

Web-Informationssysteme, WS 2009/10
Übungsblatt 2

Besprechung am Di 03.11.2009

Aufgabe 2-1 CSS Selektoren

Zur Erinnerung: Selektoren bestimmen in einer CSS Regel für welche Elemente diese Regel angewandt werden soll. Geben Sie für jeden der folgenden Selektoren eine natürlichsprachliche Beschreibung derjenigen Elemente an, mit denen der Selektor matched. Einige der Selektoren sind bereits aus der nächsten Revision von CSS, CSS 3, siehe <http://www.w3.org/TR/css3-selectors/>.

- a) $s_1 := \text{body} > \text{ol} > \text{li} \text{ p}$
- b) $s_2 := *: \text{not}(\text{a})$
- c) $s_3 := \text{p}: \text{not}(\text{.section})$
- d) $s_4 := \text{body} > \text{h2}: \text{not}(: \text{first-of-type}): \text{not}(: \text{last-of-type})$
- e) $s_5 := \text{ul li}: \text{nth-child}(2n+3): \text{not}(: \text{last-child})$
- f) $s_6 := \text{ol li}: \text{nth-child}(-3n+9)$
- g) $s_7 := \text{ol li}: \text{nth-child}(7n-3)$
- h) $s_8 := \text{button}: \text{not}([\text{DISABLED}])$
- i) $s_9 := \text{html} | *: \text{not}(: \text{link})$
- j) $s_{10} := *$
- k) $s_{11} := \text{h1\#nav.menu ol} + *[\text{rel}=\text{"up"}].$
- l) $s_{12} := \text{\#nav.menu}$
- m) $s_{13} := * | *\text{\#nav.menu}$
- n) $s_{14} := \text{ul} > \text{li.author} \sim \text{li.journal}$
- o) $s_{15} := : \text{root} > : \text{empty}$
- p) $s_{16} := \text{html} > : \text{empty}$

Die Spezifität von CSS Selektoren wird durch die folgende Formel definiert (nach <http://www.w3.org/TR/css3-selectors/#specificity>):¹

$$\text{spec}(s) = |\text{ID}(s)| \cdot b^2 + (|\text{class}(s)| + |\text{attr}(s)| + |\text{pseudocl}(s)|) \cdot b + |\text{type}(s)| + |\text{pseudoel}(s)|$$

$\text{ID}(s)$ = ID Selektoren in s (z.B. `#nav`)
 $\text{class}(s)$ = Class-Selektoren in s (z.B. `.menu`)
 $\text{attr}(s)$ = Attribut-Selektoren in s (z.B. `[rel]`)
 $\text{type}(s)$ = Typ-Selektoren in s (z.B. `ul`) ohne `*`
 $\text{pseudocl}(s)$ = Pseudo-Klassen in s (z.B. `:root`) ohne `not`
 $\text{pseudoel}(s)$ = Pseudo-Elemente in s (z.B. `::first_line`)
 $b = \max\{|\text{ID}(s)|, |\text{class}(s)| + |\text{attr}(s)| + |\text{pseudocl}(s)|, |\text{type}(s)| + |\text{pseudoel}(s)|\}$

- q) Ermitteln Sie für die oben angegebenen Selektoren s_1 bis s_{16} jeweils die Spezifität gemäß dem obigen Algorithmus (zur Vereinfachung können Sie $b = 10$ verwenden).

Aufgabe 2-2 DOM (Document Object Model)

Machen Sie sich mit den W3C Empfehlungen zum Document Object Model (kurz: DOM) vertraut.

Welche der folgenden Aussagen über das DOM sind im Allgemeinen wahr?

- Das DOM ist eine sprachunabhängige Schnittstelle für Programme und Skripte. Es dient zum dynamischen Zugriff und zur Veränderung des Inhalts, der Struktur und des Stils von Dokumenten.
- Das DOM stellt Schnittstellen zu einer Menge von vordefinierten Objekten zur Repräsentation von HTML und XML Dokumenten bereit.
- Da DOM genau vorschreibt wie Datenstrukturen für HTML Dateien implementiert werden, wird die Interoperabilität im Web unterstützt.
- DOM Level 3 besteht aus den vier Empfehlungen DOM Level 3 Core, DOM Level 3 HTML, DOM Level 3 Load and Save, DOM Level 3 Validation.
- Die Knotentypen in einer DOM-Repräsentation eines HTML Dokuments (Dom Level 3) sind Document, DocumentFragment, DocumentType, EntityReference, ProcessingInstruction, Attr, Element, Comment, Text, CDataSection, Entity, Notation.
- Ein Document Node im DOM kann als Kinder maximal ein Element, beliebig viele Processing Instructions, beliebig viele Kommentare, beliebig viele Attribute und maximal eine Dokumenttypdeklaration haben.
- Das Document Object Model sieht Attribute als keine tatsächlichen Knoten mit eigener Identität im XML Baum, sondern als Eigenschaften der Elemente, zu denen sie gehören. Deswegen sind die Eigenschaften parentNode, previousSibling, und nextSibling eines Attributobjekts im DOM mit null belegt.
- DOM Level 3 stellt eine Möglichkeit zur Verfügung direkt auf die Kinder eines Knotens zuzugreifen.
- DOM Level 3 stellt eine Möglichkeit zur Verfügung direkt auf alle Elemente mit einem bestimmten Tag im Dokument zuzugreifen.

¹Diese Fassung weicht etwas von CSS 2 (<http://www.w3.org/TR/CSS2/cascade.html#specificity>) ab.

- j) DOM Level 3 stellt eine Möglichkeit zur Verfügung direkt auf die Kinder eines Knotens, die ein bestimmtes Tag haben, zuzugreifen.
- k) Attributwerte eines Elements erhält man im DOM Level 3 durch `getAttribute`, Attributknoten eines Elements durch die Methode `getAttributeNode`.

Aufgabe 2-3 DOM und Javascript

In den Anlagen zum Übungsblatt finden Sie eine HTML Datei `studentenleben.html`, in deren Body sich eine ungeordnete Liste befindet. Im Kopf der Datei befindet sich ein leeres `script` Element. Schreiben Sie in dieses Element ein Javascript Programm, welches mit Hilfe der DOM API die Liste zum Rotieren bringt. Zur Lösung dieser Aufgabe können Ihnen unter anderem die Methoden und Attribute `document.getElementById`, `removeChild`, `appendChild`, `window.setTimeout`, `nodeType` und `nextSibling` helfen.

```

1 <html>
2 <head>
3   <title>Das Studentische Leben -- Ein Karusell der Abwechslung</title>
4   <script type="text/javascript">
5
6   </script>
7 </head>
8 <body>
9   <h1>Das Studentische Leben</h1>
10  <h3>Ein Karusell der Abwechslung</h3>
11  <p>
12    Der Student erfreut sich eines unbeschwerten und abwechslungsreichen Lebens:
13  </p>
14  <ul id="studentenLeben">
15    <li><em>Studieren</em></li>
16    <li>Essen</li>
17    <li><strong>Feiern</strong></li>
18    <li>Schlafen</li>
19  </ul>
20 </body>
21 </html>

```

Aufgabe 2-4 Auswertung Such-Maschinen

Angenommen Sie wollen Ihren Urlaub in Mexiko planen und wollen etwas über *jaguars* lernen!

- a) Gehen Sie auf `www.google.com` (nicht `google.de`!) und geben Sie die Suche "jaguars" ein. Notieren Sie wie viele der zurückgegebenen Links auf der ersten Seite relevant für Ihre Suche sind. Analog für Seite 10.
- b) Nehmen Sie an, dass die N Links, die Sie auf der 1. und 10. Seite sehen, die einzigen sind, die von Google zurückgegeben werden. Berechnen Sie die *precision* für die ersten 5, 10, 15, und N Links.
- c) Vollziehen Sie dieselbe Analyse für eine alternative Suchmaschine Ihrer Wahl (z.B. `ask.com`, `bing.com`). Achten Sie darauf die amerikanische Version zu verwenden!
- d) Unter der Annahme, dass Sie nur an *precision* interessiert sind, welche der beiden Suchmaschinen liefert die besseren Ergebnisse. Welche Ergebnisse erscheinen Ihnen besser aus Ihrer Perspektive? Welche aus der Perspektive des Suchmaschinen-Entwicklers?

- e) Warum haben wir in den obigen Übungen den *recall* nicht berechnet?
- f) Wie behandelt Google **Stopwörter**²? Probieren Sie einige Beispiele aus und geben Sie den jeweiligen Effekt des Stopworts an.
- g) Überlegen und verifizieren Sie eine Beispielanfrage, bei der die Reihenfolge der Keywords in der Query das Ergebnis beeinflußt (und bei der dieses Verhalten Sinn macht) und eine, bei der die Reihenfolge keinen Effekt haben sollte.

²Zur Erinnerung: Stopwort = typischerweise durch Suchmaschine ignoriertes, hoch-frequentes, aber wenig bedeutungstragendes Wort (z.B. "dass").

Aufgabe 2-5 W3C Prozess

Machen Sie sich mit den Seiten des W3C vertraut und beantworten Sie entsprechend:

- Wie nennen sich die vom W3C herausgegebenen "Standards"?
- Welche Phasen durchläuft ein solches Dokument bis zur endgültig "verbindlichen" Fassung?
- Welches ist die aktuellste HTML Variante/Fassung?

Vervollständigen Sie mit dem dabei gewonnenen Wissen die folgende Übersicht über den W3C Prozess:

