

# Bearbeiten von XML-Dateien in PHP

Kürzlich ist bei mir der Bedarf entstanden, aus PHP heraus XML Dateien einzulesen und zu modifizieren. Im Wesentlichen geht es darum, vorhandene Werte zu ändern, Attribut inkl. Wert zu löschen oder neue Attribut/Wert-Kombinationen hinzuzufügen. Das geht mit Hilfe der PHP-Erweiterung SimpleXML denkbar einfach. SimpleXML wird, falls nicht manuell deaktiviert, mit PHP installiert.

Als Grundlage für die folgenden Absätze nutzen wir eine XML-Datei „00\_example.xml“, die folgenden Aufbau hat und im Unterverzeichnis „xml“ liegt:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- XML Beispiel "xml/00_example.xml" -->
3 <root>
4     <person></person>
5     <name></name>
6     <vorname></vorname>
7     <zusatzinfo>
8         <spitzname></spitzname>
9         <spitzname></spitzname>
10        <spitzname></spitzname>
11    </zusatzinfo>
12 </root>
```

Es ist gegeben, dass sich die Sache mit den Spitznamen nicht anders lösen lässt, als mit multiplen Werten für das Attribut <spitzname>, abgelegt unter <zusatzinfo>. Nun möchte ich diese Beispieldatei mit Inhalt füllen und stelle dabei noch fest, dass für einen korrekten Eintrag meiner Testperson ein zusätzlicher Spitzname notwendig ist. Der folgende Code liest die Datei in PHP ein, setzt Werte für die XML-Elemente, fügt einen weiteren Spitznamen hinzu und speichert die Datei abschließend unter einem Neuen namen ab.

```
1 <?php
2     // Datei mit SimpleXML einlesen
3     $modXml = simplexml_load_file("xml/00_example.xml");
4     // Wert für <person></person> setzen
5     $modXml->person = 1;
6     // Wert für <name></name> setzen
7     $modXml->name = "Test";
8     // Wert für <vorname></vorname> setzen
9     $modXml->vorname = "Person";
10    // Werte für <spitzname></spitzname> setzen
11    $modXml->zusatzinfo->spitzname[0] = "T";
12    $modXml->zusatzinfo->spitzname[1] = "Tes";
13    $modXml->zusatzinfo->spitzname[2] = "TP";
14    $modXml->zusatzinfo->spitzname[3] = "T-Pez";
15
16    // Datei abspeichern
17    $modXml->asXML("xml/01_output.xml");
18 ?>
```

Unser Freund „Test Person“ hat sich nun gemeldet und sich über seinen neuesten Spitznamen „T-Pez“ beschwert, also lesen wir die Datei mit seinen Angaben erneut ein und löschen den letzten Spitznamen. Wie im obigen Beispiel zu sehen ist, ist das erste Element mit „...spitzname[0]“ zu adressieren. Also muss „spitzname[3]“ gelöscht werden. Die Datei wird nun wieder gespeichert.

```
1 <?php
2     // Datei mit SimpleXML einlesen
3     $modXml = simplexml_load_file("xml/01_output.xml");
4     // Anzahl von <spitzname> anzeigen
5     $elementCount = $modXml->zusatzinfo->spitzname->count();
6     echo "Die Person hat ".$elementCount." Spitznamen. "; // <-- Das Ergebnis ist: 4
7     // Viertes Element <spitzname> löschen
8     unset($modXml->zusatzinfo->spitzname[3]);
9     $elementCount = $modXml->zusatzinfo->spitzname->count();
10    echo "Die Person hat ".$elementCount." Spitznamen. "; // <-- Das Ergebnis ist: 3
11    // Datei erneut speichern
12    $modXml->asXML("xml/01_output.xml");
13 ?>
```

Die neu gespeicherte Datei sieht nun wie folgt aus:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- XML Datei "xml/01_output.xml" -->
3 <root>
4     <person>1</person>
5     <name>Test</name>
6     <vorname>Person</vorname>
7     <zusatzinfo>
8         <spitzname>T</spitzname>
9         <spitzname>Tes</spitzname>
10        <spitzname>TP</spitzname>
11    </zusatzinfo>
12 </root>
```