

---

## Übungsblatt 6

---

**Abgabe: 17.06.2024, 10:00 Uhr** (im Digicampus via VIPS: .java-Dateien für Code, .uxf für UML, .pdf für alles andere)

- Dieses Übungsblatt muss im Team abgegeben werden (Einzelabgaben sind nicht erlaubt!).
- Die **Zeitangaben** geben zur Orientierung an, wie viel Zeit für eine Aufgabe später in der Klausur vorgesehen wäre; gehen Sie davon aus, dass Sie zum jetzigen Zeitpunkt wesentlich länger brauchen und die angegebene Zeit erst nach ausreichender Übung erreichen.

\* leichte Aufgabe / \*\* mittelschwere Aufgabe / \*\*\* schwere Aufgabe

### Aufgabe 21 \*\* (*Fenster ohne Ereignisbehandlung, 30 Minuten*)

In dieser Aufgabe sollen Sie die grundlegenden Elemente von Fenstern in Java näher kennen lernen. Fügen Sie jeder Fensterklasse auch eine `main`-Methode hinzu, mit der eine Instanz des Fensters erzeugt wird.

a) (\*, *Eingabe mit Textfeld, 8 Minuten*)

Implementieren Sie ein Hauptanwendungsfenster nach folgenden Vorgaben:

- Der Titel des Fensters soll `My First Java Frame` sein.
- Beim Klicken des Fensterkreuzes soll das Fenster geschlossen und das Programm beendet werden.
- Das Fenster soll  $400 \times 100$  Pixel groß sein.
- Im Zentrum des Fensters soll ein Eingabefeld sein, das zu Beginn den Inhalt `"Text..."` hat.
- Links und rechts von diesem Eingabefeld sollen jeweils Buttons mit dem Text `Left` bzw. `Right` sein. Unterhalb des Eingabefelds soll ein unveränderbarer Text `South` stehen, der horizontal zentriert ist.

b) (\*\*, *Farben, 8 Minuten*)

Implementieren Sie ein Hauptanwendungsfenster nach folgenden Vorgaben:

- Der Titel des Fensters soll `Grey Color Scheme` sein.
- Beim Klicken des Fensterkreuzes soll das Fenster geschlossen und das Programm beendet werden.
- Das Fenster soll in einem  $16 \times 16$  Gitter Panel der Größe  $20 \times 20$  Pixel enthalten.
- Die Panel sollen einen gleichmäßigen Farbverlauf von Schwarz zu Weiß darstellen.
- Die Panel sollen in jeder Zeile von links nach rechts heller werden.
- Das erste Panel in der ersten Zeile soll den Grauwert 0 (schwarz) haben, das erste Panel in der zweiten Zeile den Grauwert 16 usw., bis das letzte Panel in der letzten Zeile den Grauwert 255 (weiß) hat.

*Hinweis: Um die Größe von Komponenten in einem Container mit `LayoutManager` anzupassen, muss die Methode `setPreferredSize` statt `setSize` verwendet werden.*

c) (\*\*, Fenster mit Menü, 14 Minuten)

Implementieren Sie ein Hauptanwendungsfenster nach folgenden Vorgaben:

- Der Titel des Fensters soll **Java Menus** sein.
- Beim Klicken des Fensterkreuzes soll das Fenster geschlossen und das Programm beendet werden.
- Das Fenster soll zwei Menüs haben: **File** und **Edit**.
- Das Menü **File** soll folgende Einträge haben:
  - zwei Einträge **New** und **Save**
  - eine Trennlinie
  - vier Radio-Buttons, von denen immer nur einer ausgewählt werden kann, die den Namen **Option 1** bis **Option 4** haben
- Das Menü **Edit** soll einen Eintrag **Undo** haben.
- Das Fenster soll ein mehrzeiliges Textfeld mit 100 Spalten und 4 Zeilen beinhalten.
- Das Fenster soll genau so groß sein, dass alle enthaltenen Komponenten dargestellt werden.

*Hinweis: Recherchieren Sie im API-Eintrag der Klasse für Radio Buttons in einem Menü eine Möglichkeit, wie mehrere Buttons zusammengefasst werden können, sodass immer nur einer ausgewählt sein kann.*

## Aufgabe 22 \* (Fenster mit Ereignisbehandlung (**ActionEvents**), 32 Minuten)

Fügen Sie jeder Fensterklasse auch eine **main**-Methode hinzu, mit der eine Instanz des Fensters erzeugt wird.

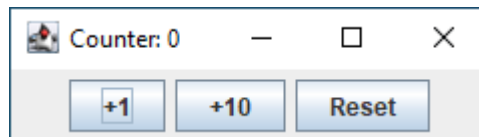
a) (\*, 9 Minuten)

Implementieren Sie ein Hauptanwendungsfenster nach folgenden Vorgaben:

- Der Titel des Fensters soll **Color Switcher** sein.
- Beim Klicken des Fensterkreuzes soll das Fenster geschlossen und das Programm beendet werden.
- Das Fenster soll ein Panel enthalten.
- Das Panel enthält zwei Buttons mit den Texten **Green** bzw. **Red**.
- Wird ein Button geklickt, soll sich die Hintergrundfarbe des Panels auf die entsprechende Farbe ändern.
- Verwenden Sie Lambda-Ausdrücke für die Ereignisabnehmer.

b) (\*, 13 Minuten)

Implementieren Sie ein Hauptanwendungsfenster das wie folgt aussieht:



Das Fenster soll folgendes Verhalten implementieren

- Der Titel des Fensters soll zu Beginn **Counter: 0** sein.
- Beim Klicken des Fensterkreuzes soll das Fenster geschlossen und das Programm beendet werden.
- Wenn der Button **+1** bzw. **+10** geklickt wird, soll die Zähler im Titel des Fensters um 1 bzw. 10 erhöht werden.
- Wenn der Button **Reset** geklickt wird, soll der Zähler auf 0 zurückgesetzt werden.
- Implementieren Sie Ihre Fensterklasse als Ereignisabnehmer.

---

c) (\*, 10 Minuten)

Implementieren Sie ein Hauptanwendungsfenster nach folgenden Vorgaben:

- Der Titel des Fensters soll zu Beginn **Please select a title** sein.
- Beim Klicken des Fensterkreuzes soll das Fenster geschlossen und das Programm beendet werden.
- Das Fenster soll  $400 \times 100$  Pixel groß sein.
- Das Fenster soll eine editierbare Dropdown-Box enthalten.
- Der Inhalt der Dropdown-Box soll dem Fenster beim Erstellen als Konstruktorparameter in Form eines **String**-Arrays übergeben werden können.
- Wird ein Eintrag ausgewählt oder eingegeben und mit Enter bestätigt, soll der Titel des Fensters entsprechend angepasst werden.
- Implementieren Sie eine eigene Abhörer-Klasse, die sowohl das Fenster als auch die Dropdown-Box als Parameter im Konstruktoraufbau erhält und das beschriebene Verhalten umsetzt.

**Aufgabe 23** \*\* (*Fenster mit Ereignisbehandlung (WindowEvents, KeyEvents, JOptionPane, 15 Minuten)*)

Implementieren Sie ein Hauptanwendungsfenster nach folgenden Vorgaben. Fügen Sie der Klasse auch eine **main**-Methode hinzu, mit der eine Instanz des Fensters erzeugt wird.

a) (\*\*, 10 Minuten)

- Es soll den Titel **Window Event Log** haben, 500 Pixel breit und 500 Pixel hoch sein.
- Im Zentrum soll es ein deaktiviertes mehrzeiliges Textfeld mit 50 Spalten, 5 Zeilen und mit Scrollleiste anzeigen.
- Löst der Benutzer irgendein Fensterereignis aus, so sollen Zeitpunkt und Beschreibung des Ereignisses als Zeichenkette an den Inhalt des Textfelds in einer neuen Zeile angehängt werden.

*Hinweis:* Implementieren Sie einen Ereignisabhörer Ihrer Wahl.

b) (\*\*, 5 Minuten)

- Zusätzlich soll sich beim Klicken auf das Fensterkreuz ein einfacher modaler Informationsdialog mit der Nachricht **Hit the 'X' key to close the application!** öffnen.

*Hinweis:* Benutzen Sie die Klasse **JOptionPane** für den Informationsdialog und implementieren Sie einen Ereignisabhörer Ihrer Wahl.

*Hinweis:* Implementieren Sie einen Ereignisabhörer Ihrer Wahl.

---

## Aufgabe 24 \*\* (Zeichnen in Java, 20 Minuten)

a) (\*\*, 17 Minuten)

Implementieren Sie eine Zeichenfläche der Größe  $500 \times 500$  Pixel, in der ein Rechteck gezeichnet werden kann:

- Das Rechteck soll mit der Maus mit gedrückter Maustaste „aufgezogen“ werden können.
- Der Punkt, an dem zuerst die Maustaste gedrückt wurde, ist eine Ecke des Rechtecks.
- Der Punkt, an dem die Maus mit gedrückter Maustaste gerade ist, soll die gegenüberliegende Ecke des Rechtecks sein.
- Das Rechteck soll laufend aktualisiert werden, während die Maus mit gedrückter Maustaste bewegt wird.
- Zusätzlich sollen die Länge und Breite des Rechtecks etwa in der Mitte der oberen bzw. rechten Kante als Zahl angezeigt werden.
- Das Rechteck soll solange bestehen bleiben, bis ein neues Rechteck begonnen wird, dann soll das alte Rechteck verschwinden.
- Achten Sie auch darauf, dass man auch von rechts nach links und von unten nach oben ein Rechteck aufziehen kann!

*Hinweis: Um die Größe von Komponenten in einem Container mit `LayoutManager` anzupassen, muss die Methode `setPreferredSize` statt `setSize` verwendet werden.*

b) (\*\*, 3 Minuten)

Implementieren Sie ein Hauptanwendungsfenster, das die Zeichenfläche der vorherigen Teilaufgabe beinhaltet. Beim Klicken des Fensterkreuzes soll das Fenster geschlossen und das Programm beendet werden. Fügen Sie der Klasse auch eine `main`-Methode hinzu, mit der eine Instanz des Fensters erzeugt werden kann.