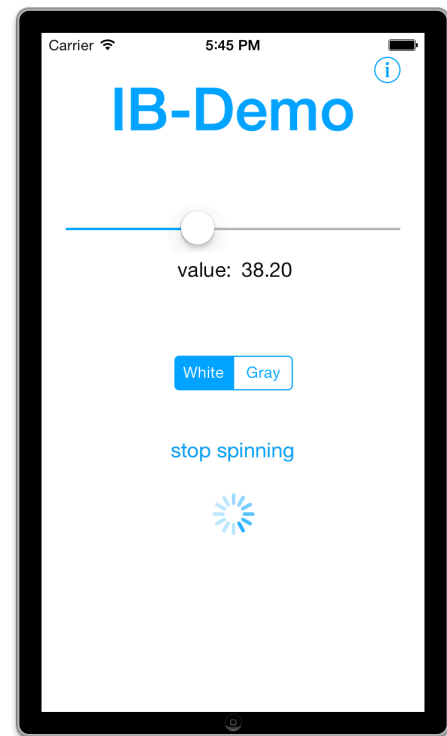


## Übung 2: Interface Builder

Das Ziel dieser Übung ist es, den Interface Builder und die entsprechenden Mechanismen (IBOutlet, IBAction) kennen zu lernen. Dazu wird eine Demo-iPhone-Applikation programmiert. Das Programm soll exemplarisch ein paar UI-Komponenten (Slider, Textfeld, Knöpfe, Spinner) verwenden und dann wie der Screenshot rechts aussehen.



### 1. Neues Projekt „IB-Demo“

Erstellen sie in Xcode ein neues Projekt für's iPhone. Wählen sie „Single View Application“, Name „IB-Demo“. Xcode generiert damit automatisch ein Projekt mit diversen vorgegebenen Dateien.

### 2. Titel platzieren im Interface Builder

Unter anderem wurde von Xcode eine Datei „Main.storyboard“ generiert. Klicken sie auf diese, dann startet sich der in Xcode integrierte Interface Builder. Öffnen sie im Interface Builder rechts die Ansichten für „Attributes Inspector“ und „Object Library“ (siehe Menüpunkt „View“ - „Utilities“). Die „Object Library“ zeigt alle zur Verfügung stehenden Objektklassen an. Ziehen sie daraus ein „Label“ auf den View Controller im Storyboard. Schreiben sie einen geeigneten Titel hin, z.B. „IB-Demo“ und ändern sie Schriftgrösse, -farbe, und -ausrichtung wie gewünscht. Dies können sie mit Hilfe des „Attribute Inspector“ tun.

### 3. Slider und Text-Label

Ergänzen sie die Bildschirmansicht mit einem Slider und zwei weiteren Labels, ein Label ist für die Beschriftung, das andere soll dann den aktuellen Wert des Sliders anzeigen (ähnlich wie im Screenshot oben rechts). Geben sie dem Slider mit Hilfe vom „Attribute Inspector“ im Interface Builder einen Wertebereich von 0 bis 100.

### 4. IBOutlet und IBAction für Slider

Nun muss der Slider noch mit dem Code verbunden werden. Öffnen sie dazu am einfachsten den „Assistant Editor“ (Menu „View“ – „Assistant Editor“ – „Show Assistant Editor“), da dieser zwei Editoren (hier einen für .storyboard- und einen für die .h-, resp. .swift-Datei) nebeneinander erlaubt. Verbinden sie den Slider und das Sliderlabel mit der Maus mit der entsprechenden IBAction, resp. dem entsprechenden IBOutlet von der ihrer ViewController-Klasse. Dazu deklarieren sie in ihrer ViewController-Klasse entweder eine passende IBAction `sliderValueChanged` und ein passendes IBOutlet `sliderLabel` oder sie lassen diese beim Verbinden (mit gedrückter ctrl-Taste) mit der Maus von Xcode generieren. Die IBAction soll als einziges Argument `sender` vom Typ `UISlider` haben und ausgelöst werden, wenn sich der Wert vom Slider ändert.

### 5. Action programmieren

Was noch fehlt, ist die Implementierung der IBAction-Methode `sliderValueChanged` für den Slider. Wenn sich der Wert vom Slider (durch Benutzerinteraktion) ändert, soll also der neue Wert im Slider-Label angezeigt

werden. Lesen sie dazu in der Doku von UISlider nach, wie sie am einfachsten zu dessen Wert kommen. Und setzen sie diesen Wert dann (als String) im Property text vom sliderLabel.

## 6. Auswahlknopf zum Hintergrundfarbe ändern

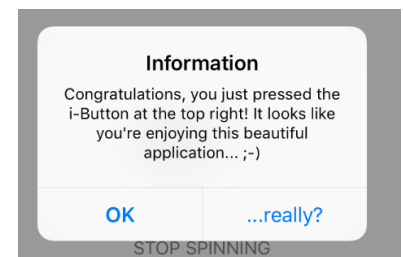
Erweitern sie das Programm um einen Auswahlknopf (Segmented Control) um die Hintergrundfarbe zu ändern. Fügen sie dazu ihrer ViewController-Klasse eine neue IBAktion `colorButtonValueChanged` hinzu, welche beim Event `Value Changed` der `UISegmentedControl` ausgeführt wird und entsprechend die Hintergrundfarbe auf grau resp. weiss ändert.

## 7. Spinner auf Knopf-Druck an- und abstellen

Erweitern sie das Programm um einen Spinner (Activity Indicator View) und einen Knopf (Button), um diesen zu steuern. Fügen sie dazu ihrer ViewController-Klasse ein weiteres IBOutlet `spinner` und eine IBAktion `spinButtonPressed` hinzu. Verbinden sie diese mit den entsprechenden UI-Elementen und programmieren sie die notwendige Action. Benutzen sie dazu die Methoden `isAnimating`, `startAnimating`, `stopAnimating` der Klasse `UIActivityIndicatorView`. Verwenden sie im weiteren die Methode `setTitle:forState` der Klasse `UIButton`, um dem Knopf den passenden Text „start spinning“ oder „STOP SPINNING“ zu setzen.

## 8. Info-Knopf mit Hinweis

Fügen sie zu ihrer ViewController-Klasse eine neue IBAktion `infoButtonPressed` hinzu, welche verbunden ist mit einem i-Knopf rechts oben (UIButton, Typ `Info Light`). Der Knopf soll auf Klick (Event `Touch Up Inside`) eine Info-Meldung mit zwei Möglichkeiten „OK“ und „...really“ anzeigen, siehe Screenshot rechts. Verbinden sie die IBAktion mit dem entsprechend Knopf und verwenden sie in der Action-Methode eine Instanz `UIAlertController` vom Style `Alert` (siehe Xcode-Doku, bzw. Modul-Unterlagen, sie müssen dazu Ihrer `UIAlertController`-Instanz u.a. eine `UIAlertAction` setzen).



## 9. 90-er Hinweis beim Slider

Erweitern sie ihren Slider, so dass eine Warnung (wie oben: `UIAlertController`, Style `Alert`) angezeigt wird, wenn der Slider einen Wert von grösser als 90 angewählt wird. Solange der Slider zwischen 90 und 100 bewegt wird, sollen keine weiteren Hinweise angezeigt werden, sondern einfach dann wieder, wenn dieser Bereich verlassen und erneut angewählt wird.