



# Inhaltsverzeichnis

Definition.....	3
Einsatzgebiet und Nutzen von XML.....	4
Regeln von XML.....	5
Gültigkeit von XML.....	6
Beispiele.....	7



## Definition

Extensible Markup Language (deu. erweiterbare Auszeichnungssprache) oder kurz XML ist eine Auszeichnungssprache, die Daten hierarchisch strukturiert in Form von Textdateien.

Zusätzlich ist XML die Grundlage für eine neue Generation von webbasierten Anwendungen zum Anzeigen und Bearbeiten von Daten.



# Einsatzgebiet und Nutzen von XML

XML dient hauptsächlich dem Datenaustausch zwischen Computersystemen und wird meist im Internet oder Intranet verwendet, um Daten einzupflegen. Dies können Dokumente, Konfigurationen, Bücher, Rechnungen und vieles mehr sein.

XML wird verwendet, um Daten lokal an einem Ort zu speichern und somit schneller Änderungen vornehmen zu können. Zudem können die Daten auf verschiedene Arten bearbeitet werden anstatt sie lediglich darzustellen. Sie können ohne zusätzliche Zugriffe auf den Server berechnet werden.

Dadurch dass sich die Daten lokal an einer Stelle befinden entstehen weniger Fehler bei Änderungen und es ist deutlich übersichtlicher.



# Regeln von XML

Der Begriff „Wohlgeformtheit“, der Ihnen im Zusammenhang mit XML immer wieder begegnen wird, bedeutet, dass eine Datei die Regeln von XML korrekt einhält.

Für Tags, Attribute und ihre Wertzuweisungen gibt es in XML bestimmte Regeln, die Sie kennen müssen.

- XML unterscheidet im Gegensatz zu klassischem HTML strikt zwischen Groß- und Kleinschreibung.
- Bei XML gilt ohne Ausnahme: alle Wertzuweisungen an Attribute müssen in Anführungszeichen stehen.
- In XML gibt es keine alleinstehenden Tags wie in HTML beispielsweise `<br>`, `<img>` oder `<hr>`
- jedes Element muss aus Anfangs- und End-Tag bestehen (Bsp. `<tag> </tag>`)
- Es gibt in XML auch keine optionalen End-Tags, wie es in HTML etwa bei `</li>` oder `</option>` der Fall ist
- Ausnahmen sind so genannte leere Elemente, also Elemente ohne Inhalt. Solche Elemente können ohne zugehöriges End-Tag notiert werden (Bsp. `<linie>`)
- Elementnamen dürfen nicht mit der Zeichenfolge `xml` und nicht mit Ziffern beginnen, und sie dürfen keine Gleichheitszeichen und keine Leerzeichen enthalten
- Ein Kommentar beginnt mit der Zeichenfolge `<!--Kommentar-->`. Er kann sich über mehrere Zeilen erstrecken. Was innerhalb des Kommentars steht, wird nicht interpretiert.



# Gültigkeit von XML

Bei XML kann ein Dokument mit DTDs (Document Type Definitions = Definitionen des Dokumenttyps) versehen werden, die im Wesentlichen die Regeln des Dokuments definieren, z.B. welche Elemente vorhanden sind und welche strukturelle Beziehung zwischen den Elementen besteht. DTDs sind für die Validierung der Daten nützlich, wenn die Zielanwendung nicht über eine integrierte Beschreibung der eingehenden Daten verfügt. DTDs können bei XML optional verwendet werden.

Daten, die zusammen mit einer DTD gesendet werden, werden als „gültige“ XML-Daten bezeichnet. Ein XML-Parser kann die eingehenden Daten anhand der Regeln in der DTD prüfen, um sicherzustellen, dass die Daten richtig strukturiert wurden. Daten, die ohne eine DTD gesendet werden, werden als „ausformatiert“ bezeichnet. Hierbei kann eine Instanz eines XML-basierten Dokuments, z.B. die hierarchisch strukturierten Wetterdaten, verwendet werden, um sich selbst implizit zu beschreiben.

XML-codierte Daten sind sowohl für gültige als auch für ausformatierte XML-Dokumente selbstbeschreibend, da die Daten mit beschreibenden Tags durchsetzt sind. Durch das von XML verwendete offene und flexible Format kann diese Codierung überall dort angewendet werden, wo Informationen ausgetauscht und übertragen werden müssen. Dadurch wird XML extrem leistungsstark.



# Beispiele

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<!DOCTYPE adresse SYSTEM „adresse.dtd“>
```

```
<adresse> _____ Anfangs-Tag
```

Anfang 1. Adresse

```
<einzeladresse>
```

```
<name>Müller</name>
```

```
<vorn>Margaret</vorn>
```

```
<strasse>Nirgendwostr. 15</strasse>
```

```
<plz>10000</plz>
```

Klasse

```
<ort>Nirgendwo</ort>
```

```
<tel art="priv">00000/56789</tel>
```

```
<email>l.mueller@provider.net</email>
```

Weitere Angaben zur Einzeladresse beinhalten die Bedeutung des jeweiligen Anfangs- und Endtags z.B. <plz></plz> beinhaltet die PLZ usw.

Ende 1. Adresse

```
</einzeladresse>
```

```
<!-- Das ist ein Kommentar --> _____ Kommentar
```

```
<einzeladresse> _____ Anfang 2. Adresse
```

```
<name>Müller</name>
```

```
<vorn>Wolfgang</vorn>
```

```
<strasse>Nirgendwostr. 20</strasse>
```

```
<plz>10000</plz>
```

```
<ort>Nirgendwo</ort>
```

```
<tel>00000/24680</tel>
```

```
<email>wm@provider.com</email>
```

Weitere Angaben zur Einzeladresse beinhalten die Bedeutung des jeweiligen Anfangs- und Endtags z.B. <ort></ort> beinhaltet den Ort usw.

```
</einzeladresse> _____ Ende 2. Adresse
```

```
End-Tag _____ </adresse>
```

