15.5.1 Frames

A Frame is a subclass of Window and has title bar, menu bar borders and resizing corners. It can be created using constructors, some of which are as follows :

- Frame() : Creates a Frame which is invisible
- Frame(String Title) : Creates an invisible Frame with the given title

Commonly used methods of Frame Object are :

Methods	Purpose
getTitle ( )	To get the title of the Frame Window
setTitle ( )	To set the title of the Frame Window
setVisible (boolean)	To make Frame visible or invisible
resize (Hori x, vert y)	To reset the dimension of the frame
setSize (size, size)	To set the dimension of the frame
show ( )	To display the frame on the screen
dispose ( )	To close frame window

Example

This program demonstrates how to create a Frame.

```
import java.awt.*;
class fsample extends Frame
{
    public fsample(String title)
    {
        super(title);
    }
    public static void main(String args[])
    {
        fsample Obj = new fsample("This is an example of a Frame");
        Obj.setSize(400, 400);
        Obj.setVisible(true);
    }
}
```

The user-defined class **fsample** is a subclass of the **Frame class**. This class has a constructor in which the method **super( )** is invoked. The purpose of super( ) here is to invoke the parent class constructor, which will then create an object of the child class in order to create the frame. However, the frame will be invisible and has no dimensions. Therefore, two methods, namely, setSize( ) and setVisible( ) are used in main method.

Output :

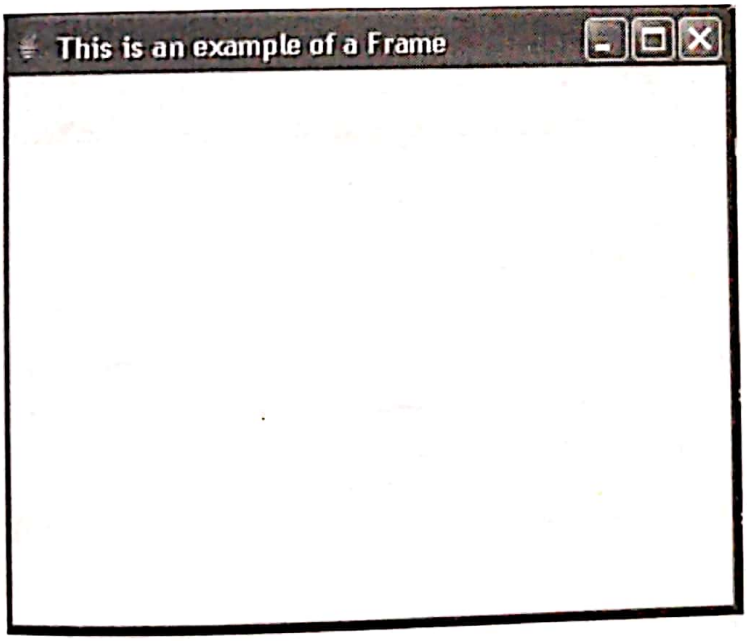

FIGURE 15.5

15.5.2 Panel

Panel is a container used to group a number of components together. Panel does not contain a title bar, menu bar or border. The simplest way to create a Panel is through its constructor **Panel( )**.

Example

This program demonstrates how to create a Panel.

```
import java.awt.*;
class psample extends Panel
{
    public static void main(String args[ ])
    {
        psample Obj = new psample( );
        Frame ObjFr = new Frame("This is an example of Panel");
        ObjFr. add(Obj);
        ObjFr.setSize(400, 400);
        ObjFr.setVisible(true);
    }
}
```

Since a Panel cannot be seen directly, it has to be added to a frame. So in the above example a new Frame is created and the newly created Panel instance is added to the Frame. Again Frame is invisible, therefore the two methods setSize( ) and setVisible( ) to set the size and display the frame respectively has been used.

Output :

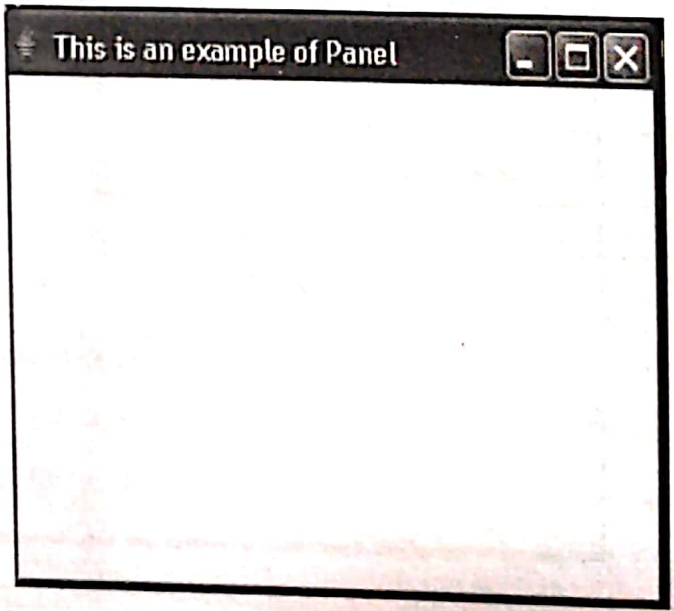

FIGURE 15.6

15.6 AWT Components

The **java.awt** package has a big collection of **AWT components** or **AWT Controls.**

Three steps are necessary to create and place a GUI component :

(i) Declare the component with an identifier (name)

(ii) Construct the component by invoking an appropriate constructor via the new operator;

(iii) Identify the container (such as Frame or Panel) designed to hold this component. The container can then add this component onto itself using **add( )** method. Note that, Every container has a **add (Component)** method.

Let us discuss some commonly used AWT components or AWT controls.

15.6.1 LABEL

Label class is used to create a text label on the screen. The display text is non editable. Label never performs any type of action. Label can be created by using one of the following constructors.

Constructor	Description
Label ( )	Create an empty Label
Label (string text)	creates a label with given text
Label (string str, int Alignment)	Creates a label displaying string str with the alignment which can be one of the right, left or center, as specified. Therefore alignment can be : • Label.LEFT (2) • Label.RIGHT (0) • Label.CENTER(1)

Methods of Label Class are :

Methods	Description
setText(String str)	Sets the text displayed by label
getText( )	Retrieve the text displayed by label
setAlignment(int Alignment)	Sets the alignment of the label
getAlignment( )	Retrieve the alignment of the label

Example

This program illustrates how to add label in a frame.

```
import java.awt.*;
class Labelsample extends Frame
{
    public static void main(String args[ ])
    {
        Label l1 = new Label("Label Demo" , Label.CENTER);
                                        //Declare and create a label
        Labelsample f1 = new Labelsample();  //Create a frame
        f1.setTitle("Adding a Label in a Frame");  //Set title of frame
        f1.add(l1);  //adding a component (Label) to the frame
        f1.setSize(400, 300);  //set size of frame
        f1.setVisible(true);
    }
}
```

Output :

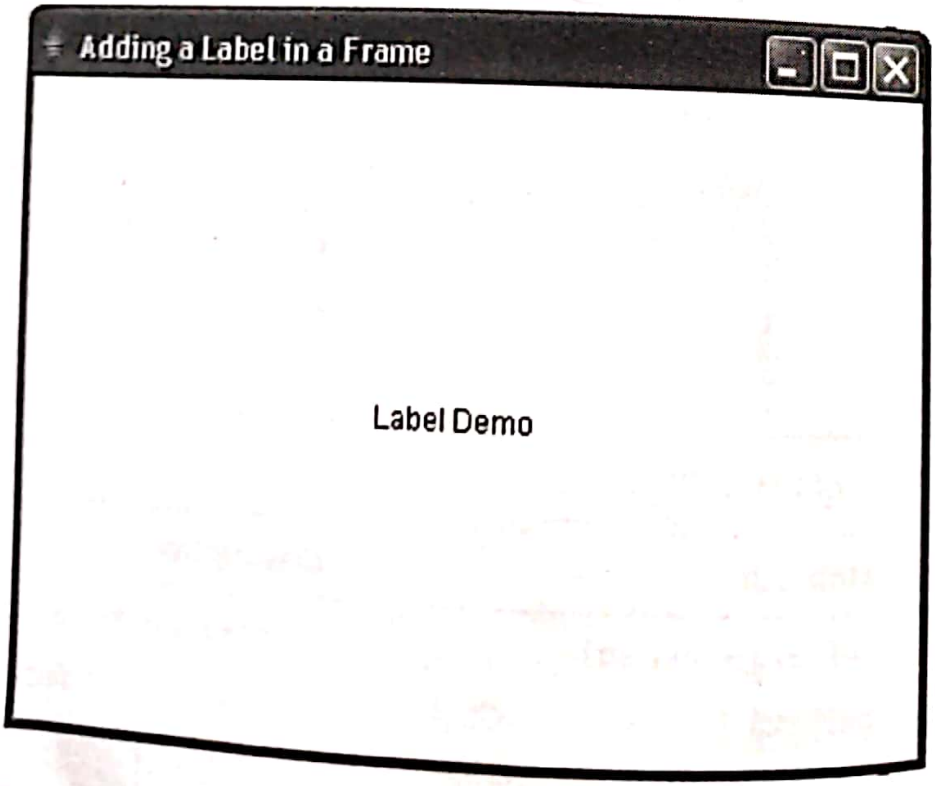

FIGURE 15.7