

12. Preloader



Vorbereitung

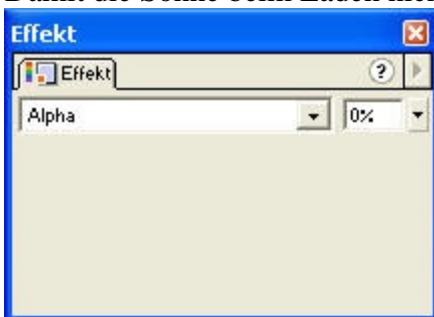
- Wir bauen uns einen Preloader, der eine Animation von 0% bis 100% vergrößert, ja nach Ladezustand des Hauptfilmes. Außerdem wird der Prozentwert noch als Zeichenfolge unten, mittig angezeigt.
- Dazu nehmen wir einen neuen Film und fügen 2 Szenen ein. Die Erste nennt sich Preloader, die Zweite Intro. Diese Szenen können im Bedienfeld Szene erstellt werden. Dazu rufen wir **Fenster / Bedienfelder / Szene** auf.
- In der Preloader Szene erstellen wir 3 Ebenen:
 - Aktionen - für das ActionScript
 - Text - für die Prozentanzeige als Zeichenfolge
 - Symbol - für das sich vergrößernde Objekt
- In der Ebene Text erstellen wir ein dynamisches Feld mit der Anfangsanzeige 000 und einem anschließendem statischen Text "% geladen". Das dynamische Feld bekommt den Variablen Namen geladen zugewiesen.



- In der Ebene Symbol können wir die Sonne aus der vorangegangenen Übung einfügen. Dies geht über **Datei / Als Bibliothek öffnen (Strg + Umschalt + O)** öffnen. Diese Instanz nennen wir Sonne:



- Die bisherige Animation macht noch nicht viel [flash/11_02a.swf](#)  [flash/11_02a fla](#) 
- Damit die Sonne beim Laden nicht gleich gezeigt wird, setzen wir den Alphawert für die Instanz auf 0



- Wir wechseln in die Szene Intro und fügen auf der Bühne über **Datei / Importieren (Strg + R)** ein möglichst großes Bild ein. Ich habe dafür einfach ein Hardcopy vom Bildschirm gemacht und als BMP-Datei gespeichert. Diese Datei hat eine Größe von 3,4 MB. Dies wird benötigt, um eine entsprechende Ladezeit zu simulieren.

getBytesTotal und getBytesLoaded

- Nun wechseln wir zum 1. Schlüsselbild in der Szene Preloader in der Ebene Aktionen und fügen dort folgende ActionScript Befehle ein:
 - `FilmGroesse = this.getBytesTotal();`
Der Variable FilmGroesse wird die Größe in Bytes des aktuellen Filmes zugewiesen
 - `FilmProzent = FilmGroesse / 100;`
Speichert in der Variablen FilmProzent die Bytes für je 1%
- Jetzt fügen wir ein leeres Schlüsselbild im 2. Bild ein. Im 2. Schlüsselbild wird dann folgendes ActionScript eingetragen:
 - `ZurZeitGeladen = this.getBytesLoaded()`
Überprüft wieviel Bytes bereits geladen sind
 - `_root.Prozent = Math.round(ZuerZeitGeladen / FilmProzent);`
Errechnet, wieviel Prozent das vom gesamten Film sind
 - `setProperty (_root.Sonne, _xscale, _root.Prozent);`
`setProperty (_root.Sonne, _yscale, _root.Prozent);`
Setzt die Größe der Ladeanimation auf die entsprechende Prozentzahl
 - `setProperty (_root.Sonne, _alpha, 100);`
Setzt den Alpha Wert des Objektes, welches sich vergrößern soll, wieder auf 100, damit es sichtbar wird.

If then else

- Wir fügen ein weiteres leeres Schlüsselbild ein. Das 3. Schlüsselbild enthält folgenden ActionScript Code:
 - `if (_root.Prozent >= 100)`
`{ gotoAndPlay ("Intro",1);}`
Wenn die vollen 100 Prozent geladen sind gehe zur Szene Intro Bild 1
 - `else`
`{gotoAndPlay (2);}`
Wenn die Bedingung von oben nicht erfüllt ist, dann gehe einfach wieder zum Bild 2 zurück
- In der Szene Intro fügen wir eine zweite Ebene hinzu. Diese nennen wir Aktionen, die vorhandene nennen wir Grafik.
- In der Ebene Script fügen wir bei Bild 2 folgenden ActionScript Befehl ein:
`stop();`

- Der Preloader mit Grafik: [flash/12_01.swf](#)  (190Kb) [flash/12_01 fla](#)  (209Kb)

Bandbreitenfilter im Filmtestmodus

- Wenn wir jetzt den Filmtestmodus aktivieren, spielt die Animation so schnell ab, dass wir leider vom Preloader nicht viel zu sehen bekommen. Deshalb aktivieren wir über Fehlersuche / 14.4 die Emulation eines Modems, welches unsere Animation mit 14.4 kbps aus dem Internet lädt.
- Dann aktivieren wir Ansicht / Streaming anzeigen, danach startet die Animation von neuem und wir können sehen, wie die Prozentzahl langsam bis 100 zählt und die Sonne sich von 0% auf 100% vergrößert.