Swing Interview Questions Answers in Java

<http://www.techinterviews.com/java-swing-interview-questions>

**1) Can a class be it’s own event handler? Explain how to implement this.**

**Answer**: Sure. an example could be a class that extends Jbutton and implements ActionListener. In the actionPerformed method, put the code to perform when the button is pressed.

**2) Why does JComponent have add() and remove() methods but Component does not?**

Answer: because JComponent is a subclass of Container, and can contain other components and jcomponents.

**3) How would you create a button with rounded edges?**

Answer: there’s 2 ways. The first thing is to know that a JButton’s edges are drawn by a Border. so you can override the Button’s paintComponent(Graphics) method and draw a circle or rounded rectangle (whatever), and turn off the border. Or you can create a custom border that draws a circle or rounded rectangle around any component and set the button’s border to it.

**4) If I wanted to use a SolarisUI for just a JTabbedPane, and the Metal UI for everything else, how would I do that?**

Answer: in the UIDefaults table, override the entry for tabbed pane and put in the SolarisUI delegate. (I don’t know it offhand, but I think it’s "com.sun.ui.motiflookandfeel.MotifTabbedPaneUI" - anything similar is a good answer.)

**5) What is the difference between the ‘Font’ and ‘FontMetrics’ class?**

Answer: The Font Class is used to render ‘glyphs’ - the characters you see on the screen. FontMetrics encapsulates information about a specific font on a specific Graphics object. (width of the characters, ascent, descent)

**6) What class is at the top of the AWT event hierarchy?**

Answer: java.awt.AWTEvent. if they say java.awt.Event, they haven’t dealt with swing or AWT in a while.

**7) Explain how to render an HTML page using only Swing.**

Answer: Use a JEditorPane or JTextPane and set it with an HTMLEditorKit, then load the text into the pane.

**8) How would you detect a keypress in a JComboBox?**

Answer: This is a trick. most people would say ‘add a KeyListener to the JComboBox’ - but the right answer is ‘add a KeyListener to the JComboBox’s editor component.’

**9) Why should the implementation of any Swing callback (like a listener) execute quickly?**

A: Because callbacks are invoked by the event dispatch thread which will be blocked processing other events for as long as your method takes to execute.

**10) In what context should the value of Swing components be updated directly?**

A: Swing components should be updated directly only in the context of callback methods invoked from the event dispatch thread. Any other context is not thread safe?

**11) Why would you use SwingUtilities.invokeAndWait or SwingUtilities.invokeLater?**

A: I want to update a Swing component but I’m not in a callback. If I want the update to happen immediately (perhaps for a progress bar component) then I’d use invokeAndWait. If I don’t care when the update occurs, I’d use invokeLater.

**12) If your UI seems to freeze periodically, what might be a likely reason?**

Ans : A callback implementation like ActionListener.actionPerformed or MouseListener.mouseClicked is taking a long time to execute thereby blocking the event dispatch thread from processing other UI events.

**13) Which Swing methods are thread-safe?**

A: The only thread-safe methods are repaint(), revalidate(), and invalidate()

**14) Why won’t the JVM terminate when I close all the application windows?**

A: The AWT event dispatcher thread is not a daemon thread. You must explicitly call System.exit to terminate the JVM.

**15): What is EDT thread in Swing?**

A: This is one of the basic questions in Swing and AWT interviews. EDT stands for Event dispatcher thread. EDT is one of the most important thing to learn about Swing, Since Swing is single-threaded all GUI drawing and event handling is done in EDT thread and that's why its recommended not to perform any time consuming task e.g.[connecting to database](http://javarevisited.blogspot.sg/2012/04/java-program-to-connect-oracle-database.html) or opening network connection in Event Dispatcher thread or EDT thread. If you do that, it may lead to frozen or hung GUI. This question leads to several other questions in Java, e.g. If you can not open database connection in EDT thread than how will you open db connection with button click etc. well this can be done by spawning a new thread from button click and opening db connection there.

**16): What is difference between invokeAndWait and invokeLater in Java?**

A: This is one of my favourite Swing question in Java Interviews. Knowledge of invokeAndWait and invokeLater is must for a good Swing developer because Swing is not [thread-safe](http://javarevisited.blogspot.sg/2012/12/how-to-create-thread-safe-singleton-in-java-example.html) and at same time you can not perform time consuming task in EDT thread, as discussed in first Swing interview question. InvokeAndWait and InvokeLater method allows to enqueue task for EDT thread to perform, InvokeAndWait is a [blocking method in Java](http://javarevisited.blogspot.sg/2012/02/what-is-blocking-methods-in-java-and.html) and does it synchronously and invokeLater does it asynchronously. Since GUI can only be updated in Event dispatcher thread, if you need to show progress of any task, you can use these two methods. See my post [Difference between invokeAndWait and invokeLater in Java](http://javarevisited.blogspot.sg/2011/09/invokeandwait-invokelater-swing-example.html) for more detailed answer of this Swing question.

**17): What is difference between Swing and AWT in Java**

A : One of the most frequently asked AWT and Swing Interview question. One answer of this question is that, Swing is a considered as light weight and AWT is considered as heavy weight. Another difference between AWT and Swing is that, Swing offers uniform look and feel across platform while look and feel of AWT GUI application are platform dependent because AWT mostly use native components e.g. a AWT windows will look different in DOS and Windows operating system.

**18) : What is difference between paint and repaint in Java Swing?**

A : This can be a [tough Java question](http://java67.blogspot.sg/2012/09/top-10-tough-core-java-interview-questions-answers.html) if you are not familiar with Swing API. Well, this is similar to [start() and run() method of thread](http://javarevisited.blogspot.sg/2012/03/difference-between-start-and-run-method.html) class. As calling start() method will eventually calls run() method of [Runnable](http://java67.blogspot.sg/2012/08/what-is-thread-and-runnable-in-java.html) interface, Calling repaint() will call paint() method in Swing. Since painting can only be done in EDT thread, repaint() just put a paint request in EDT Queue, later EDT thread may combine several repaint request to one and can perform repainting. repaint() is a not a blocking method in Java and returns quickly.

**19) : What is difference between BorderLayout and GridLayout ?**

A : This is one of interesting Swing interview question. BorderLayout and GridLayout are two widely used LayoutManagerfrom Swing API, former arranges components in predefined position e.g. NORTH, SOUTH, EAST and WEST while later arranges components one after other until there is space and wraps them afterwards.

**20) : How to change a button from enable to disable after click ?**

A : When button is clicked an action event is fired which can be captured by implementing ActionListener interface and actionPerformed(ActionEvent ae) method. You can then call button.setEnable(false) to disable this button.

**21) : Why Swing is called light weight ?**

A : Most of Swing component are inherited form JComponent and doesn't required a native peer and that's why they are referred as light weight component. light weight component implement look and feel in Java rather than using look and feel of native component and that's why Swing's look and feel remains same across platform.

**22) : Is Swing Thread safe in Java ?**

A : This is one of the [tricky question](http://java67.blogspot.sg/2012/09/top-10-tricky-java-interview-questions-answers.html) in Java Swing and AWT. No, Swing is not thread-safe in Java. Swing components are not thread-safe they can not be modified from multiple threads. All swing components are designed to be updated by using Event dispatcher thread or EDT thread.

**23) : Which method of Swing are thread-safe?**

A : This AWT and Swing question is asked as follow-up of previous Swing interview question. Only couple of methods like **repaint() and revalidate()** are thread-safe in Swing, i.e. they can be called from multiple threads without additional [synchronization in Java](http://javarevisited.blogspot.sg/2011/04/synchronization-in-java-synchronized.html).

**24) : What is difference between Container and Component ?**

A : Main difference between Container and Component is that former can hold other components e.g. JFrame which is used as container to hold other components e.g. JButton. This is rather a simple Swing question and mostly asked in telephonic or upto 2 years experienced programmers.