1. By referencing the javasoft web site, determine how many constructors are listed for the **JButton** class?

2. How does the third **JButton** constructor differ from the fourth?

3. How many methods are listed on the javasoft web site for the **JButton** class?

4. In examining the methods defined in the **JButton** class there are no methods to set the background and foreground (the label) colors of a **JButton**, yet you do have the ability to do that with method calls. How? (Hint — study the class hierarchy.)

5. What is the purpose of the second parameter in the **JTextArea** constructor used in the **Lab3Experiment**?

6. Which form of the **JCheckBox** constructor must be used in order to pre-set the state of the **JCheckBox** to true?

7. Experimenting with the Lab3ExperimentCode applet should enable you to answer the following questions. Is it possible to use the mouse and select 3 characters from:

a)   the **JLabel** widget? (i.e., Can you select by highlighting only 3 letters of a Label object?)

b)   the **JTextArea** widget?

c)   the **JTextField** widget?

d)   the **JList** widget?

For questions 8-13 give the method calls required to perform each of the tasks described. (You should modify the Lab3Experiment code to verify your answers.)

8. Give the exact Java statement that will invoke a method to change the text displayed on the cancel **JButton** to "QUIT".

9. Give the exact Java statement that will change the color of the **JButton** itself to red.

10. Give the exact Java statement that will change the color of the text displayed on the **JButton** to white.

11. Give the exact Java statement that allows multiple items to be selected from the **JList**object.

12. Give the exact Java statement that allows the new item "Basketball" to be placed as the second item in the **JComboBox** widget (i.e., between "Soccer" and "Football").

13. Give the exact Java statement that allows all items to be removed from the **JList**widget.

 - string.removeAllElements();

* String.clear();

The Layout Manager for the **JApplet** has been set to **FlowLayout** (we will study Layout managers in more detail in Chapter 5). If you resize the window currently being displayed by Eclipse you will note that the widgets reposition themselves. Lab3Experiment is not supplying absolute locations of where the widgets are to be placed, the layout manager is determining that based on the type of layout set for the container. You should be aware that this layout manager is called each time there is a change to the applet window.

14. What would be the effect of inserting the method call,

**myList.setLocation(10,10);**

in the **init()**method?  Does anything change?  Why or why not?