selected at a time. To create a choice menu first creates an instances of the Choice Class and then add individual items to the object.

Constructor of Choice Class is :

Constructor	Description
Choice( )	Creates a new Choice instance

Methods of Choice Class are :

Methods	Description
add(String item)	To Add the item to the Choice
insert(String item, int position)	To Add the item at the specified position to the Choice
getSelectIndex( )	To Return the index position of item that is selected
getSelectedItem( )	To Return selected item as a string
select(pos)	To Select the item at a given position
select(String)	To Select the item with given string

Example

This program illustrates how to add Choice Menu in a frame.

```
import java.awt.*;
class demochoice extends Frame
{
    public static void main(String args[ ])
    {
        Choice c=new Choice( );
        c.add("Dal Makhni");
```

```
        c.add("Shahi Paneer");
        c.add("Mix Raita");
        c.add("Mix Veg.");

        demochoice f1 = new demochoice();  //Create a frame
        f1.setTitle("Adding Choice Menu in a Frame"); //Set title of frame
        f1.setLayout(new FlowLayout());
        f1.add(c);                         //adding a component to the frame
        f1.setSize(400, 200);              //set size of frame
        f1.setVisible(true);
    }
}
```

Output :

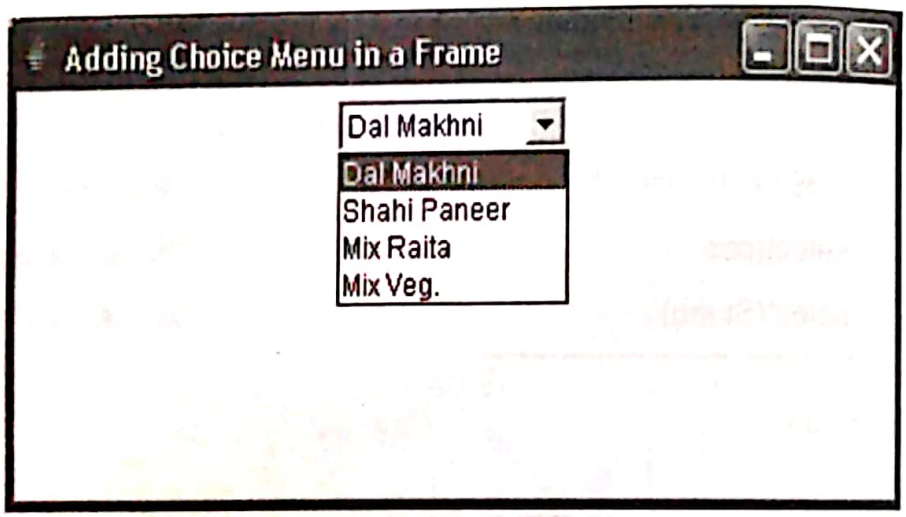

FIGURE 15.13

15.6.8 List

A List is similar to Choice. But in List we can choose more than one item at a time. However, there is also an option to restrict to a single selection. But a list looks different, all items are visible in the List.

Constructors of List class are :

Constructor	Description
List( )	Creates a new List
List(int n)	Creates a new List with the specified number of items visible
List (int n, Boolean state)	Created a new list with specified number of items visible. The Boolean state is allowed multiple selections, if it set to be true

Methods of List Class are :

Methods	Description
add(String item)	To Add the item to the List
add(String item, int pos)	To Add the item at the specified position to the List
clear( )	To delete all items from List
select(pos)	To Select the item at the specified index position
select(text)	To Select the first item in the list with the given text
getItem(index)	To Return text of the list item at the index position specified in the argument
getSelectedIndex( )	To Return an array of integers containing the index position of each selected item
getSelectedItem( )	To Return an array of strings containing the index position of each selected item

Example

This program illustrates how to add List in a frame.

```
import java.awt.*;
class demolist extends Frame
{
    public static void main(String args[ ])
    {
            List c=new List( 5,true);
            c.add("Dal Makhni");
            c.add("Shahi Paneer");
            c.add("Mix Raita");
            c.add("Veg. Pulaow");
            c.add("Rajma Chawal");

            demolist f1 = new demolist();   //Create a frame
            f1.setTitle("Adding List in a Frame"); //Set title of  frame
            f1.setLayout(new FlowLayout());
            f1.add(c);                      //adding a component to the frame
            f1.setSize(400, 200);           //set size of frame
            f1.setVisible(true);

    }
}
```

Output :

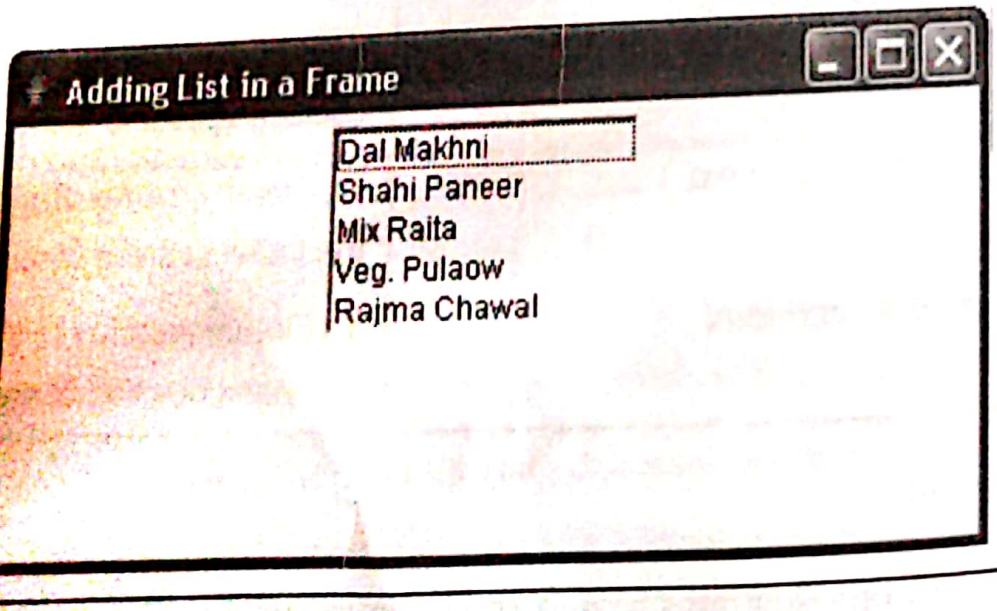

FIGURE 15.14