

Page no 3:

Ans 1: A "CheckboxGroup" differs from a "Checkbox" in that only one item in the "CheckboxGroup" can be selected at a time.

1. You can check multiple check boxes at once but this can never be done in the case of radio button. You can select only one radio button at once from a group of the radio button.
2. You can check or uncheck the check box but you can only check the radio button by clicking it once.

Ans 2: The code `setEnabled(false)`, disables this TextField. It is not selectable and the user can not copy data from it and the user cannot change the TextField's contents directly. While `setEnabled(false)` grays out the field completely, `setEditable(false)` just prevents it from being edited, but it will still look the same.

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

```
/*<applet code="DisableButtonExample" width=100 height=200>  
</applet>*/
```

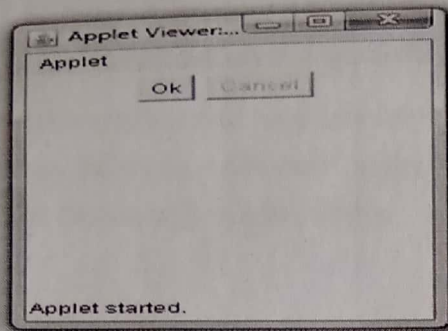
```
public class DisableButtonExample extends Applet{
```

```
    public void init(){  
        //create Buttons  
        Button Button1 = new Button("Ok");  
        Button Button2 = new Button("Cancel");  
        //add Buttons  
        add(Button1);  
        add(Button2);
```

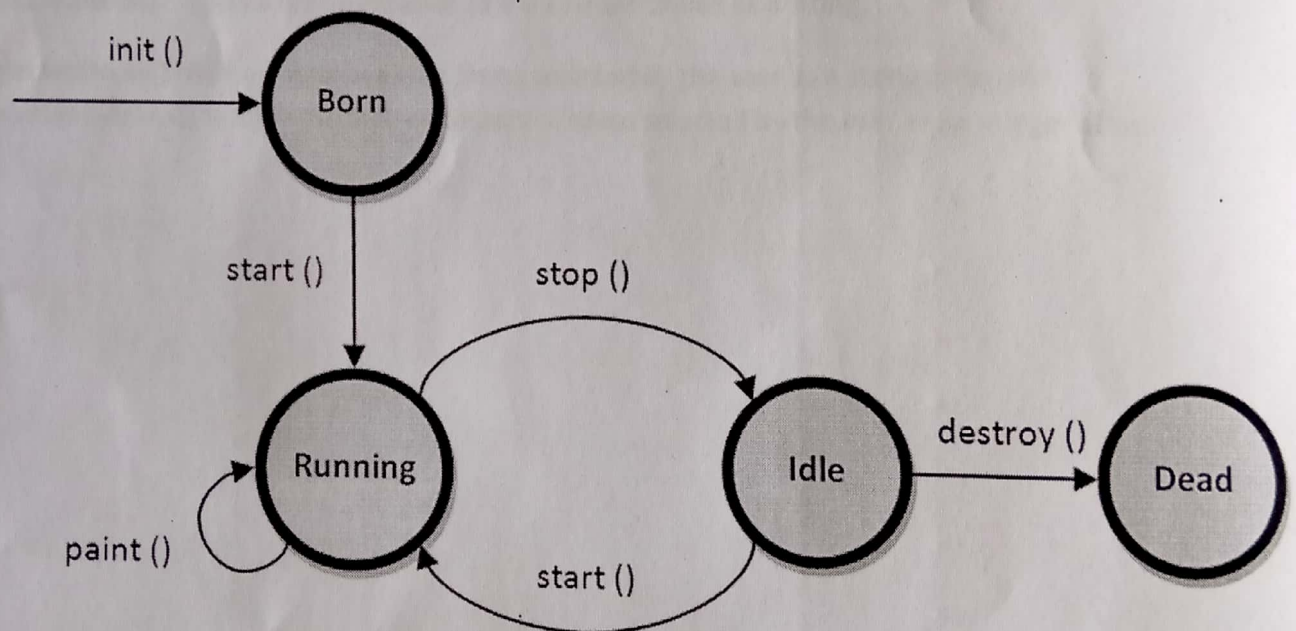
```
        Button2.setEnabled(false);
```

```
    }  
}
```

Example Output



Ans 3: Draw the life cycle of an Applet.



PAGE 9:

QNO 2: State difference between List and Choice in java

ANS 2: A choice is displayed in a compact form that requires you to pull it down to see the list of available choices and only one item may be selected from a choice.
A List may be displayed in such a way that several list items are visible and it support the selection of one or more list items.

QNO3: write the use of `getSelectedItem()` and `getSelectedIndex()` for List.

Ans:

`getSelectedIndex()` returns the index of the currently selected item.

`getSelectedItem()` gets a representation of the current choice as a string.

`getSelectedItems()` method returns all the items selected by the user as a string array and

`getSelectedIndexes()` returns the index numbers of items selected by the user as an integer array.

Qno 1: Give name of default Layout for Different Container.

Ans: Every container, by default, has a layout manager. For Panels, including Applets, the default layout manager belongs to the class `FlowLayout`. For Windows, the default layout manager is a `BorderLayout`. You can change the layout manager of a container using its `setLayout(LayoutManager)` method.

Qno2: List the names of Border Layout regions.

Ans: A border layout lays out a container, arranging and resizing its components to fit in five regions: North, South, East, West, and Center.

Constant	Description
<code>BorderLayout.NORTH</code>	It is a narrow side at the top, within the container
<code>BorderLayout.SOUTH</code>	It is a narrow side at the bottom, within the container
<code>BorderLayout.EAST</code>	It is a narrow side at the far right, within the container.
<code>BorderLayout.WEST</code>	It is a narrow side at the far right, within the container..
<code>BorderLayout.CENTER</code>	It is a large part at the center of container.

Qno3: Write the default horizontal and vertical gap in `FlowLayout`.

Ans: `FlowLayout()`: It will Construct a new `FlowLayout` with centered alignment. The horizontal and vertical gap will be 5 pixels.

Qno4: Write the use of Insets in `borderlayout`.

Ans 4: Insets Specifies the external padding of the component -- the minimum amount of space between the component and the edges of its display area. The value is specified as an Insets object. By default, each component has no external padding.

Every component has a set of insets. Insets specify the widths of the component's margins. Insets include:

- top inset
- left inset
- bottom inset
- right inset

Insets add space between the edges of the component and its cell. An Insets object has four fields, to specify the margins on the top, bottom, left, and right of the component. The relationship between insets and padding can be confusing. As shown in the following diagram, padding is added to the component itself, increasing its size. Insets are external to the component and represent the margin between the component and its cell.

