

Chapter 12 GUI Basics

1. java.awt.Component is the root of all Java GUI component classes. A container class such as JFrame is a subclass of Component. JComponent is the root of Swing GUI component classes.
2. The AWT components are heavy weight while the Swing components are lightweight.
3. You use the constructor of the JFrame class to create a frame. Use the setSize method to set the size of the frame. Use the getSize method to find the size of the frame. if the statement `frame.setSize(400, 300)` and `frame.setVisible(true)` were swapped in the `MyFrameWithComponents` class, you have to resize the frame to display the components in the frame. This is because the `setVisible(true)` statement causes the container to be repainted, but the `setSize(400, 300)` doesn't. When you resize the window, all the components are repainted in the frame.

4. You can add a button to a frame.

Answer: True

You can add a frame to a panel.

Answer: False

You can add a panel to a frame.

Answer: True

You can add any number of components to a panel, to a frame, or to an applet.

Answer: True

You can derive a class from JPanel, JFrame, or JApplet.

Answer: True

5. Line 7 created an instance of JFrame. It should be new Test() instead.
6. Component and JComponent are concrete classes, so you cannot create instances from them. The last line is wrong, because you cannot add an object to a container. Only an instance of Component can be added to a container.
7. The layout manager provides a platform-independent way to place components in a GUI interface. The default layout manager for the content pane of a frame is BorderLayout. To add a component to a frame, you have to add it to the content pane of the frame.
8. Using FlowLayout, the components are arranged in the container from left to right in the order in which they were added. If one row becomes filled, a new row is

started. To use a FlowLayout manager, you need to set the layout in a container to FlowLayout, such as with `setLayout(new FlowLayout())`. There is no limit on the number of components that can be added to a FlowLayout container.

9. The GridLayout manager arranges components in a grid (matrix) formation with the number of rows and columns defined by the constructor. The components are placed in the grid from left to right, starting from the first row, then the second, and so on, in the order in which they were added. To use a GridLayout manager, you need to set the layout in a container to GridLayout, such as with `setLayout(new GridLayout())`. The number of components you can add to a GridLayout container is unlimited. (Please try adding more components into the container, to see what happens when the number of components exceeds the grid size. Interestingly, the column is enlarged.)
10. The BorderLayout manager divides the window into five areas: East, South, West, North, and Center. Components are added to a BorderLayout using `add(String, Component)`, where String is "East", "South", "West", "North", or "Center". You can use one of the following two constructors to create a new BorderLayout:

```
public BorderLayout(int hGap, int vGap)
public BorderLayout
```

You can add only one component into a section. If you need to add multiple components in a section, group the components in a panel, and add the panel into the section.

11. Use the new `Color(int, int, int)` to create a color or use a standard color such as `Color.RED` to specify a color. It is wrong to create a color using `new Color(400, 200, 300)`, because the values 400 and 300 are out of range. `new Color(10, 0, 0)` is darker than `new Color(200, 0, 0)`, because the smaller value, the darker is the color.
12. Use `new Font(name, style, size)` to create a new font. To find out all the available colors
13. Use `new JPanel(LayoutManager)`.
14. The default layout manager for a JPanel is FlowLayout. Components are added directly to the JPanel.
15. No. There is no `setTitle()` method in JPanel. This method is defined for Frame; Panel is used to organize components. The panel itself is invisible.
16. You could add components into a button. It is legal, but a button should not be used as a container. See the last NOTE in Section 11.8.

17. use setBackground(Color), setForeground(Color), setFont(Font), and setToolTipText(String) to set background, foreground, font, and a tool tip text.

The tool tip text is not displayed, because Line 16 sets a tool tip text on jbtOK, but Line 17 adds a new button (different from jbtOK) to the content pane of the frame.

18. true
false

19. new ImageIcon("image/us.gif").

20. The effect is that only one button is added to the container. No syntax errors, nor runtime errors.

21. It displays only two buttons. The buttons cannot be shared.

22. Yes. (See the NOTE box in the Section 11.10).