

DOM چیست ؟

Document (Object Model DOM) یک رابط برنامه نویسی برای سندهای XML و Html است . با استفاده از اینترفیس فوق، نحوه دستیابی و انجام پردازش های لازم در رابطه با سند های XML و Html فراهم می گردد . برنامه نویسان با استفاده از DOM ، قادر به ایجاد یک سند ، حرکت در طول ساختار سند، افزودن ، اصلاح و یا حذف المان های یک سند XML و یا Html می باشند . DOM توسط کنسرسیوم وب استاندارد و بمنظور استفاده از طریق زبان های برنامه نویسی متعددی طراحی شده است .

گره ها

برنامه ای با نام پارسر، امکان استقرار یک سند XML در حافظه را فراهم می نماید . پس از استقرار سند در حافظه ، اطلاعات مربوطه بکمک DOM ، قابل بازیابی و پردازش خواهد بود . DOM ، یک نمایش شبه درختی از یک سند XML را ایجاد می نماید . المان Document ، دارای بالاترین سطح در درخت ایجاد شده است . المان فوق ، دارای یک و یا چندین فرزند است . یک اینترفیس گره ، امکان خواندن و یا نوشتن المان های خاصی در ساختار درختواره ای یک سند XML را میسر می نماید . با استفاده از خصلت ChildNodes مربوط به المان Document و بکارگیری یک حلقه تکرار می توان هر یک از گره ها را انتخاب و در ادامه عملیات مورد نظر در رابطه با آن را انجام داد . پارسر شرکت ماکروسافت (MSXML) ، دارای توابع متفاوت بمنظور حرکت در طول درخت، دستیابی به گره ها به همراه خصلت مربوطه، درج و حذف گره ها و تبدیل ساختار درختی به شکل اولیه XML است . در حال حاضر، سیزده نوع متفاوت گره توسط پارسر شرکت ماکروسافت، حمایت می گردد . جدول زیر متداولترین نوع گره ها را نشان می دهد .

نوع گره	مثال
Document type	<!DOCTYPE food SYSTEM "food.dtd">
Processing instruction	<?xml version="1.0"?>
Element	<drink type="tea">Irani</drink>
Attribute	type="tea"

Irani	Text
-------	------

استفاده از پارسر

بمنظور خواندن، ویرایش و یا ایجاد و پردازش یک سند XML، می بایست از یک پارسر استفاده کرد. پارسر شرکت ماکروسافت، یک عنصر مبتنی بر COM بوده که به همراه IE نسخه پنج، ارائه شده است. پس از نصب مرورگر فوق، امکان استفاده از پارسر فراهم می گردد. پارسر MSXML، دارای امکانات متعدد برنامه نویسی بمنظور استفاده توسط زبانهای زیر است:

- حمایت از جاوااسکریپت VBScript Perl جاوا و C++
- حمایت از استاندارد کنسرسیوم وب و XML DOM
- حمایت از DTD و معتبر سازی

مثال: نحوه ایجاد شی Document، با استفاده جاوااسکریپت، VBScript و ASP در جدول زیر نشان داده شده است:

تکنولوژی	مثال
Javascript	var xmlDoc = new ActiveXObject("Microsoft.XMLDOM")
VBscript	set xmlDoc = CreateObject("Microsoft.XMLDOM")
ASP	set xmlDoc = Server.CreateObject("Microsoft.XMLDOM")

استقرار یک سند XML در حافظه

کدهای زیر نحوه استقرار یک سند XML با نام Test.xml در حافظه را نشان می دهد:

استقرار یک سند XML در حافظه
<pre><script language="JavaScript"> var xmlDoc = new ActiveXObject("Microsoft.XMLDOM")</pre>

```
xmlDoc.async="false"
xmlDoc.load("Test.xml")

// پردازش های لازم .....

</script>
```

اولین خط در اسکریپت های فوق ، یک نمونه از پارسر XML شرکت ماکروسافت را ایجاد می نماید . در دومین خط ، به پارسر اعلام می گردد منتظر دریافت کامل سند XML بوده قبل از اینکه پردازش خود را آغاز نماید . در سومین خط ، به پارسر اعلام شده است که یک سند XML با نام Test.xml را در حافظه مستقر نماید .

استقرار متن XML در پارسر

کدهای زیر، نحوه استقرار یک رشته متن را در پارسر XML نشان می دهد . در مثال فوق از متد LoadXML در مقابل متد load استفاده شده است . از متد loadXML بمنظور استقرار یک رشته متن در پارسر استفاده می گردد .

استقرار یک رشته متن XML در حافظه

```
<script language="JavaScript">
var text="<note>"
text=text+"<to>Ali</to><from>Reza</from>"
text=text+"<heading>Reminder</heading>"
text=text+"<body>Don't forget me this
weekend!</body>"
text=text+"</note>"
var xmlDoc = new
ActiveXObject("Microsoft.XMLDOM")
xmlDoc.async="false"
xmlDoc.loadXML(text)

// پردازش های لازم .....

</script>
```

شی ParseError

خطای مربوط به فایل : با استفاده از شی ParseError ، می توان از کد ، متن خطاء و حتی شماره خطی که خطاء در آن اتفاق افتاده است ، آگاهی یافت . در برنامه زیر پارسر، سعی در استقرار سندی XML در حافظه نموده که وجود خارجی ندارد . پس از تشخیص خطاء ، برخی از خصلت های مربوط به خطاء در خروجی نمایش داده خواهد شد .

تشخیص خطاء و استفاده از شی ParseError در رابطه با وجود یک فایل

```
var xmlDoc = new
ActiveXObject("Microsoft.XMLDOM")
xmlDoc.async="false"
xmlDoc.load("Test1.xml")
document.write("<br>Error Code: ")
document.write(xmlDoc.parseError.errorCode)
document.write("<br>Error Reason: ")
document.write(xmlDoc.parseError.reason)
document.write("<br>Error Line: ")
document.write(xmlDoc.parseError.line)
```

خطای مربوط به XML . فرض کنید یک سند XML که "خوش شکل " نمی باشد، توسط پارسر خوانده شود . کدهای زیر نحوه تشخیص و برخورد با خطاء را نشان می دهد :

تشخیص خطاء و استفاده از شی ParseError در رابطه با سند "خوش شکل

"

```
var xmlDoc = new
ActiveXObject("Microsoft.XMLDOM")
xmlDoc.async="false"
xmlDoc.load("note_error.xml")
document.write("<br>Error Code: ")
document.write(xmlDoc.parseError.errorCode)
```

```
document.write("<br>Error Reason: ")
document.write(xmlDoc.parseError.reason)
document.write("<br>Error Line: ")
document.write(xmlDoc.parseError.line)
```

خصلت های ParseError

عملکرد	خصلت
کد خطای بوجود آمده ، برگردانده می شود .	errorCode
علت خطای بوجود آمده ، برگردانده می شود .	reason
شماره خط خطای بوجود آمده ، برگردانده می شود	line
محل بروز خطاء در خط مربوطه را برمی گرداند .	linePos
رشته ای که شامل خط مربوط به خطای بوجود آمده است، برگردانده می شود	srcText
url مربوط به سند مستقر سده در حافظه ، برگردانده می شود .	url
محل بروز خطاء در فایل مربوطه، برگردانده می شود .	filePos

حرکت در طول درخت

یکی از متداولترین روش های بازیابی المان های یک سند XML ، حرکت در طول درخت (از گره ای به گره دیگر) و استخراج مقدار متن ذخیره شده به همراه هر یک از المان ها است . کدهای نوشته شده زیر ، امکان حرکت در طول یک درخت XML را فراهم و در ادامه هر یک از المان های سند XML ، در خروجی نمایش داده می شوند (کدها با استفاده از VBScript نوشته شده اند) .

حرکت در طول درخت و نمایش المان های یک سند XML

```

set xmlDoc=CreateObject("Microsoft.XMLDOM")
xmlDoc.async="false"
xmlDoc.load("note.xml")
for each x in xmlDoc.documentElement.childNodes
document.write(x.nodeName)
document.write(": ")
document.write(x.text)
next

```

ایجاد محتویات مبتنی بر Html برای یک سند XML

یکی از مهمترین قابلیت های XML ، تمایز و تفکیک سندهای Html از داده های مربوطه است . با استفاده از یک پارسر XML ، موجود در مرورگرها ، یک صفحه وب قادر به ایجاد محتویاتی پویا است . در این زمینه می توان از پتانسیل های جاوااسکریپت بمنظور تولید و ارائه محتویات پویا نیز استفاده کرد . در همین راستا ، امکان استفاده از ASP برای ایجاد محتویات پویا با تاکید بر نقش سرویس دهنده وب نیز وجود دارد . برنامه زیر ، داده های موجود در یک سند XML را خوانده و آنها را با فرمت Html در خروجی نمایش خواهد داد .

نمایش داده های موجود در یک سند XML با استفاده از جاوااسکریپت

```

var xmlDoc = new
ActiveXObject("Microsoft.XMLDOM")
xmlDoc.async="false"
xmlDoc.load("note.xml")
nodes = xmlDoc.documentElement.childNodes
to.innerText = nodes.item(0).text
from.innerText = nodes.item(1).text
header.innerText = nodes.item(2).text
body.innerText = nodes.item(3).text

```

دستیابی به المان ها از طریق نام

بهنام زیر داده های موجود در یک سند XML را خوانده و آنها را با فرمت Html در خروجی نمایش خواهد داد .

دستیابی به المان ها در یک سند XML از طریق نام و بکممک جاوااسکریپت

```
var xmlDoc = new ActiveXObject("Microsoft.XMLDOM")  
xmlDoc.async="false"  
xmlDoc.load("note.xml")  
document.write(xmlDoc.getElementsByTagName("from").item(0).text)
```