

T-SQL

Transact Structured Query Language

```
IP: 88.217.70.57
DNS: ppp-88-217-70-57.dynamic.mnet-on
Browser: Netscape
OS: Mac OS
http://myip.tsq.de
```

[T-SQL](#)[C#](#)[PHP](#)[Visual Basic](#)[C++](#)[Sonstiges](#)

[Startseite](#)
[T-SQL](#)
[C-Sharp](#)
[PHP](#)
[Visual Basic](#)
[C++](#)
[Download](#)
[Kontakt](#)
[Impressum](#)
[Datenschutz](#)

[Google-Anzeigen](#)

[Csharp Tutor](#)
[Csharp XML](#)
[C# Schulung](#)
[UML C#](#)

[Was ist XML](#)[Attribute eines XML-Dokumentes mit C# auslesen](#)[C# - XML Dokumentationskommentare](#)[C# - XML schreiben - XMLTextWriter](#)[XmlException](#)

Erzeugen eines XML-Dokuments mit C#

Das kleinste XML-Dokument würde wohl aus einem leeren Rootelement bestehen. Wir erzeugen dieses kleinste, quasi "helloXMLWorld" Dokument mit CSharp Code. Es wird ein leeres Rootelement erzeugt und zum Dokument hinzugefügt. Anschließend speichern wir das XML-Dokument als Datei ab.

[Google-Anzeigen](#)[C# Lehrer](#)[C# Matrix](#)[C# Datenbank](#)[C# Tutor](#)

XML

```
<HelloXMLWorld />
```

C# CODE

```
using System.Xml;
...
void Csharp_HelloWorld_XML()
{
    XmlDocument doc = new XmlDocument();
    XmlNode myRoot;
    myRoot = doc.CreateElement("HelloXMLWorld");
    doc.AppendChild(myRoot);
    doc.Save(@"c:\helloxmlworld.xml");
}
```

Erzeugen von Unterknoten

Mit dem Befehl "AppendChild" werden dem Rootelement weitere Untereinträge hinzugefügt. Im Beispiel wird die Root um vier leere Kindknoten erweitert. Bei den Varianten Child3 und Child4, wird das erzeugte Objekt direkt an die Methode AppendChild weitergegeben. Das Ergebnis ist bei allen Childnodes das gleiche, bei den beiden letzten können aber Variablen und Codezeilen gespart werden, was die Übersichtlichkeit erhöht.

XML

```
<appendchild>
<child1 />
<child2 />
<child3 />
<child4 />
</appendchild>
```

C# CODE

```
using System.Xml;
...
void Csharp_AppendChild()
{
    XmlDocument doc = new XmlDocument();
    XmlNode myRoot, myNode;

    myRoot = doc.CreateElement("AppendChild");
    doc.AppendChild(myRoot);

    myNode = doc.CreateElement("Child1");
    myRoot.AppendChild(myNode);

    myNode = doc.CreateElement("Child2");
    myRoot.AppendChild(myNode);

    myRoot.AppendChild(doc.CreateElement("Child3"));
    myRoot.AppendChild(doc.CreateElement("Child4"));

    doc.Save(@"c:\AppendChild.xml");
}
```

Einfügen von Text und Attributen

Im Beispiel werden der InnerText und eine Attribut angelegt. Child1 zeigt dabei die ausgiebige Variante. Bei Child2 wird die kurze Variante demonstriert.

XML

```
<AppendChild>
<Child1 Attribute1="AttributeText1">Text1</Child1>
```

```
<Child2 Attribute2="AttributeText2">Text2</Child2>
</AppendChild>
```

C# CODE

```
using System.Xml;
...
void Csharp_Text_Attribute()
{
    XmlDocument doc = new XmlDocument();
    XmlNode myRoot, myNode;
    XmlAttribute myAttribute;

    myRoot = doc.CreateElement("AppendChild");
    doc.AppendChild(myRoot);

    myNode = doc.CreateElement("Child1");
    myNode.InnerText = "Text1";

    myAttribute = doc.CreateAttribute("Attribute1");
    myAttribute.InnerText = "AttributeText1";
    myNode.Attributes.Append(myAttribute);

    myRoot.AppendChild(myNode);

    myRoot.AppendChild(doc.CreateElement("Child2").InnerText = "Text2";
    myRoot.SelectSingleNode("Child2").Attributes.Append
        (doc.CreateAttribute("Attribute2").InnerText = "AttributeText2";

    doc.Save(@"c:\TextAttribute.xml");
}
...
```

[Google-Anzeigen](#)[C# Excel](#)[C# Seminar](#)[Csharp Tutor](#)[Csharp XML](#)