

1 Explanation of DOM-Tree

Siehe Beiliegendes .js und .html

Für unser .html, welches so aussieht:

```
<a href="http://www.orf.at/" target="_new">Link zum ORF</a><br>
<a href="http://www.sms.at" target="_new">Link zu sms.at</a><br>
<a href="http://www.aau.at" target="_new">Link zur AAU Seite</a>
<script src="DOMTree.js"></script>
```

.. wurde ein .js definiert, welches in das .html eingebunden wird.

Das .js beginnt mit:

```
var link = document.getElementsByTagName("a")[2];
```

.. einer Schaffung und Wertzuweisung der Variable 'link'. 'link' wird dem 3. Element des Arrays gleichgesetzt, welches rechtsseitiger definiert wird.

e.g.: var link = urls[2] - `< a > this.data < /a >`

◇ Block 1

```
alert (link.attributes["href"].nodeValue);
alert (link.getAttributeNode("href").nodeValue);
alert (link.href);
```

Alle 3 Zeilen machen dasselbe: Sie geben das Argument des href-Wertes wieder. `link.attributes["href"].nodeValue` greift direkt auf den Link zu, welcher in `href="..."` steht. `link.getAttributeNode("href").nodeValue` greift über die Attributnode "href" zu und schnappt sich dessen Wert: Der Text von `href="x"`. `link.href` ist der spezifischste Zugriff, direkt auf href des Links.

◇ Block 2

```
alert(link.firstChild.data);
```

Diese Zeile gibt die Kind-Data wieder, in unserem Link ist das der inbeschriebene Text: 'Link zur AAU Seite'

◇ Block 3

```
alert (link.attributes[0].nodeType);
alert (link.firstChild.nodeType);
```

Die erste Zeile dieses Blocks beschreibt den NodeType, welcher 2 ist. '2' steht hier lediglich für ein Element, dessen Attribute per se beschrieben werden mit '2'. Gemäß Standard (DOM-L4), gibt diese Abfrage immer 2 zurück.¹ Zweitere gibt den Typ des ersten Kindes wieder. Das Kind unseres `< a > < /a >` ist der darinliegende Text. Die Attributzahl von Text ist 3.

¹<https://developer.mozilla.org/de/docs/Web/API/Node/nodeType>

◇ Block 4

```
alert(link.parentNode.nodeName);  
alert(link.lastChild.parentNode.nodeName);
```

Die erste Zeile beschreibt den Namen des Parent-Node. Der Parent-Node unseres `< a >< /a >` ist der `< body >< /body >`, dementsprechend liefert dies BODY zurück. `Parent-Node.nodeName` gibt den Namen der Parent-Node des letzten Kindsknoten wieder. Hier 'A'.

◇ Block 5:

```
alert(link.attributes.length);
```

Diese Zeile gibt lediglich die Anzahl der Attribute wieder, die im Link spezifiziert wurden. mit href und target bestehen hier 2 Attribute, wodurch der Wert 2 ist.