

Übungsblatt 7

31 Feature Detection und jQueryUI

Verwenden Sie als Grundlage Aufgabe 9 von Übungsblatt 2. Ihre Aufgabe besteht nun darin, das Slider Element sowie den Datepicker durch jQuery UI anzuzeigen. Verwenden Sie daher eine Feature Detection, wie die aus der Vorlesung. Überprüfen Sie ob der Browser bereits die Attributwerte `type=date` und `type=range` versteht und führen Sie gegebenenfalls den jQuery UI Code als Fallback ein.

Bauen Sie des weiteren noch einen Testbutton ein, um auf modernen Browsern manuell in den Fallback-Modus umzuschalten. Dieser Button soll nur angezeigt werden, wenn die Seite ohne Fallback läuft.

32 Dialoge mit jQueryUI

Erstellen Sie eine einfache Seite in der Sie mit Hilfe von jQuery UI Dialoge einbauen. Folgende Dialoge sollen bei Clicks auf entsprechende Buttons erscheinen:

- Ein modaler (d.h. blockender) Dialog, der einen “OK” und einen “Abbrechen” Button besitzt. Größe und Position sollen fest eingestellt sein. Der Dialog soll zentriert angezeigt werden.
- Ein normaler Dialog (nennt man nicht-modal), dessen Größe und Position verändert werden kann.

Daneben soll noch ein (modaler) Dialog beim Betreten der Seite geöffnet werden (kurze Begrüßung).

33 JSONP - Mastered

In dieser Aufgabe sollen Sie Fotos des Internet-Dienstes Flickr auslesen und die ersten drei Bilder darstellen. Dies soll über JSONP (und jQuery) erreicht werden. Hierfür werden Sie folgende Url benötigen:

`http://www.flickr.com/services/feeds/photos_public.gne`

Zusätzlich benötigen Sie noch die Parameter `jsoncallback=?` und `format=json`. Die von Flickr erhaltenen Daten geben ein normales JSON Objekt zurück, welches ein Array (genannt items) enthält. Folgende Daten für den *i*-ten Eintrag sind daher noch sinnvoll:

- Link zu den Bild auf Flickr: `items[i].link`

- Quelle des Bildes: `items[i].media.m`

34 Geolocation einsetzen

In dieser Aufgabe sollen Sie die Geolocation in eine Seite einbauen. Fügen Sie hierzu einen Button “Position ermitteln” in eine Seite ein. Beim Drücken des Buttons soll der Browser die aktuelle Position ermitteln und diese dann in einer (readonly) Textbox ausgeben.

35 Eine einfache Notizen App

Durch Ausnutzen des `localStorage` Objekts kann man sehr schön beliebige Objekte (über JSON – bis zu einer bestimmten Größe) speichern. Erstellen Sie ein Formular, das folgende Felder besitzt:

- Titel
- Beschreibung
- Priorität

Mit einem Klick auf *Hinzufügen* sollen Sie Einträge erstellen (Erstellen eines neuen Storage Eintrags, inkl. Erstelldatum und letztem Änderungsdatum – alles als serialisiertes JSON Objekt). Die erstellten Einträge sollen in einer separaten Liste (*<aside>*) angezeigt werden und sich bei Klick in das Formular laden. Anstelle des *Hinzufügen* Buttons soll dann ein *Ändern* Button angezeigt werden.