

— Übungsblatt 1 (P) —

Aufgabe 1 (P)

(Frames)

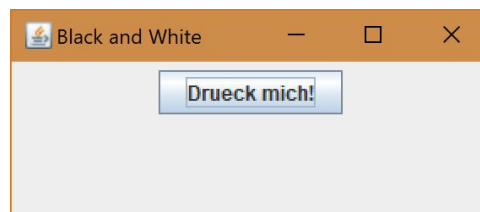
Schreiben Sie ein kleines Programm mit einer grafischen Oberfläche, die lediglich einen Button enthält.

Durch Druck auf diesen Button soll die Hintergrundfarbe des Frames auf die Farbe Schwarz gesetzt werden.

Durch Mausklick in den Frame-Hintergrund soll die Hintergrundfarbe des Frames auf die Farbe Weiß gesetzt werden.

Gehen Sie wie folgt vor:

- Schreiben Sie eine eigene Frame-Klasse, mit der Fenster der Form



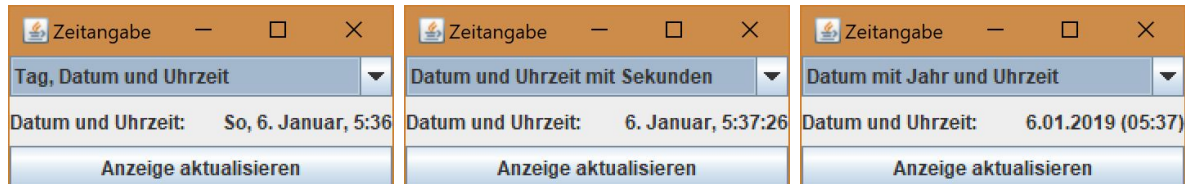
erzeugt werden können.

- Im Konstruktor soll ein JButton-Objekt erzeugt und dem Frame-Container hinzugefügt werden. Außerdem soll
 - ein Objekt der Klasse ButtonBearbeiter erzeugt und beim JButton-Objekt als ActionListener registriert werden und
 - ein Objekt der Klasse MausBearbeiter erzeugt und beim Frame-Container als MouseListener registriert werden.
- Realisieren Sie die Ereignis-Verarbeitung für das JButton-Objekt durch eine innere Klasse ButtonBearbeiter, deren Instanzmethode actionPerformed lediglich den Hintergrund des Frames auf die Farbe Schwarz setzt.
- Realisieren Sie die Mausklick-Ereignis-Verarbeitung durch eine innere Klasse MausBearbeiter, deren Instanzmethode mousePressed lediglich den Hintergrund des Frames auf die Farbe Weiß setzt. Die anderen Methoden, die das Interface MouseListener vorschreibt, können einen leeren Rumpf erhalten.
- Schreiben Sie eine main-Methode, in der ein entsprechender Frame der Größe 250 × 150 Punkte erzeugt wird.

Aufgabe 2 (P)

(Frames)

Erstellen Sie ein Java-Programm mit grafischer Oberfläche, das jeweils eine aktuelle Datum- und Uhrzeit-Angabe in drei unterschiedlichen Formaten anzeigen kann und die nachfolgend dargestellte Oberfläche und Funktionalität haben soll.



Der Frame soll also beim Start die im linken Bild dargestellte Form haben und der bzw. die Benutzer(in) soll durch Auswahl in der Klapptafel eine andere Darstellungsform für Datum und Uhrzeit wählen können. Mit dem Button soll eine Aktualisierung der Anzeige durch Neubestimmung des aktuellen Zeitpunkts durchgeführt werden können.

Verwenden Sie in Ihrem Programm private Instanzvariablen `beschriftung` und `datumsAnzeige` vom Typ `JLabel`, `formatAuswahl` vom Typ `JComboBox`, `datum` vom Typ `Date`, `aktualisierung` vom Typ `JButton` sowie Klassenvariablen `kurz`, `mittel` und `lang` vom Typ `SimpleDateFormat` für die formatierte Darstellung des Datums und außerdem eine öffentliche Klassenvariable `gewaehlt` vom Typ `SimpleDateFormat`. Sorgen Sie dafür, dass `kurz`, `mittel` und `lang` final sind und die entsprechenden `SimpleDateFormat`-Objekte bereits bei der Deklaration erzeugt werden. Die Variable `gewaehlt` soll zunächst den Wert von `lang` erhalten.

Implementieren Sie den Konstruktor so, dass

- die benötigten Swing-Komponenten erzeugt und in der richtigen Reihenfolge in den Frame im Border-Layout eingefügt werden,
- das Datum erzeugt und für die Beschriftung des `datumsAnzeige`-Labels verwendet wird,
- bei der Auswahl-Klapptafel ein Event-Listener registriert wird, der als Objekt einer **inneren** Klasse `AnzeigeListener` erzeugt wurde,
- beim Button ein Event-Listener registriert wird, der als Objekt einer **separaten** Klasse `KnopfHorcher` erzeugt wurde.

Die innere Klasse `AnzeigeListener` muss das Interface `ItemListener` implementieren und die Methode `itemStateChanged` so überschreiben, dass bei Änderung der Auswahl der entsprechende Auswahl-Index des `JComboBox`-Objekts bestimmt und (abhängig von dessen Wert) das entsprechende Darstellungsformat für das aktuelle Datum gewählt und zur Beschriftung des `datumsAnzeige`-Labels verwendet wird. Dazu passend soll außerdem die Variable `gewaehlt` auf einen der Werte `kurz`, `mittel` oder `lang` gesetzt werden.

Die separate Klasse `KnopfHorcher` muss das Interface `ActionListener` implementieren und die Methode `actionPerformed` so überschreiben, dass bei Betätigen des Buttons der aktuelle Zeitpunkt bestimmt und damit die Beschriftung des Anzeige-Labels aktualisiert wird. Dabei soll das durch die Variable `gewaehlt` der Frame-Klasse festgelegte Format verwendet werden. Beachten Sie, dass Sie auch einen Konstruktor schreiben müssen, dem die Referenz auf das Anzeige-Label des Frames übergeben wird.

Aufgabe 3 (P)

(Frames)

Schreiben Sie ein Programm in Form eines Frames, dessen grafische Oberfläche sich als einfache, handgeschaltete Ampel der Form



präsentiert und es ermöglicht, per Mausklick die Ampel umzuschalten.

Statten Sie dazu den Container Ihres Frames in einem geeigneten Layout mit drei Auswahl-Kästchen, von denen jeweils nur eines aktiviert sein darf, und mit drei Buttons (als Lampen) aus. Zu Beginn soll „rot“ eingestellt sein. Arbeiten Sie mit einem einzigen Listener-Objekt, das Sie bei allen Komponenten registrieren.

Realisieren Sie die Ereignisbehandlung in einer inneren Klasse. Sorgen Sie dafür, dass bei jedem Aktivieren eines Auswahl-Kästchens die entsprechende Lampe der Ampel eingeschaltet wird und die anderen ausgeschaltet werden. Dies soll auch bei einem Mausklick auf die Lampen selbst funktionieren. Dabei müssen Sie beachten, dass die Anzeige konsistent gehalten werden soll, d. h. auch das zugehörige Auswahl-Kästchen muss von der Listener-Methode aktiviert werden!

Aufgabe 4 (P)

(Frames)

Schreiben Sie eine grafische Java-Applikation, die es ermöglicht, zwei Zahlen einzugeben, die danach als Operanden für eine der vier Grundrechenarten verwendet werden. Optisch soll sich Ihr „Rechner“ folgendermaßen präsentieren:



Verwenden Sie zwei Textfelder für die Eingabe der beiden Operanden und ein Label für die Ausgabe des Ergebniswerts sowie zwei Labels für die Anzeige der gerade durchgeführten Operation und für das Gleichheitszeichen (=). Für die auszuführenden Operationen benötigen Sie fünf jeweils mit einer der vier Grundrechenarten bzw. dem Text "Loesche alles" beschriftete Buttons, bei denen jeweils ein eigenes Event-Listener-Objekt vom Typ `OperatorListener` registriert ist.

Die zugehörige Ereignis-Behandlung soll für alle `JButton`-Objekte über ein und dieselbe innere Klasse `OperatorListener` realisiert werden, die das Interface `ActionListener` implementiert. Welche Operation bei Knopfdruck ausgeführt werden muss, soll dabei beim Erzeugen des Listener-Objekts als `char`-Wert (+, -, *, / oder 1 für Löschen) übergeben werden. Dieser Wert ist daher in der `actionPerformed`-Methode zu berücksichtigen.

Wird eine der fünf Schaltflächen betätigt, muss also zunächst festgestellt werden, ob es sich um den Lösch-Knopf handelt. Wenn ja, sind alle Eingabefelder, das Ergebnisfeld und das Anzeigefeld für die durchgeführte Operation zu löschen.

In allen anderen Fällen muss die entsprechende arithmetische Operation ausgeführt werden. Dazu müssen zunächst die beiden Textfelder ausgelesen und in `double`-Werte umgewandelt werden. Danach kann die Berechnung durchgeführt, das Ergebnis in dem dafür vorgesehenen Label angezeigt und das Symbol der durchgeführten Operation zwischen den beiden Operanden angezeigt werden. Denken Sie auch daran, im Falle unzulässiger Eingaben die entsprechenden Exceptions abzufangen und über das Ergebnisfeld eine entsprechende Fehlermeldung anzuzeigen.

Aufgabe 5 (P)

(Frames)

Schreiben Sie eine Java-Klasse, die als einfaches Euro-Umrechnungsprogramm der Form

Euro	Währung	Ergebnis
100	Deutsche Mark	195.583
100	Oesterreichische Schilling	1376.0300000000002
51.12918811962185	Deutsche Mark	100

dienen kann. Dabei sollen die zweite und vierte Zeile jeweils ein Textfeld zur Eingabe *und* Ausgabe darstellen. In der dritten Zeile soll mit Hilfe eines `JComboBox`-Objekts (Klapptafel) die Umrechnungswährung eingestellt werden können. Nach Eingabe eines Euro-Betrags in der zweiten Zeile soll beim Druck auf die Eingabetaste in die eingestellte Währung umgerechnet und der Betrag in der untersten Zeile angezeigt werden. Wählt man in der Klapptafel eine andere Währung aus, so soll ebenfalls sofort in die neu eingestellte Währung umgerechnet und der Betrag in der untersten Zeile angezeigt werden. Nach Eingabe eines Betrags in der untersten Zeile soll beim Druck auf die Eingabetaste gemäß der eingestellten Währung in Euro umgerechnet und der Euro-Betrag in der zweiten Zeile angezeigt werden.

Für Ihre Implementierung steht Ihnen die Klasse `EuroConverter` zum Download zur Verfügung, die neben den *Währungskennungen* DEM, ATS, ..., GDR auch die drei von Ihnen benötigten Klassen-Methoden `getBezeichnung`, `convertFromEuro` und `convertToEuro` bereitstellt.

Statten Sie die grafische Oberfläche Ihres Programms mit einem geeigneten Layout und den benötigten Swing-Komponenten aus. Verwenden Sie eine geeignete Schleife über die Währungs-Kennungen und die Methode `getBezeichnung`, um der `JComboBox`-Komponente alle benötigten Einträge hinzuzufügen. Verknüpfen Sie die beiden Textfelder und das `JComboBox`-Objekt mit je einem `EventListener`.

Die zugehörige Ereignis-Behandlung soll wie folgt arbeiten:

- Wird nach Eingabe eines Betrags im *oberen* Textfeld die *Eingabetaste betätigt* oder der *Klapptafel-Eintrag geändert*, so soll der `double`-Wert aus dem *oberen* Textfeld ausgelesen, mit Hilfe der `EuroConverter`-Klasse in die eingestellte Währung umgerechnet und der neue Betrag im *unteren* Textfeld angezeigt werden.
- Wird nach Eingabe eines Betrags im *unteren* Textfeld die *Eingabetaste betätigt*, so soll der `double`-Wert aus dem *unteren* Textfeld ausgelesen, mit Hilfe der `EuroConverter`-Klasse in Euro umgerechnet und der neue Betrag im *oberen* Textfeld angezeigt werden.