

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 25 Februar 2009
---	---

Tag 1 - Übung 2 - Online Doku

Aufgabenstellung:

Findet mit Hilfe der Online Dokumentation Alternativen zur Verwendung der Label Komponente und fügt sie der Anwendung hinzu.

Zusatzaufgabe(n):

- Fügt der Anwendung fünf weitere Komponenten eurer Wahl hinzu.
- Sucht in der Dokumentation nach einer Möglichkeit das TitleWindow per Mouse zu bewegen (Tip: nur ein spezielles Attribut muss gesetzt werden)

Hinweise:

- Verwendet nicht die Design View.
- Nehmt das Paket mx.controls unter die Lupe.

Code Snippets:

Keine.

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 25 Februar 2009
---	---

Tag 1 - Übung 3 - MXML & AS3

Aufgabenstellung:

Es liegen zwei Layoutskizzen vor. Anhand dieser sollen die dargestellten Komponenten umgesetzt werden.

Das Kalenderfenster sollte in MXML oder mit Hilfe des Design Views erstellt werden. Das Konfigurationsfenster soll mit AS3 in einem Skriptblock erstellt werden.

Zusatzaufgabe(n):

- Am Schriftgrößen-Slider sollen in 8er-Schritten Skalenbeschriftungen angezeigt werden.
- Tage, die in der Vergangenheit liegen, dürfen im DateChooser nicht auswählbar sein.

Hinweise:

Keine.

Tag 1 Übung 3

Code Snippets:

// Einen Script Block erstellt ihr durch:

```
<mx:Script>
    <![CDATA[

        import paket.KlassenName;

        private function nameDerMethode():void{
        }

    ]]>
</mx:Script>
```

// Um eine Methode direkt mit dem Start der Anwendung aufzurufen könnt ihr das „creation Complete“ Event benutzen:

```
<mx:Application creationComplete="init()">
```

```
<mx:Script>
    <![CDATA[
        // Und die Funktion die dann ausgeführt wird.
        private function init():void
        {
            Alert.show("application created");
        }
    ]]>
</mx:Script>
```

// Comboboxeinträge können mit folgender folgender Syntax erstellt werden

```
comboBox.dataProvider = ['Eintrag 1', 'Eintrag 2', '...'];
```

// Um ein GUI Objekt per AS3 einem Container hinzuzufügen geht so vor:

```
var guiObject:Label = new Label();
container.addChild(guiObject);
```

Daniel Bertram
Kerstin Götze

ASS Flex Workshop Wintersemester 2009
für die FH Köln, Campus Gummersbach

25 Februar 2009

Tag 1 - Übung 3 - Layoutskizze 1

Kalender

◀ Januar 09. ▶

1	2	3	4	5	6	7
8	9	10
...
...

Neuer Eintrag

I....

☐ Erinnerung aktivieren

5 min vorher

5 min vorher
10 min vorher
1 h vorher
1 Tag vorher
1 Woche vorher

Eintrag speichern

Daniel Bertram
Kerstin Götze


ASS Flex Workshop Wintersemester 2009
für die FH Köln, Campus Gummersbach

25 Februar 2009

Tag 1 - Übung 3 - Layoutskizze 2

Kalenderkonfiguration


Farbschema



Akustische Benachrichtigung

☒ Sound abspielen bei Erinnerung

Schriftgröße



Werte:

- (8 - 64)
- 16 vorausgewählt

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 25 Februar 2009
---	---

Tag 1 - Übung 4 - Onlinekatalog

Aufgabenstellung:

Es soll ein Fenster mit drei Tabs mit den Titeln „Onlinekatalog“, „Newsletter“ und „Werbung“ geben.

- Im ersten Tab befinden sich untereinander drei Checkboxes mit den Beschriftungen „MS DOS“, „Floppy Disk“ und „VGA Monitor“, ein Button mit dem Label „Bestellen“ und ein Texteingabefeld für Anmerkungen.
- Im zweiten Tab sollen sich nebeneinander Label mit der Beschriftung „E-Mail Adresse:“, ein Texteingabefeld und ein Button mit dem Label „Newsletter abonnieren“ befinden.
- Im dritten Tab soll das Bild unseres Hochleistungsrechners (pc.jpg) gezeigt werden. Monitor, Maus und Tastatur sollen mit je einem Label beschriftet werden.

Zusatzaufgabe(n):

Keine.

Hinweise:

- Kopiert euch die Assets für Tag 1 Übung 3 vom Server und fügt

Tag 1 Übung 4

sie in den assets-Ordner eures Flex Projektes ein.

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 26 Februar 2009
---	---

Tag 2 - Übung 1 – Events

Aufgabenstellung:

Fügt der Kalenderanwendung folgendes Verhalten hinzu:

Button „Eintrag speichern“:

- Klick: Alert erscheint mit Text „Eintrag gespeichert“
- Mouse über Button: Schrift des Buttons wird kursiv
- Mouse verlässt Button: Schrift des Buttons wieder normal
- ColorPicker: Bei Farbauswahl Hintergrundfarbe des Kalenderfensters auf die gewählte Farbe setzen
- Slider: Ändert die Schriftgröße des Kalenderfensters

Zusatzaufgabe(n):

- Über den „x“-Button in der Titelleiste des Konfigurationsfensters, soll dieses geschlossen werden.

Hinweise:

- Schaut euch in der Online Doku an, welche Events die Komponenten werfen können.

Tag 2 Übung 1

Code Snippets:

```
// Event Listener werden in MXML so hinzugefügt:  
<mx:Button label="click me" click="onButtonClick(event)"/>
```

```
// Event Listener werden in AS3 so hinzugefügt:  
myButton.addEventListener(MouseEvent.CLICK, onButtonClick);
```

```
// So wird ein Style per AS3 gesetzt  
guiObj.setStyle("fontSize", 12);
```

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 26 Februar 2009
---	---

Tag 2 - Übung 2 – Klassen, Getter und Setter

Aufgabenstellung:

Erstellt ein neues Projekt „FlexMediaPlayer“.

Erzeugt eine MXML Komponente „MediaPlayer“, die von TitleWindow ableitet. Diese soll Getter und Setter für die Eigenschaft `_url` vom Typ String erhalten. Fügt außerdem der Klasse ein Label hinzu, welches diese URL anzeigt. Fügt außerdem eine Instanz von VideoDisplay hinzu.

Erzeugt ein AS3 Klasse „Playlist“, die ebenfalls von TitleWindow ableitet. Setzt im Konstruktor Breite, Höhe und Titel des Fensters.

Fügt je eine Instanz beider Klassen der Anwendung hinzu und setzt `_url` für die MediaPlayer-Instanz auf „../assets/video/demo.flv“.

Zusatzaufgabe(n):

Keine.

Tag 2 Übung 2

Hinweise:

- Getter und Setter einer Eigenschaft, dürfen nicht genau den gleichen Namen haben, wie diese Eigenschaft, sonst entsteht bei Aufruf des Getters oder Setters eine Endlosschleife
- Setzt den Wert für die Variable `_url` der MediaPlayer Instanz erst, wenn die Anwendung vollständig erzeugt wurde
- Getter und Setter werden in einem MXML Klasse in den Skript-Block geschrieben

Code Snippets:

Getter und Setter

```
private var _volume:Number;
```

```
// Setter
```

```
public function set volume(value:Number):void{  
    _volume = value;  
}
```

```
// Getter
```

```
public function get volume():Number{  
    return _volume;  
}
```

// Eigene GUI Klassen in der Anwendung verwenden (MXML Syntax)

```
<mx:Application  
    xmlns:mx="http://www.adobe.com/2006/mxml"  
    xmlns:local="*"  
>  
    <local:MyClass />  
</Application>
```

// Eigene Klassen in der Anwendung verwenden (AS3 Syntax)

```
private var createGui():void{  
    var myObject:MyClass = new MyClass();  
    this.addChild(myObject);  
}
```

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 26 Februar 2009
---	---

Tag 2 - Übung 3 – Audio & Video

Aufgabenstellung:

Erweitert die Klasse MediaPlayer so, das der Setter für `_url` die Source für das VideoDisplay setzt.

Fügt außerdem folgende Komponenten mit entsprechender Funktionalität hinzu:

- Einen 'Start' Button
- Einen 'Stop' Button
- Einen 'Pause' Button

Fügt einen Slider hinzu, der die Lautstärke des Videos regelt.

Zusatzaufgabe(n):

- Fügt einen weiteren Slider hinzu, mit dem ihr in dem Video navigieren könnt.

Hinweise:

- Kopiert euch die Assets für Tag 2 Übung 3 vom Server und fügt sie in den assets-Ordner eures Flex Projektes ein.

Code Snippets:

Tag 2 Übung 3

Keine.

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 26 Februar 2009
---	---

Tag 2 - Übung 4 – Bild & Font

Aufgabenstellung:

Fügt das Flex Logo (flexlogo.png) als eingebettetes Bild in die rechte obere Ecke der Anwendung ein.

Bettet außerdem die euch zur Verfügung gestellte Schriftart in die Anwendung ein. Setzt sie als Standardschriftart für die Anwendung.

Zusatzaufgabe(n):

Keine

Hinweise:

- Kopiert euch die Assets für Tag 2 Übung 4 vom Server und fügt sie in den assets-Ordner eures Flex Projektes ein.
- Für die verschiedenen Schriftstile (bold, italic, ...) muss jeweils die entsprechende Font-Datei eingebettet werden, die diesen Stil enthält.

Tag 2 Übung 4

Code Snippets:

Eine Schriftart mit mx:Style einbinden. (Diese muß noch gesetzt werden.)

```
<mx:Style>
    @font-face {
        src: url("../assets/font/verdana.TTF");
        fontFamily: myFontFamily;
        advancedAntiAliasing: true;
        fontWeight: bold;
    }
</mx:Style>
```

Bilder einbetten mit MXML

```
<mx:Image source="@Embed('Bildpfad')" />
```

Bilder einbetten mit AS3

```
[Embed(source="icon.png")] private var icon:Class;
```

```
private function useImage():void{
    var imgObj:BitmapAsset = new icon() as BitmapAsset;
    var img:Image = new Image();
    img.source = imgObj;
}
```

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 26 Februar 2009
---	---

Tag 2 - Übung 5 – Styles & Skins

Aufgabenstellung:

Passt die Anwendung mittels externem CSS und Filtern folgendermaßen an:

- Hintergrundfarbe der Applikation: #ADCCA7
- Buttons allgemein: Farbe #7B9177, Alpha 1
 - Pause Button: Farbe #FFD91C
 - Stop Button: Farbe #FF361C
- Slider: slider.png als Skin für den Anfasser
- Mediaplayerbuttons: GlowFilter in ihrer jeweiligen Farbe

Zusatzaufgabe(n):

- Erstellt einen Programmatischen Skin für den Pause Button.
 - upSkin: Farbe #FFD91C
 - overSkin: Farbe #FFA14F
 - 2px breiter Rand in der Farbe #FF6100

Tag 2 Übung 5

Hinweise:

- Kopiert euch die Assets für Tag 2 Übung 5 vom Server und fügt sie in den assets-Ordner eures Flex Projektes ein.
- Filter erstellt ihr mit `<mx:filters>`
- Beachtet bitte das die Flex Syntax nicht zu 100% mit der CSS Syntax übereinstimmt !

Code Snippets:

CSS Syntax

```
Button {  
    fontSize: 15;  
    fillColors: #FF0000, #FF0000;  
}  
.greenButtonStyle {  
    fontSize: 12;  
    fillColors: #00FF00, #00FF00;  
}
```

Einbinden eines CSS mit

```
<mx:Style source="../../../assets/Styles.css"/>
```

Objekten eine CSS Style zuweisen:

```
guiObj.styleName="greenButtonStyle";
```

Filter verwenden

```
<mx:Label id="label1" text="Filter Test" fontSize="20">  
    <mx:filters>  
        <mx:DropShadowFilter blurX="5" blurY="5"  
distance="20"/>  
    </mx:filters>  
</mx:Label>
```

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 27 Februar 2009
---	---

Tag 3 - Übung 1 – Dataprovider

Aufgabenstellung:

Erweitert die AS3 Klasse Playlist, indem ihr

- ein List Objekt namens `_liste` einfügt
- zwei globale Variablen `_listData` (ArrayCollection) und `_selectedURL` (String) hinzufügt und jeweils Getter und Setter für sie schreibt
- der Setter für `_listData` setzt den Data Provider von `_liste`

Beim Auftreten des `creationComplete` Events der Anwendung, soll diese `_listData` der Playlist mit Demodaten füllen.

Bei Klick auf einen Playlisteintrag soll `_selectedURL` in Playlist entsprechend gesetzt werden. Bindet die `_url` des `MediaPlayers` an `_selectedURL` der Playlist.

Zusatzaufgabe(n):

- Fügt mit Hilfe des Data Providers den ersten zwei Playlist-Einträgen ein Icon hinzu. Die Icons findet ihr in `assets/img`

Tag 3 Übung 1

Hinweise:

- Kopiert euch die Assets für Tag 3 Übung 1 vom Server und fügt sie in den assets-Ordner eures Flex Projektes ein. Füllt den Data Provider der Playlist mit den Pfaden der Videos.
- Um ein DataBinding durchzuführen müssen für die zu bindende Variable ein Getter und Setter existieren, die mit dem [Bindable]-Tag gekennzeichnet sind.
- Nur eingebettete Bilder können als Listenicons verwendet werden

Code Snippets:

Binden der Daten eines TextInputs an ein Text Objekt:

```
<mx:TextInput id="myTI"/>                                <!--source-->
<mx:Text id="myText" text="{ myTI.text}"/>              <!--target-->
```

Der folgende Block bindet das Label der List an den Text der TextArea.

```
<mx>List id="list" width="400" itemClick="onItemClick()"/>
<mx:TextArea id="output"/>

<mx:Script> <![CDATA[
    import mx.events.ListEvent;
    import mx.collections.ArrayCollection;

    private var dp:ArrayCollection;

    private function initDataProvider():void{
        dp = new ArrayCollection();
        dp.addItem( {label:"Google", data:"www.google.de"} );
        dp.addItem( {label:"Yahoo", data:"www.yahoo.de"} );
        list.dataProvider = dp;
    }

    private function onItemClick():void{
        output.text = list.selectedItem.data as String;
    }
}]></mx:Script>
```

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 27 Februar 2009
---	---

Tag 3 - Übung 2 – Effekte

Aufgabenstellung:

Verseht die Komponenten der Anwendung mit folgenden Effekten:

- Klick auf Pause Button: ein Fade Effekt lässt VideoDisplay innerhalb von 0,5 Sekunden halb transparent werden.
- Klick auf Start Button: Fade Effekt wieder rückgängig

Zusatzaufgabe(n):

MediaPlayer-Komponente und Playlist-Komponente:

- Cursor über TitleWindow: Fenster erhält Glow Effekt und der Rand des Fensters wird komplett undurchsichtig
- Cursor verlässt TitleWindow: vorherige Effekte rückgängig

Hinweise:

- Auch Effekte lösen Events aus.
- Effekte sind in Flex als eigene Klasse geschrieben, die ihr unter ***mx.effects*** finden könnt.

Tag 3 Übung 2

Code Snippets:

Erstellen eines Effektes und auslösen durch einen Button:

```
<mx:WipeLeft id="myWL" duration="1000" target="{ myPanel}"/>  
<mx:Button label="Play effect" click="{ myWL.play()}" />
```

Ein Objekt per AS3 automatisch einen Effekt auslösen lassen:

```
object.setStyle(„effectName“, effectInstance);
```

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 27 Februar 2009
---	---

Tag 3 - Übung 3 – XML Parsing

Aufgabenstellung:

Lest die Einträge der Playlist aus der XML-Datei
www.rockabyte.de/kgoetze/flex/playlist.xml aus.

Zusatzaufgabe(n):

Lest die Einträge der Playlist aus der XML-Datei
www.rockabyte.de/kgoetze/flex/playlistWithNamespace.xml aus.

Hinweise:

Keine.

Code Snippets:

Ein XML Dokument und der Parser dazu:

```
<news>
  <item id="0">
    <title>Krass: Rollstuhlrennen im Altersheim</title>
    <description>
      Geschwindigkeit mal ganz langsam
    </description>
  </item>
</news>
```

```
if (xmlData.hasOwnProperty("item"))
{
  for each (var item:XML in xmlData.item as XMList)
  {
    output.htmlText += "\nNews item number " + item.@id + "\n";

    if (item.hasOwnProperty("title"))
      output.htmlText += "<b>" + item.title + "</b>\n";

    if (item.hasOwnProperty("description"))
      output.htmlText += item.description + "\n";
  }
}
```

Daniel Bertram Kerstin Götze	ASS Flex Workshop Wintersemester 2009 für die FH Köln, Campus Gummersbach 27 Februar 2009
---	---

Tag 3 - Übung 4 – Web Services

Aufgabenstellung:

Fügt der Anwendung eine neue Klasse „ShoutBox“ hinzu, die von TitleWindow ableitet.

Fügt einen Button und ein Textfeld hinzu. Bei Klick auf den Button soll folgender Webservice aufgerufen werden:

- URL: <http://www.boyzoid.com/comp/randomQuote.cfc?wsdl>
- Operation: getQuote

Das Ergebnis des Dienstes soll im Textfeld angezeigt werden.

Fügt eine Instanz von ShoutBox der Anwendung hinzu.

Zusatzaufgabe(n):

- Fügt ein weiteres TitleWindow hinzu, in welchem ihr den folgenden Webservice abfragt:
<http://webservices.daehosting.com/services/TemperatureConversions.wso?WSDL>
Verwendet den Web Service um Temperaturen von Celsius in Fahrenheit (und umgekehrt) umzurechnen. Der umzurechnende Wert soll über ein Texteingabefeld eingegeben werden können. Entnehmt die benötigten Operationen der WSDL.

Hinweise:

Tag 3 Übung 4

Keine

Code Snippets:

Web Service abrufen

```
<mx:WebService
    id="WS"
    wsdl="http://ws.invesbot.com/stockquotes.asmx?WSDL"
    fault="Alert.show(event.fault.faultString, 'Error'"
>
    <mx:operation name="GetQuote" resultFormat="object">
        <mx:request>
            <symbol> {stockSymbol.text} </symbol>
        </mx:request>
    </mx:operation>
</mx:WebService>

<mx:Panel title="WebService Example" >

    <mx:Label text="Enter a stock symbol to obtain a quote."/>
        <mx:TextInput id="stockSymbol" text="ADBE"/>
        <mx:Button label="Get Quote"
click="WS.GetQuote.send()"/>

    <mx:Text
        htmlText="Company:{WS.GetQuote.lastResult.GetQuoteResult.StockQuote.Company}"
    />
    <mx:Text
        htmlText="Current price: ${WS.GetQuote.lastResult.GetQuoteResult.StockQuote.Price}"
    />

</mx:Panel>
```