

## Kontrollfragen A

1. Die meisten GUI-Komponenten in Swing leiten sich von einer gemeinsamen Oberklasse ab. Von welcher?

*JComponent*

Welches sind die Ausnahmen?

*Top-Level-Container*

2. Welches sind die Top-Level-Container von Swing und woher werden sie abgeleitet?

*JFrame, JApplet, JWindow, JDialog*

*Sie sind von den AWT-Pendants abgeleitet, z.B. JFrame von Frame, JWindow von Window, etc.*

3. Für Swing Top-Level-Container gilt die Regel, dass Komponenten nicht direkt hinzugefügt werden. Wer nimmt die GUI-Komponenten auf?

*JRootPane*

## Kontrollfragen B

1. JComponent steht am Anfang der Swing Klassenhierarchie. Welche Arten von Operationen stellt die Klasse zur Verfügung? Nennen Sie mindestens drei.

*setBorder(), setToolTipText(), registerKeyboardAction()*

*Alle Operationen von Component (ist ja davon abgeleitet)*

*Weitere „Optionen“ wie z.B. Position, Grössenangaben, Zustände (aktiv, sichtbar, usw).*

2. In Swing können viele Komponenten zusätzlich zu einem Beschriftungs-Text auch Grafiken aufnehmen. Nennen Sie Beispiele.

*JLabel, JButton, JList, JComboBox*

## Kontrollfragen C

1. Erklären Sie den Unterschied zwischen der Implementation eines Listeners

- als eigenständige Klasse
- als innere Klasse
- als anonymer innere Klasse

*Die eigenständige / innere Klasse sind fast gleich, eine „innere Klasse“ wird innerhalb einer bestehender Klasse implementiert.*

*Die innere Klasse muss nicht immer gegen aussen sichtbar sein.*

*Die anonyme innere Klasse hat keinen Klassennamen, sie kann nicht angesprochen werden. Die volle Kontrolle ist bei der umliegenden Klasse der anonymen inneren Klasse.*

*Die anonyme innere Klasse braucht etwas weniger Speicher und ist minim effizienter.*

2. Was gewinnt man durch den Einsatz von Adapter-Klassen?

*Es müssen nicht alle Methoden implementiert werden, einzig die Methoden welche benötigt werden.*