

Veritabanı Yönetim Sistemleri

1906003022015

Dr. Öğr. Üy. Önder EYECİOĞLU
Bilgisayar Mühendisliği

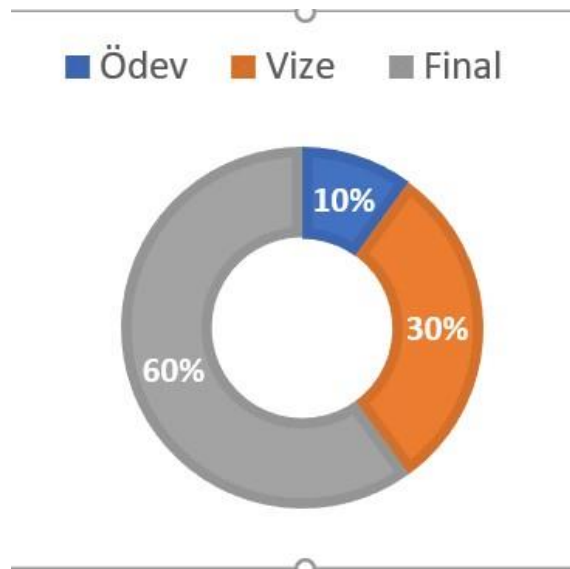


Giriş

Ders Günü ve Saati:

Pazartesi: 09:15-12:45

- Devam zorunluluğu %70
- Uygulamalar MS SQL ve MongoDB üzerinde gerçekleştirilecektir.



HAFTA	KONULAR
Hafta 1	VT ve VTYS'ne giriş
Hafta 2	ER Veri Modeli
Hafta 3	İlişkisel Modeller, İlişkisel model tasarımı
Hafta 4	İlişkisel Cebir ve Hesaplamalar
Hafta 5	İlişkisel Sorgular, SQL giriş
Hafta 6	SQL ile veri tabanı programlama
Hafta 7	SQL-Kısıtlar:Veri-tipi,birincil-anahtar,ikinci-anahtar,
Hafta 8	Vize
Hafta 9	İlişkisel Veri Tabanı Tasarımı ve Normalizasyon
Hafta 10	yarı-yapısal veri modelleri, XML
Hafta 11	JSON
Hafta 12	İlişkisel olmayan DB, NoSQL
Hafta 13	NoSQL
Hafta 14	DBMS -Eşzamanlılık (Concurrency) Kontrolü

İÇERİK

XML (eXtensible Markup Language)

- XML Nedir?
- XML Sözdizimi
- XML Element ve Özellikleri
- XML Hiyerarşisi
- XML isim uzayları
- XML Ayırıştırıcı
- XML DTS

JSON (JavaScript Object Notation)

- Genel Bakış
- JSON – Sözdizimi
- JSON - Veri Türleri
- JSON – Nesneler
- JSON – Şeması
- JSON - XML ile karşılaştırma

İÇERİK



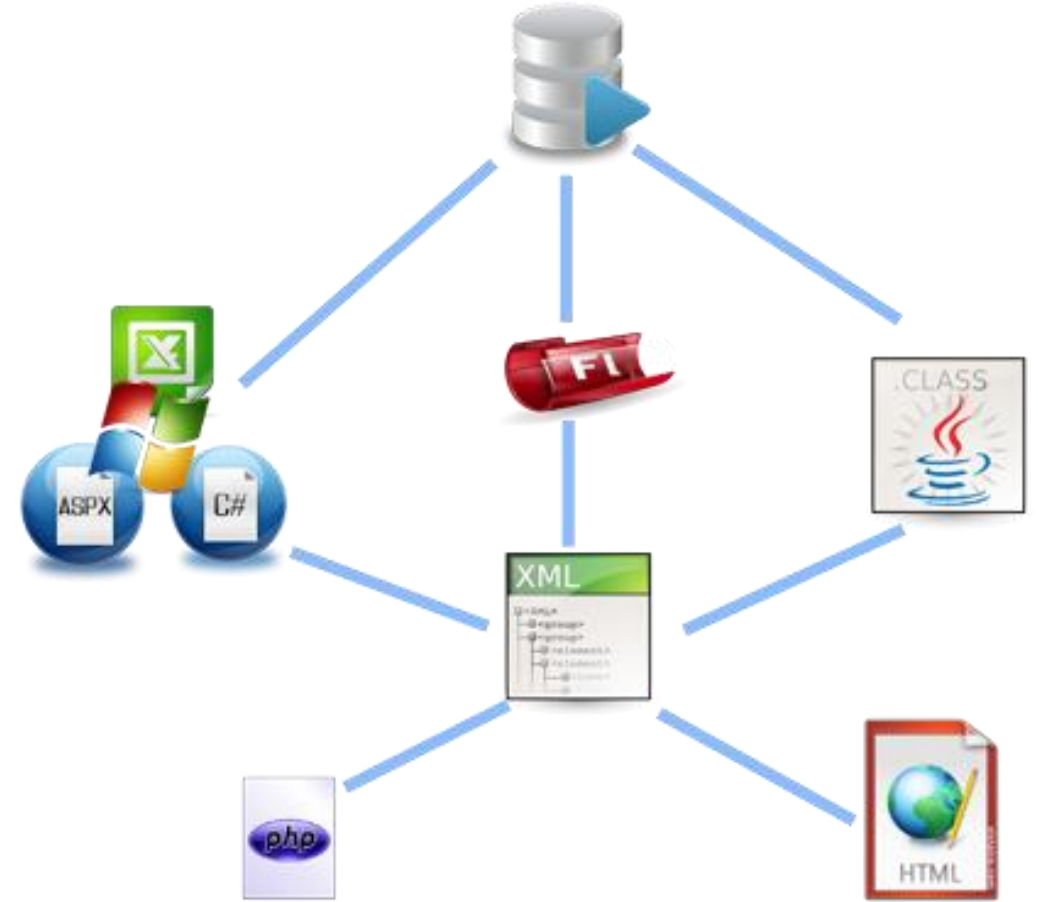
XML (eXtensible Markup Language)

XML Nedir?

- XML platformdan bağımsız veri taşıyabilen işaretleme dilidir.
- XML, eXtensible Markup Language sözcüğünün kısaltmasıdır.
- XML datanın tanımıyla ilgilidir.
- HTML gibi bir işaretleme(Markup) dilidir.
- Document Type Defination (DTD) veya XML Schema kullanarak datayı tanımlar.
- XML programsal anlamda hiçbir iş yapmaz.
- Basit bir text dosyasıdır.
- XML şu ana kadar bulunan en sağlam, güvenilir ve ASCII'den sonra en esnek doküman formatıdır

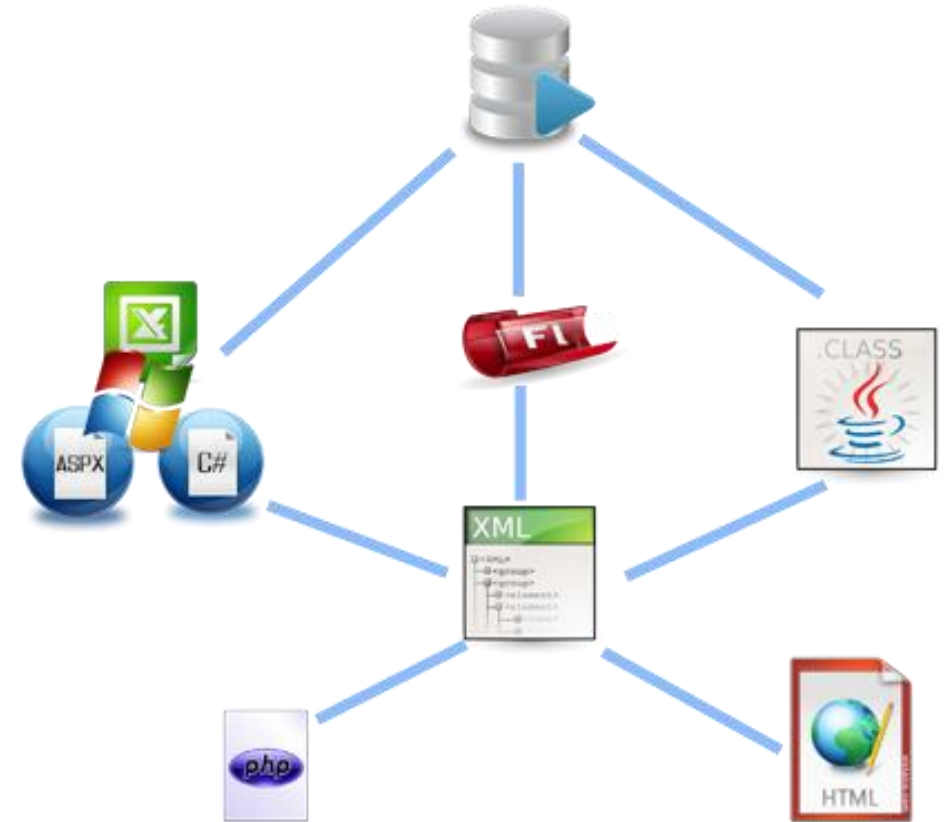
XML Nedir?

- Extensible Markup Language kelimesinin kısaltması olarak kullanılan XML, Bağımsız bir kuruluş olan W3C (World Wide Web Consortium) tarafından oluşturulan, her kullanıcının kendisine göre şekillendirip değiştirebildiği, bununla birlikte kurumsal olarak belirli bir kalıp çıkarılarak veri transferinde de yoğunlukla kullanılan, esnek genişletilebilir, platformdan bağımsız bir meta dildir.



XML Nedir?

- Sistemler verilerini veri tabanlarında barındırırlar ve her sistemin veritabanı mimarisi birbirinden farklıdır. Bu sistemler arası bilgi alışverişi, entegrasyon işlemleri yapılacağında veriler XML formatında transfer edilir. Bilgi alınacak veri tabanından veriler belirli bir standarda göre oluşturulur ve bu standartlara göre karşı taraf veriyi çekerek kendi sistemine alabilir. Böylece XML ile hatasız şekilde veri transferi gerçekleşmiş olur.



XML Nedir?

- XML, verileri depolamak ve taşımak için tasarlanmıştır.
- XML, hem insan hem de makine tarafından okunabilir olacak şekilde tasarlanmıştır.
- Temelde işlevi farklı programlama dilleri ve platformlar arasında veri aktarımını sağlayacak bir dil sunmak. Yani PHP, Python, Java ve C# gibi farklı dillerle XML dosyalarını işleyebilir ve bir programlama diliyle oluşturulan XML dosyasını bir başka programlama diliyle işleyebilirsiniz.
- XML, nasıl görüntüleneceğiyle ilgili herhangi bir bilgi taşımamaktadır.
- Aynı XML verileri birçok farklı sunum senaryosunda kullanılabilir.

XML kullanıcıya şunları sunar:

- Çoklu kütükleri birleştirerek bileşik belgeler yaratmak
- Metin kütüklerinde istenilen yere ve istenilen formatta resim eklemek
- Belge geçerleyiciler ve tarayıcılar gibi programlara bilgi denetim işlemini sağlamak
- XML sayesinde veri aktarımı yapılan her yerde, aktarılan veri, “meta-verisi ile birlikte” aktarılabilir.
- Meta-veri hakkındaki kurallar, yalnızca bir kez, ayrıca aktarılır.
- Bir çok sistem yazılımının ayar dosyaları XML dosyasıdır.
- Fakat unutmamak lazımdır ki XML’de şunlar bulunmaz:
 - HTML’deki gibi belgeleri işaretlemek için daha önceden tanımlanmış etiketler.
 - Belirli tiplerdeki belgeler üretmek için standart bir şablon

XML pozitif yanları

- XML hiyerarşik bir text yapısına sahiptir.
- Lisans sorunu yoktur, platform odaklı olmayan ve çok desteklenen bir dildir.
- XML insanların okuyabileceği bir dildir.
- XML dokümanları bir Onaylayıcı-Denetleyici (Validating Parser) tarafından onaylanır.
- XML internet tarayıcısı gibi basit araçlarda bile gösterilebilir.

XML ile HTML arasındaki farklar

- XML data tanımlamak ve taşımakla görevlidir.
- XML, HTML yerini almayacaktır. Ancak HTML'in eksik kaldığı noktaları tamamlar.
- HTML datanın görseliyle ilgilenir.
- XML tag'leri bağımsız türetilir.(!predefined)

XML Sözdizimi

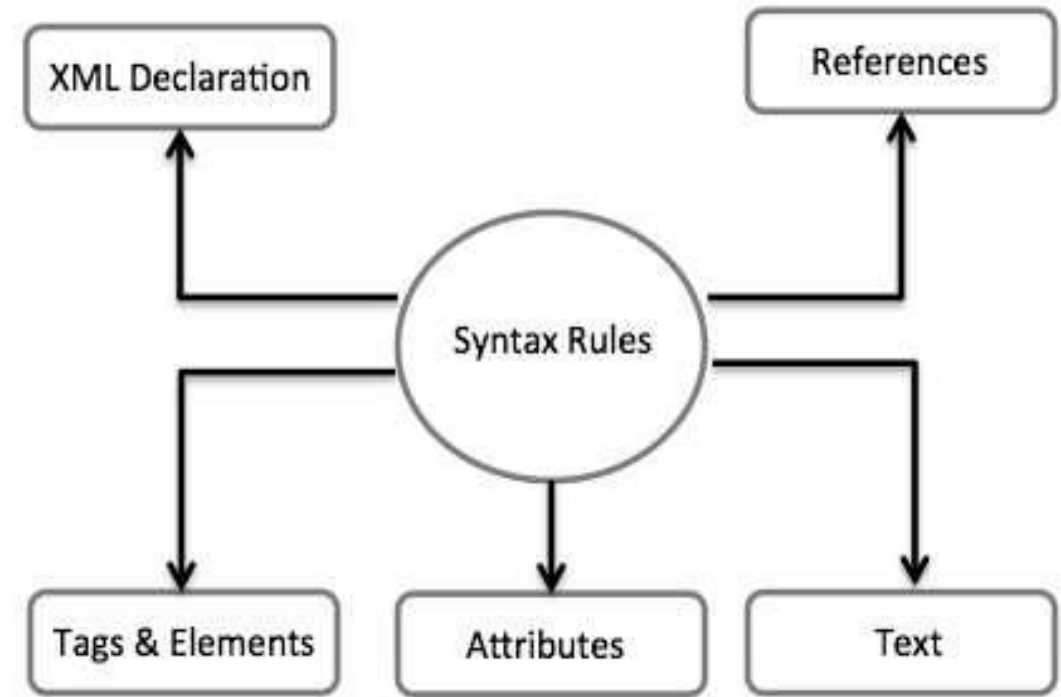
- XML ön tanımlı etiketler içermez.
- XML belgeleri, diğer tüm öğelerin ana ögesi olan bir kök ögesi içermelidir:

```
<root>  
  <child>  
    <subchild>.....</subchild>  
  </child>  
</root>
```

```
<?xml version="1.0"?>  
<contact-info>  
  <name>Tanmay Patil</name>  
  <company>TutorialsPoint</company>  
  <phone>(011) 123-4567</phone>  
</contact-info>
```

Document Prolog

Document Elements



XML Sözdizimi

XML declaration with no parameters:

```
<?xml >
```

XML declaration with version definition:

```
<?xml version="1.0">
```

XML declaration with all parameters defined:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
```

XML declaration with all parameters defined in single quotes:

```
<?xml version='1.0' encoding='iso-8859-1' standalone='no' ?>
```

XML Element ve Özellikleri

- XML ön tanımlı etiketler içermez.
- XML etiketleri XML'in temelini oluşturur. XML'de bir öğenin kapsamını tanımlarlar. Ayrıca, yorum eklemek, ortamı ayrıştırmak için gerekli ayarları bildirmek ve özel talimatlar eklemek için de kullanılabilirler.
- XML etiketleri büyük / küçük harfe duyarlıdır.
- XML etiketleri uygun bir sırayla kapatılmalıdır, yani dış eleman kapatılmadan önce başka bir elemanın içinde açılan bir XML etiketi kapatılmalıdır.

<price>29.99</price>

XML Element ve Özellikleri

```
<element-name attribute1 attribute2>  
....content  
</element-name>
```

XML Element ve Özellikleri

- XML öğelerinin HTML gibi özellikleri olabilir.
- Özellikler (Attributes), belirli bir öğeyle ilgili verileri içerecek şekilde tasarlanmıştır.

```
<gangster name="George &quot;Shotgun&quot; Ziegler">
```

Ham XML dosyaları tüm önemli tarayıcılarda görüntülenebilir.

XML dosyalarının HTML sayfaları olarak görüntülenmesini beklemeyin.

```
<note date="2008-01-10">  
  <to>Tove</to>  
  <from>Jani</from>  
</note>
```

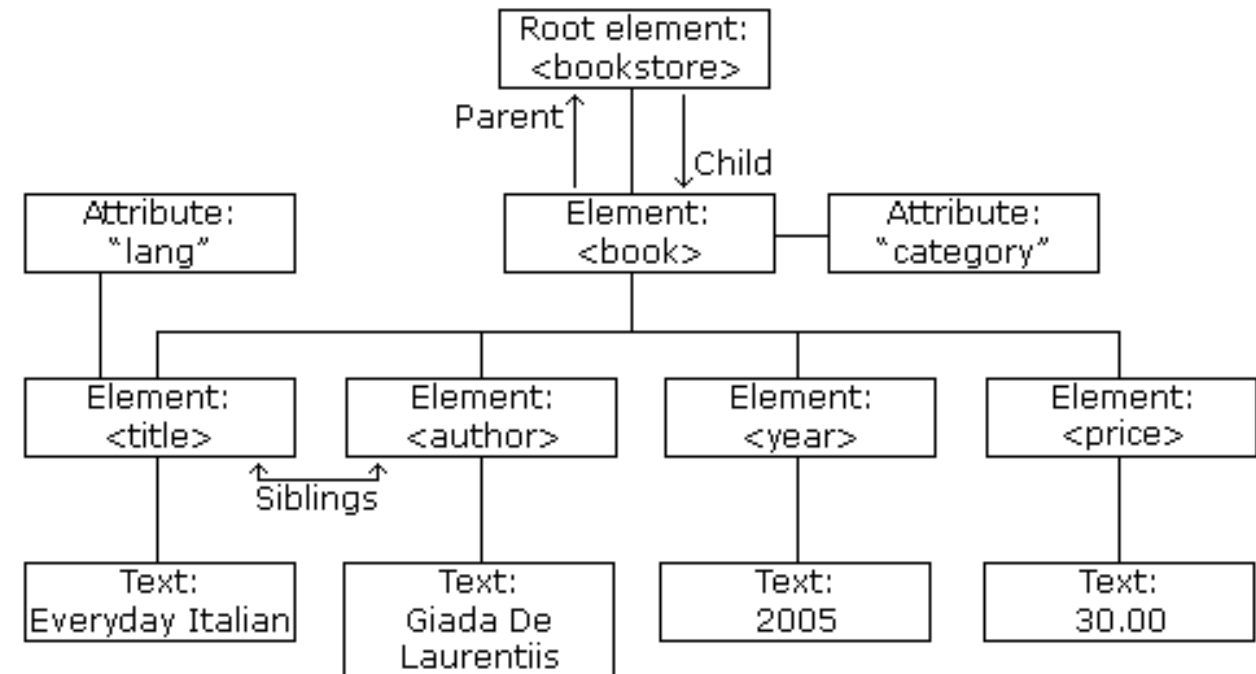
```
<note>  
  <date>2008-01-10</date>  
  <to>Tove</to>  
  <from>Jani</from>  
</note>
```


XML Karakterler

Entity name	Character	Decimal reference	Hexadecimal reference
quot	"	"	"
amp	&	&	&
apos	'	'	'
lt	<	<	<
gt	>	>	>

XML Hiyerarşisi

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book category="cooking">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
  <book category="children">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
  <book category="web">
    <title lang="en">Learning XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>
```



XML -CSS

CSS (Basamaklı Stil Sayfaları), bir XML belgesine stil eklemek ve bilgileri görüntülemek için kullanılabilir. Tüm XML belgesini biçimlendirebilir.

```
<? xml-stylesheet type = "text/css" href = "cssemployee.css" ?>
```

XML Teknolojileri

XML' in tanımlanmasını, biçimlenmesini, dönüştürülmesini ve içeriğinin okunmasını sağlayan teknolojiler

- **DOM**, Doküman Obje Modeli:
- **SAX**, XML için Basit Arayüz:
- **XML Namespaces**, XML Ad alanı dili
- **XML Schema** (.xsd),
- **DTD**, Doküman Tip Tanımı
- **XSL**, XML stil sayfası dili
- **XSLT**, XML çeviriler için stil sayfası dili

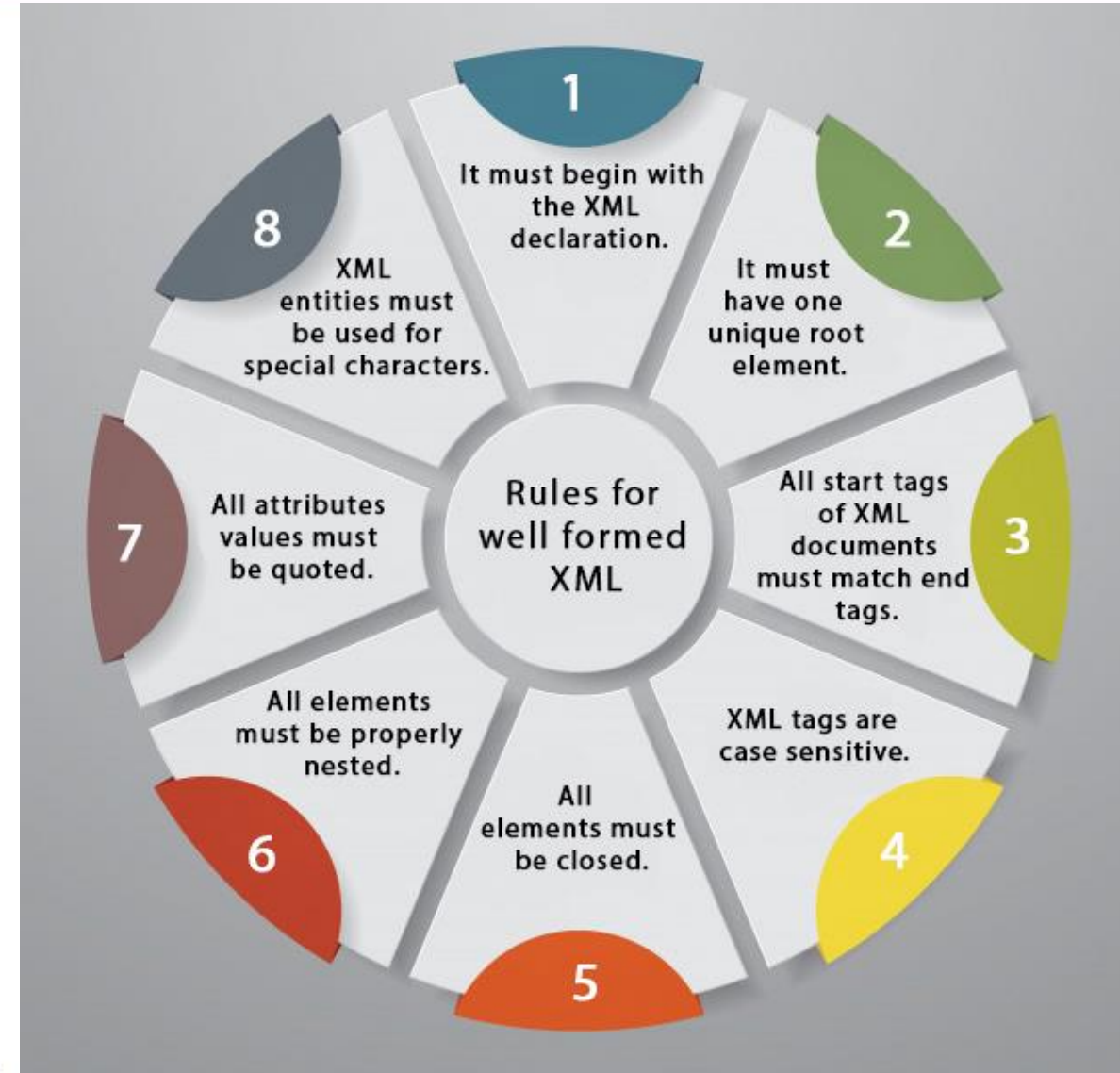
XML Teknolojileri

#.	teknoloji	Anlam	Tanım
1)	XHTML	Genişletilebilir html	XML'in daha net ve daha katı bir sürümüdür. XML biçimlendirme dilleri ailesine aittir. Html'yi daha genişletilebilir hale getirmek ve diğer verilerle birlikte çalışabilirliği artırmak için geliştirildi.
2)	XML DOM	XML belge nesne modeli	XML'e erişmek ve bunları işlemek için kullanılan standart bir belge modelidir. XML dosyasını ağaç yapısında tanımlar.
3)	XSL üç bölümden oluşur: i) XSLT (xsl dönüşümü) ii) XSL iii) XPath	Genişletilebilir stil sayfası dili	i) XML'i html gibi diğer biçimlere dönüştürür. ii) XML'i ekrana, kağıda vb. biçimlendirmek için kullanılır. iii) XML belgelerinde gezinmek için bir dildir.
4)	XQuery	XML sorgu dili	XML tabanlı verileri sorgulamak için kullanılan XML tabanlı bir dildir.
5)	DTD	Belge türü tanımı	Bir XML belgesindeki yasal unsurları tanımlamak için kullanılan bir standarttır.
6)	XSD	XML şema tanımı	dtd'ye XML tabanlı bir alternatiftir. Bir XML belgesinin yapısını tanımlamak için kullanılır.

XML Doğrulayıcı (Validator)

Doğru sözdizimine sahip bir XML belgesi "İyi Oluşturulmuş" olarak adlandırılır.

- XML belgelerinin bir kök ögesi olması gerekir
- XML ögeleri bir kapanış etiketine sahip olmalıdır
- XML etiketleri büyük / küçük harfe duyarlıdır
- XML ögeleri uygun şekilde iç içe olmalı
- XML özellik değerleri alıntılanmalıdır



XML Doğrulayıcı (Validator)

Geçerli XML Belgeleri

"İyi biçimlendirilmiş" bir XML belgesi, "geçerli" bir XML belgesiyle aynı şey değildir.

"Geçerli" bir XML belgesi iyi biçimlendirilmiş olmalıdır. Ayrıca, bir belge türü tanımına uygun olmalıdır.

XML ile kullanılabilecek iki farklı belge türü tanımı vardır:

- DTD - Orijinal Belge Türü Tanımı
- XML Şeması - DTD'ye XML tabanlı bir alternatif

Belge türü tanımı, bir XML belgesi için kuralları ve yasal öğeleri ve öznitelikleri tanımlar.

```
<!ELEMENT note (to,from,heading,body)>  
<!ELEMENT to (#PCDATA)>  
<!ELEMENT from (#PCDATA)>  
<!ELEMENT heading (#PCDATA)>  
<!ELEMENT body (#PCDATA)>
```

XML DTD (Document Type Descriptor)

- DTD, bir XML belgesinin yapısını ve yasal öğelerini ve özniteliklerini tanımlar.
- Doğru sözdizimine sahip bir XML belgesi "İyi Oluşturulmuş" olarak adlandırılır.
- DTD'ye karşı doğrulanmış (validation) bir XML belgesi "İyi Oluşturulmuş" ve "Geçerli" dir.
- DTD dosyası dokümanda kullanılmasına izin verilen ve verilmeyen tüm nesneleri belirtir, bunların kullanılacağı yerleri belirtir.
- DTD direkt olarak veya bir URI pointer yardımıyla dıştan eklenerek kullanılabilir.
- DTD kullanımı zorunlu değildir.
- DTD, XML'de hangi tiplerin kullanılacağını belirtir. Detaylı bir formatlama sağlamaz (örneğin tarih bilgisinin belirli bir aralıkta olmasını sağlamaya yetmez).
- DTD dosyası Doctype bildirisi ile dışsal bir dosya olarak tanımlanmak yerine [] işaretleri içinde içsel olarak yazılabilir.

XML DTD (Document Type Descriptor)

DTD açısından bakıldığında, tüm XML belgeleri aşağıdaki yapı taşlarından oluşur:

- Elements
- Attributes
- Entities
- PCDATA
- CDATA

XML DTD (Document Type Descriptor)

```
<!ELEMENT element-name category>
```

or

```
<!ELEMENT element-name (element-content)>
```

```
<!ATTLIST element-name attribute-name attribute-type attribute-value>
```

DTD example:

```
<!ATTLIST payment type CDATA "check">
```

XML example:

```
<payment type="check" />
```

```
<!ENTITY entity-name "entity-value">
```

XML DTD (Document Type Descriptor)

PCDATA

PCDATA, ayrıştırılmış karakter verileri anlamına gelir.

Karakter verilerini, bir XML öğesinin başlangıç etiketi ile bitiş etiketi arasında bulunan metin olarak düşünün.

PCDATA, bir ayrıştırıcı tarafından ayrıştırılacak olan metindir . Metin, varlıklar ve işaretleme için ayrıştırıcı tarafından incelenecektir .

Metin içindeki etiketler, işaretleme olarak değerlendirilecek ve varlıklar genişletilecektir.

Ancak, ayrıştırılmış karakter verileri &, < veya > karakterleri içermemelidir.(&, < >)

XML DTD (Document Type Descriptor)

CDATA

CDATA, karakter verileri anlamına gelir.

CDATA, bir ayrıştırıcı tarafından ayrıştırılmayacak olan metindir . Metin içindeki etiketler işaretleme olarak DEĞERLENDİRİLMEMEYECİK ve varlıklar genişletilmeyecektir.

XML DTD (Document Type Descriptor)

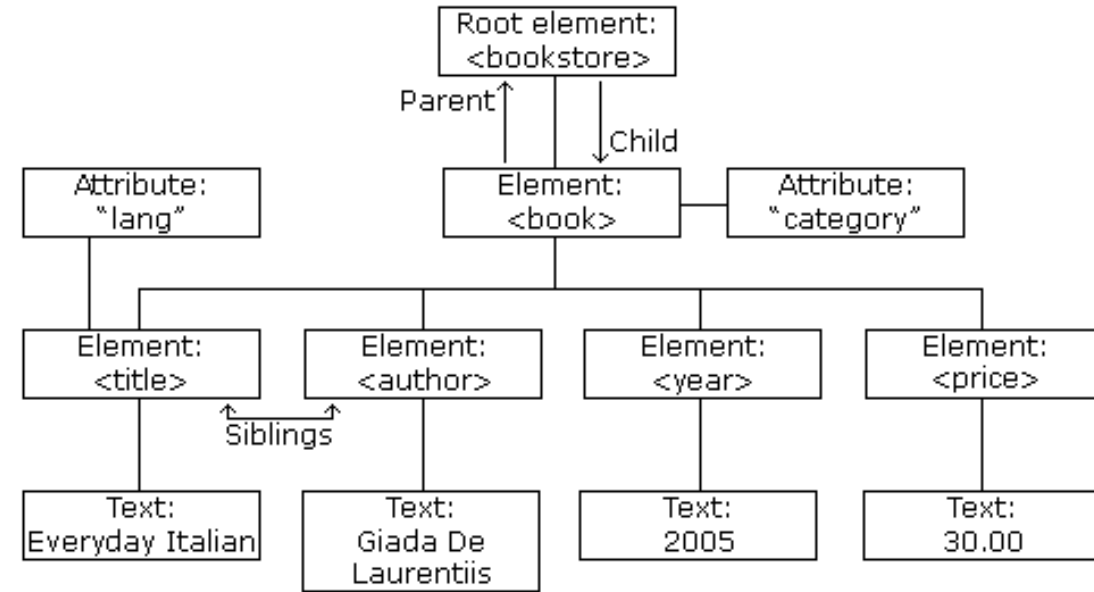
```
<!DOCTYPE SEMINER [  
  <!ELEMENT SUNUM (TANISMA, SLAYT+, SORU+)>  
  <!ELEMENT TANISMA (#PCDATA)>  
  <!ELEMENT SLAYT (#PCDATA)>  
  <!ELEMENT SORU (#PCDATA)>  
  <!ATTLIST SUNUM SUNAN CDATA #REQUIRED>  
  <!ATTLIST SUNUM TARİH CDATA #IMPLIED>  
  <!ENTITY SUNUM "XML ve Web Servisleri">  
>
```

XMI Schema

```
<?xml version = "1.0" encoding = "UTF-8"?>
<xs:schema xmlns:xs = "http://www.w3.org/2001/XMLSchema">
  <xs:element name = "contact">
    <xs:complexType>
      <xs:sequence>
        <xs:element name = "name" type = "xs:string" />
        <xs:element name = "company" type = "xs:string" />
        <xs:element name = "phone" type = "xs:int" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

XML PARSER-DOM

DOM ; **D** ocument **O** bject **M** odel kelimelerinin baş harflerinden oluşmaktadır. DOM ile bir XML belgesinin içeriğini, stilini ve yapısını görüp değiştirebiliriz . DOM ayrıştırıcısı kullanıldığı zaman XML belgesi kök elemanından başlanarak hafızada bir ağaca yerleştirilir. Bu ayrıştırıcı XML belgesini öncelikle baştan sona okur ve tüm bileşenler mantıksal düğümler modeli olarak hafızadaki yerini alır. Bu aşamadan sonra içerdiği metotlar ile bu düğümler arasında ileri ve geri hareket edebilirsiniz. Yine aynı şekilde DOM metotlarını kullanarak bu düğümlerdeki değerleri değiştirebilirsiniz. DOM' da en önemli noktalardan biri de etiklerde yer alan özellikler bir alt düğüm olarak ağaca yerleştirilmezler ve o düğümün özelliği olarak ele alınırlar.



XML PARSER-DOM

- DOM, belgelere erişmek ve bunları yönetmek için bir standart tanımlar:
- HTML DOM, HTML belgelerine erişmek ve bunları yönetmek için standart bir yol tanımlar. Bir ağaç yapısını bir HTML belgesi olarak sunar.

```
<h1 id="demo">This is a Heading</h1>  
<button type="button"  
onclick="document.getElementById('demo').innerHTML = 'Hello World!'">Click Me!  
</button>
```

- XML DOM, XML belgelerine erişmek ve bunları yönetmek için standart bir yol tanımlar. Bir XML dokümanı bir ağaç yapısı olarak sunar.

XML PARSER-DOM

- `<p id="demo"></p>`

```
<script>
```

```
var text, parser, xmlDoc;
```

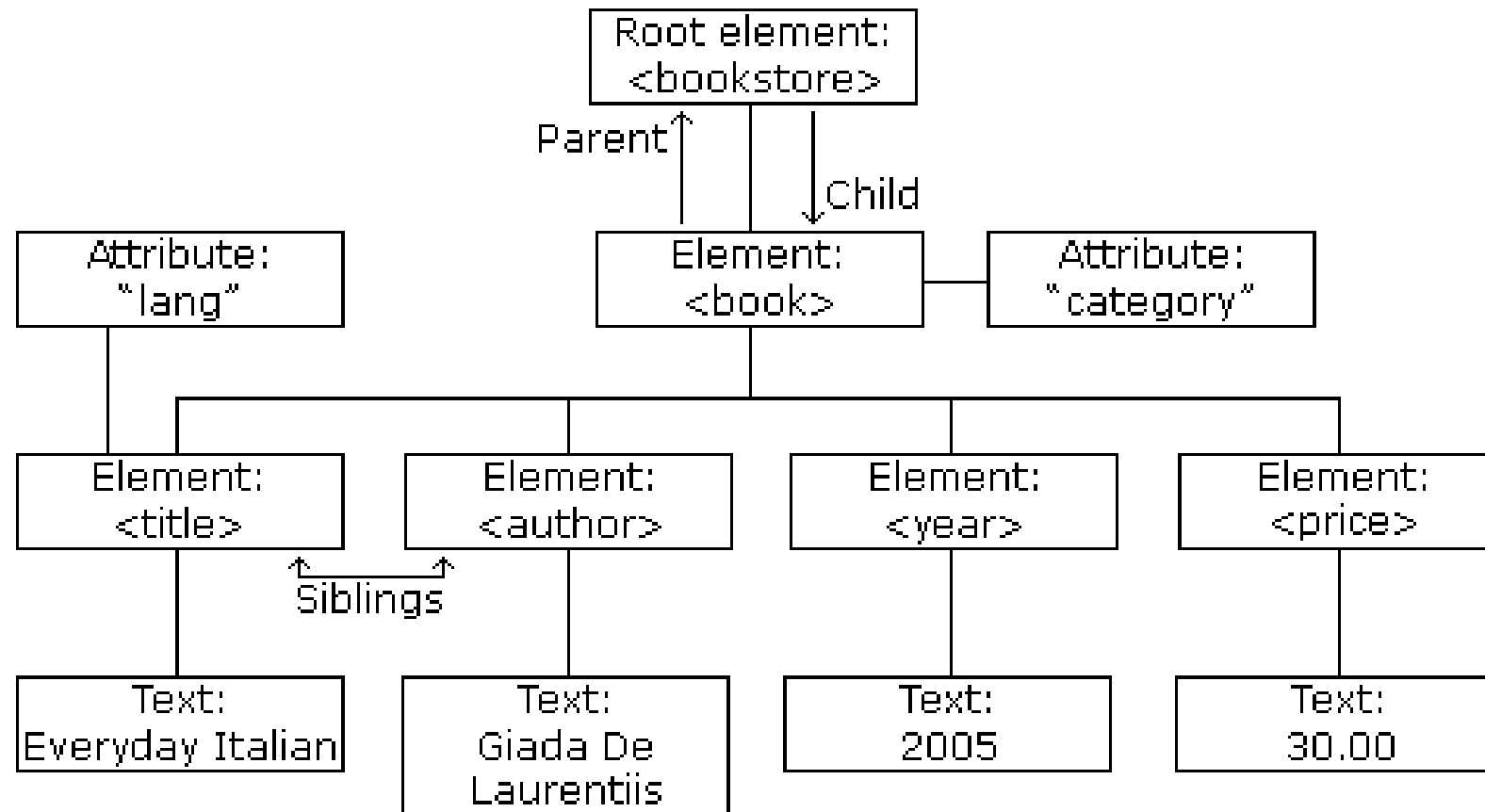
```
text = "<bookstore><book>" +  
"<title>Everyday Italian</title>" +  
"<author>Giada De Laurentiis</author>" +  
"<year>2005</year>" +  
"</book></bookstore>";
```

```
parser = new DOMParser();
```

```
xmlDoc = parser.parseFromString(text, "text/xml");
```

```
document.getElementById("demo").innerHTML =  
xmlDoc.getElementsByTagName("title")[0].childNodes[0].nodeValue;  
</script>
```

XML PARSER-DOM



XML PARSER-DOM

```
var xhttp = new XMLHttpRequest();
xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status ==
200) {
        // Action to be performed when the
document is read;
    }
};
xhttp.open("GET", "filename", true);
xhttp.send();
```

XML PARSER-DOM

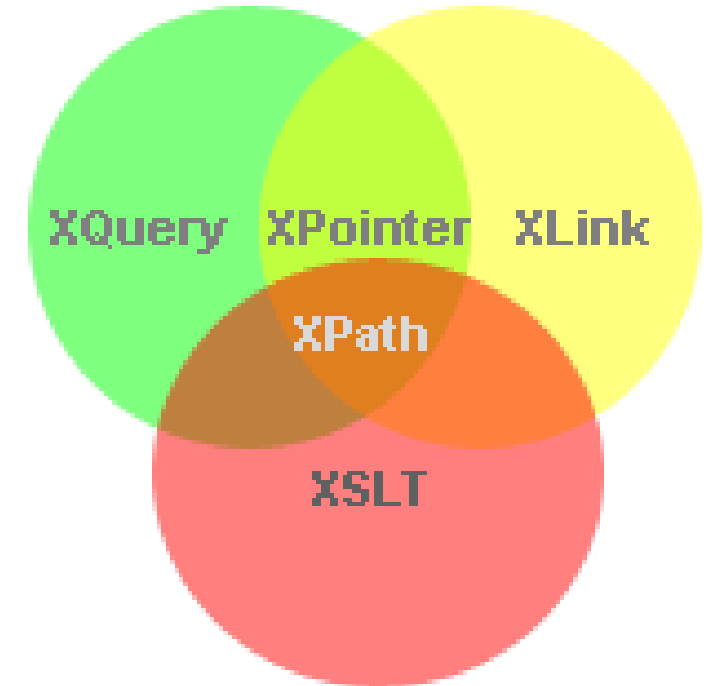
XMLHttpRequest Object Properties

Property	Description
onreadystatechange	Stores a function (or the name of a function) to be called automatically each time the readyState property changes
readyState	Holds the status of the XMLHttpRequest. Changes from 0 to 4: 0: request not initialized 1: server connection established 2: request received 3: processing request 4: request finished and response is ready
responseText	Returns the response data as a string
responseXML	Returns the response data as XML data
status	Returns the status-number (e.g. "404" for "Not Found" or "200" for "OK")
statusText	Returns the status-text (e.g. "Not Found" or "OK")



XML and XPath

- XPath, bir XML belgesinin bölümlerini tanımlamak için kullanılan bir sözdizimidir.
- XPath, XML belgelerinde gezinmek için yol ifadelerini kullanır
- XPath, standart işlevlerden oluşan bir kitaplık içerir
- XPath, XSLT ve XQuery'de önemli bir unsurdur



XML and XPath

XPath Expression	Result
/bookstore/book[1]	Selects the first book element that is the child of the bookstore element
/bookstore/book[last()]	Selects the last book element that is the child of the bookstore element
/bookstore/book[last()-1]	Selects the last but one book element that is the child of the bookstore element
/bookstore/book[position() < 3]	Selects the first two book elements that are children of the bookstore element
//title[@lang]	Selects all the title elements that have an attribute named lang
//title[@lang='en']	Selects all the title elements that have a "lang" attribute with a value of "en"
/bookstore/book[price>35.00]	Selects all the book elements of the bookstore element that have a price element with a value greater than 35.00
/bookstore/book[price>35.00]/title	Selects all the title elements of the book elements of the bookstore element that have a price element with a value greater than 35.00

