

Internet and Web Applications, Übung 1

HENRY HAUSTEIN

Aufgabe 1: XML/XML Schema

(a) XML-Dokument

```
1  <?xml version="1.0" encoding="UTF-8"?>
2
3  <newspaper>
4    <article id="5">
5      <title>First man on the moon</title>
6      <published>1969-07-21</published>
7      <author>Ted Markson</author>
8    </article>
9    <article id="6">
10     <title>Curiosity lands on Mars</title>
11     <published>2012-08-06</published>
12     <author>Jim Northwalk</author>
13   </article>
14 </newspaper>
```

(b) XML Schema

```
1  <?xml version="1.0" encoding="UTF-8"?>
2
3  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" >
4    <xs:element name="article">
5      <xs:complexType>
6        <xs:sequence>
7          <xs:element name="title" type="xs:string"/>
8          <xs:element name="published" type="xs:date"/>
9          <xs:element name="author" type="xs:string"/>
10        </xs:sequence>
11      </xs:complexType>
12    </xs:element>
13    <xs:element name="newspaper">
14      <xs:complexType>
15        <xs:sequence>
16          <xs:element name="article" type="article"/>
17        </xs:sequence>
18      </xs:complexType>
19    </xs:element>
20  </xs:schema>
```

Aufgabe 2: YAML/JSON

```
1  [  
2    {  
3      "id": 5,  
4      "title": "First man on the moon",  
5      "published": 1969-07-21,  
6      "author": "Ted Markson"  
7    }, {  
8      "id": 6,  
9      "title": "Curiosity lands on Mars",  
10     "published": 2012-08-06,  
11     "author": "Jim Northwalk"  
12   }  
13 ]
```

Aufgabe 3: DOM/SAX

- (a) DOM ist flexibler, aber dafür muss die gesamte Struktur des Dokuments in den Speicher geladen werden. SAX braucht das nicht, ist aber auch dafür nicht so schnell.
- (b) Eine Alternative ist XPath. Hier wird auch das Dokument als Baum betrachtet, aber es erlaubt mehr Operationen (z.B. 5. Kind-Element, Durchschnitt aller Kinder, ...).
Das Java API for XML Processing oder JAXP ist eine der Java-XML-APIs. Es handelt sich um ein leichtgewichtiges standardisiertes API zum validieren, parsen, generieren und generieren von XML-Dokumenten. Die jeweilige Implementierung des APIs ist austauschbar.

Aufgabe 4: Javascript

- (a) Javascript ist Code, der nicht auf dem Server, sondern auf dem Client ausgeführt wird. Damit kann man die Webseite dynamisch verändern ohne diese neuladen zu müssen, aber zu viel Javascript kann auch den ganzen Rechner blockieren (Werbung...) oder ist ein Einfallstor für Schadware, wenn die Sandbox nicht richtig funktioniert.
- (b) HTML-Seite

```
1  <!DOCTYPE html>  
2  <html lang="de">  
3    <head>  
4      <meta charset="UTF-8">  
5      <meta name="viewport" content="width=device-width, initial  
6        -scale=1.0">  
7      <script src="javascript.js"></script>  
8      <title>Button Dokument</title>  
9    </head>  
10   <body>  
11     <main>  
12       <button onclick="showDate()" id="button">klick mich!</  
13         button>  
14     </main>  
15   </body>
```

```
14 </html>
```

Javascript

```
1 function showDate() {  
2     var today = new Date()  
3     var date = today.getFullYear() + " - " + (today.getMonth() +  
4         1) + " - " + today.getDate()  
5     document.getElementById("button").innerHTML = date  
6 }
```

(c) HTML-Seite mit Javascript

```
1 <!DOCTYPE html>  
2 <html lang="de">  
3     <head>  
4         <meta charset="UTF-8">  
5         <meta name="viewport" content="width=device-width, initial  
6             -scale=1.0">  
7         <title>Liks</title>  
8     </head>  
9     <body>  
10        <main>  
11            <a href="xyz.de">xyz</a><br />  
12            <a href="abc.de">abc</a>  
13        </main>  
14    </body>  
15    <script src="javascript.js"></script>  
16 </html>
```

Javascript

```
1 elems = document.getElementsByTagName("a")  
2 for (elem of elems) {  
3     elem.insertAdjacentText("afterend", elem.innerHTML)  
4 }
```

(d) HTML-Seite mit Javascript

```
1 <!DOCTYPE html>  
2 <html lang="de">  
3     <head>  
4         <meta charset="UTF-8">  
5         <meta name="viewport" content="width=device-width, initial  
6             -scale=1.0">  
7         <title>Liks</title>  
8     </head>  
9     <body>  
10        <main>  
11            <a type="text/html">xyz</a><br />  
12            <a type="application/json">abc</a>  
13        </main>  
14    </body>
```

```
14     <script src="javascript.js"></script>
15 </html>
```

Javascript

```
1  elems = document.getElementsByTagName("a")
2  for (elem of elems) {
3      elem.insertAdjacentText("afterend", elem.type)
4  }
```