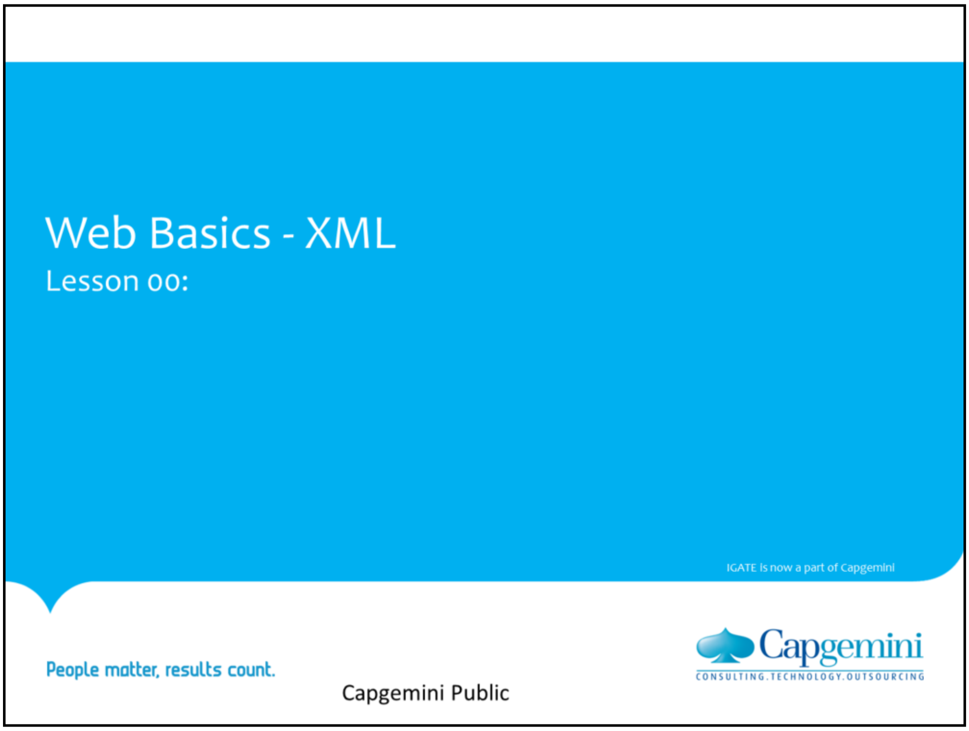


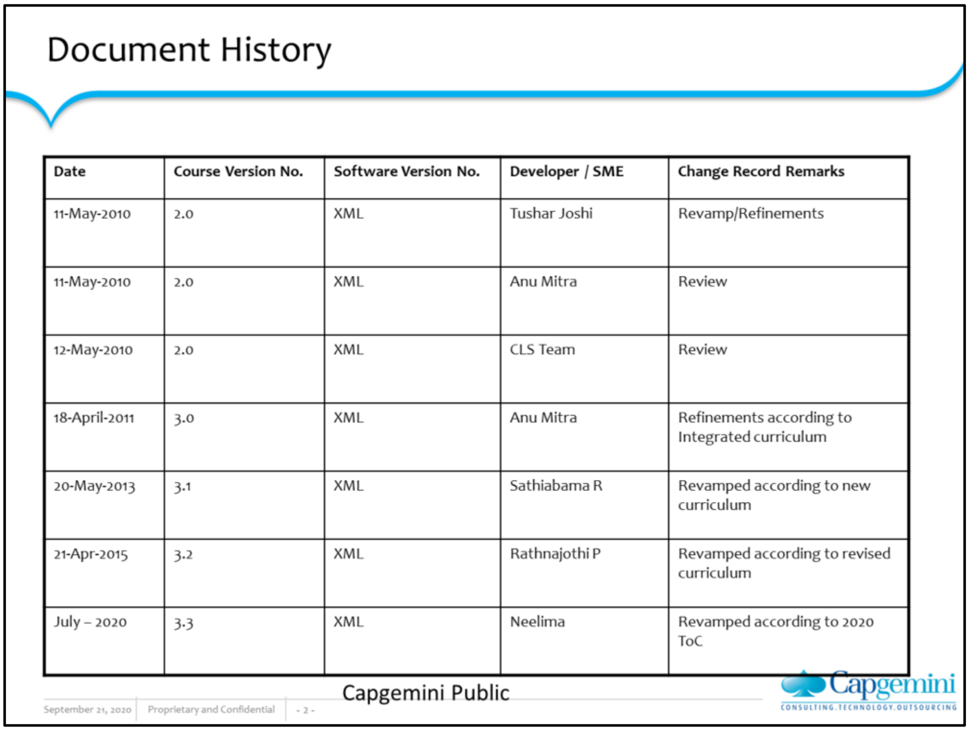
Web Basics - XML

Lesson 00:

IGATE is now a part of Capgemini

People matter, results count.

Capgemini Public

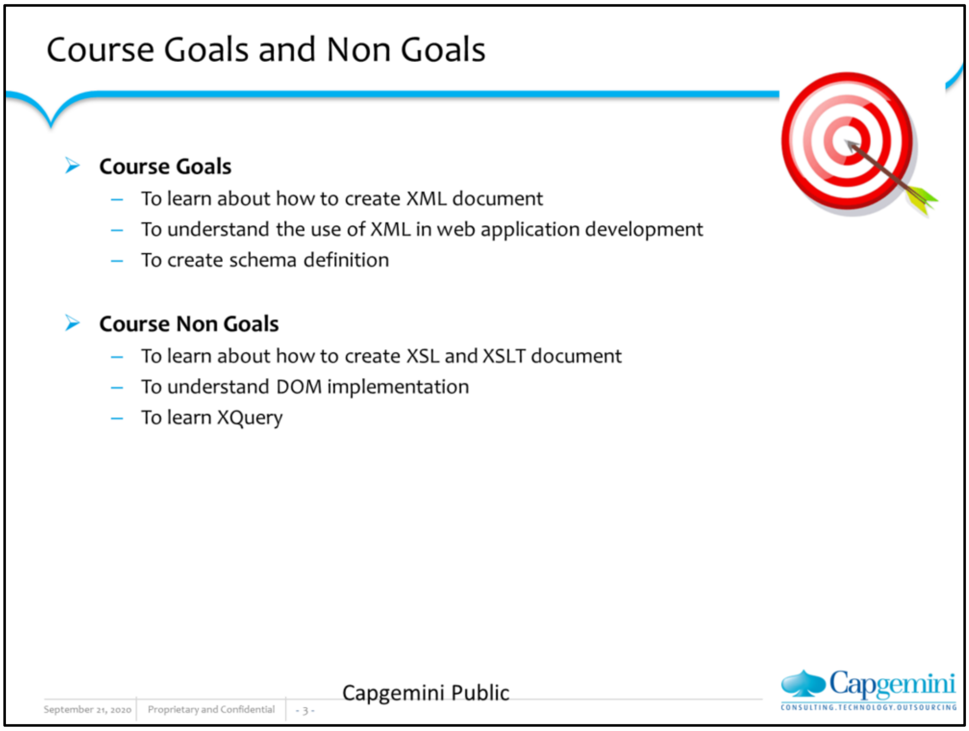


Document History

Date	Course Version No.	Software Version No.	Developer / SME	Change Record Remarks
11-May-2010	2.0	XML	Tushar Joshi	Revamp/Refinements
11-May-2010	2.0	XML	Anu Mitra	Review
12-May-2010	2.0	XML	CLS Team	Review
18-April-2011	3.0	XML	Anu Mitra	Refinements according to Integrated curriculum
20-May-2013	3.1	XML	Sathiabama R	Revamped according to new curriculum
21-Apr-2015	3.2	XML	Rathnajothi P	Revamped according to revised curriculum
July – 2020	3.3	XML	Neelima	Revamped according to 2020 ToC

Capgemini Public

September 21, 2020 | Proprietary and Confidential | - 2 -



Course Goals and Non Goals

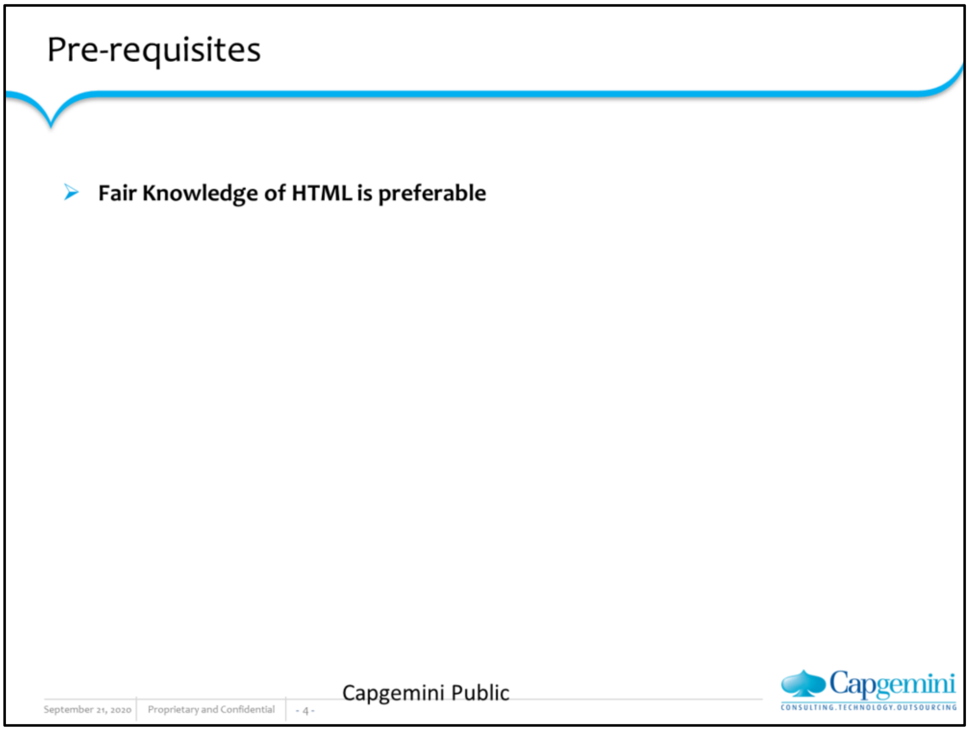
- **Course Goals**
  - To learn about how to create XML document
  - To understand the use of XML in web application development
  - To create schema definition

- **Course Non Goals**
  - To learn about how to create XSL and XSLT document
  - To understand DOM implementation
  - To learn XQuery

September 21, 2020 | Proprietary and Confidential | - 3 -

Capgemini Public

Note:
Goals: Participants should be able to know how to create XML document, understand the use of XML in web application development, and creating schema definition.
Non-Goals: Participants will not learn about creation of XSL and XSLT file. DOM implementation is beyond the scope of this course. XQuery is not in scope of this course.

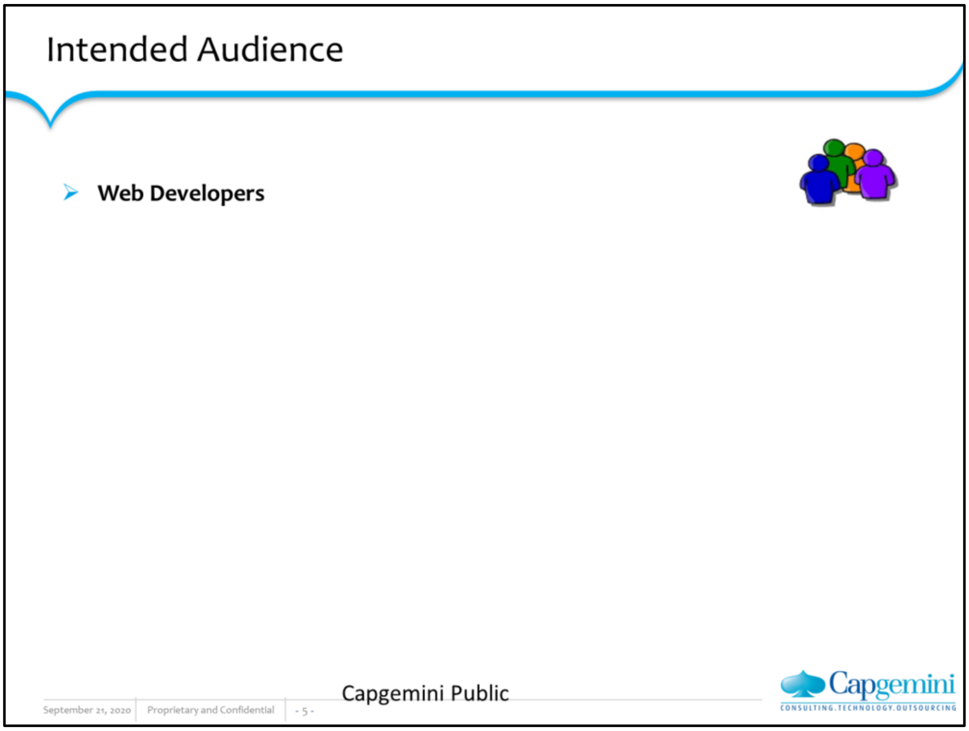


Pre-requisites

- **Fair Knowledge of HTML is preferable**

Capgemini Public

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING



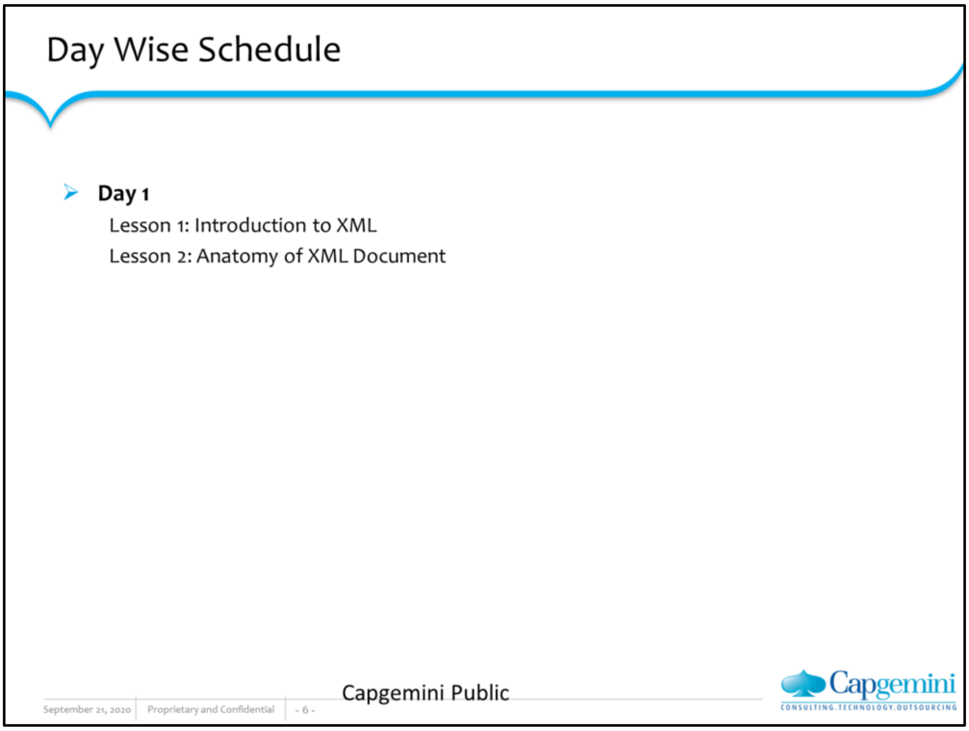
Intended Audience

- Web Developers

September 21, 2020 | Proprietary and Confidential | - 5 -

Capgemini Public

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING



Day Wise Schedule

- **Day 1**

  Lesson 1: Introduction to XML

  Lesson 2: Anatomy of XML Document

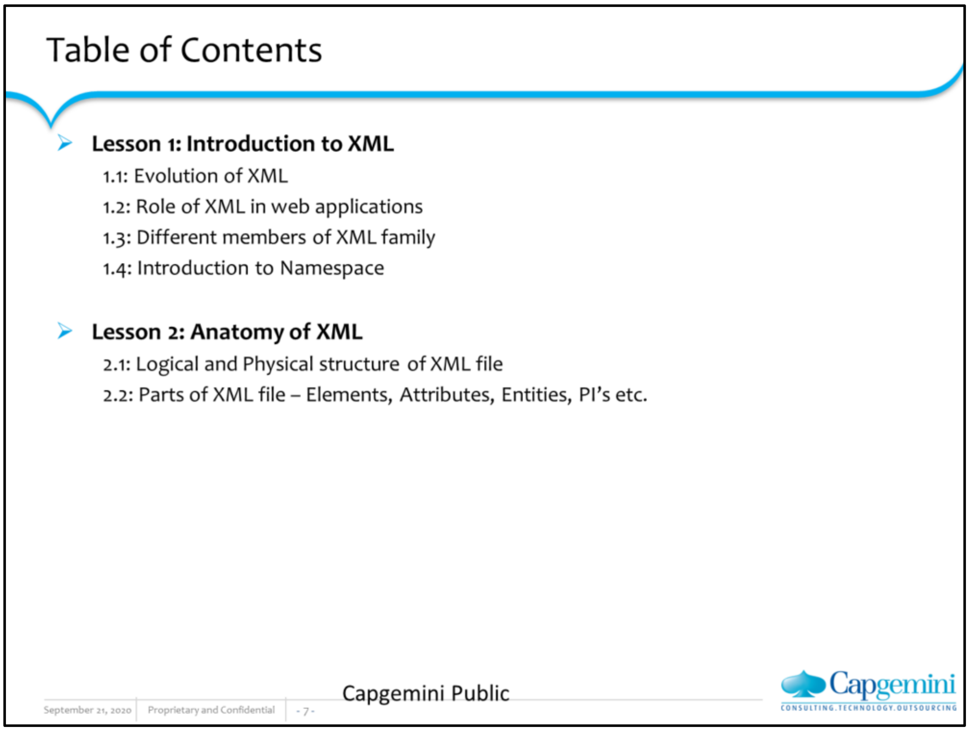
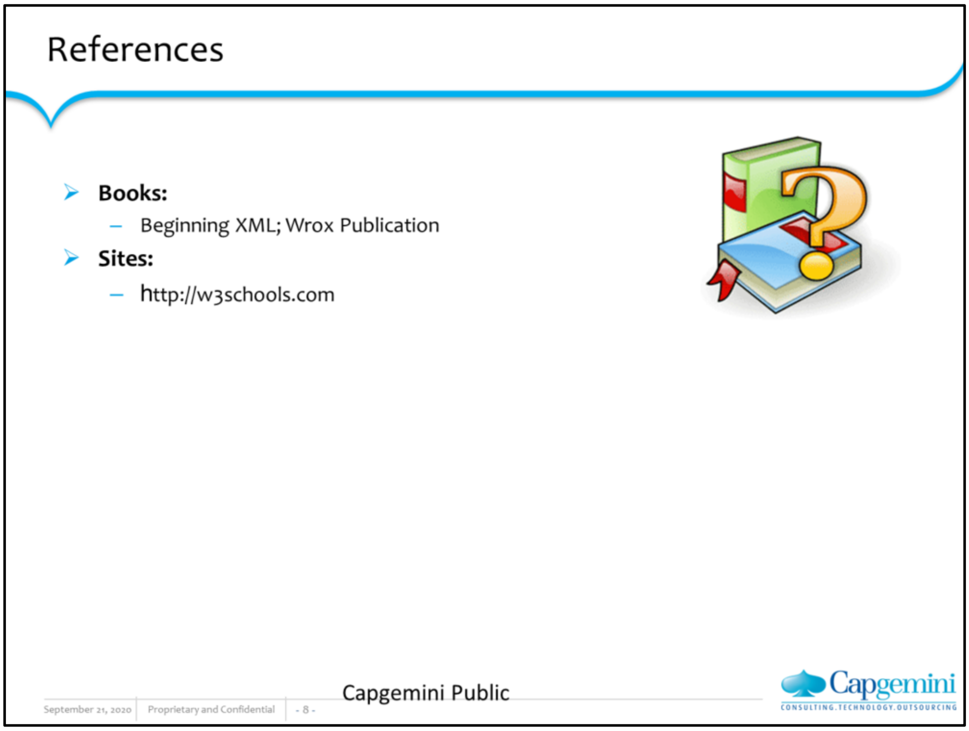


Table of Contents

- **Lesson 1: Introduction to XML**
  - 1.1: Evolution of XML
  - 1.2: Role of XML in web applications
  - 1.3: Different members of XML family
  - 1.4: Introduction to Namespace
- **Lesson 2: Anatomy of XML**
  - 2.1: Logical and Physical structure of XML file
  - 2.2: Parts of XML file – Elements, Attributes, Entities, PI's etc.

Capgemini Public

September 21, 2020 | Proprietary and Confidential | - 7 -

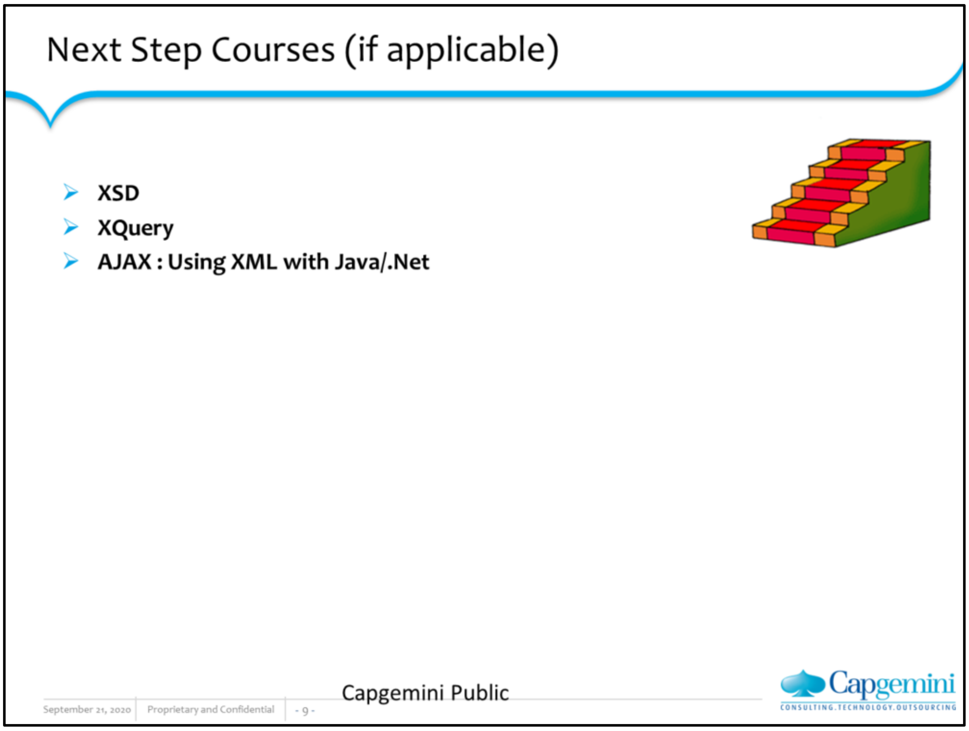


References

- **Books:**
  - Beginning XML; Wrox Publication
- **Sites:**
  - http://w3schools.com

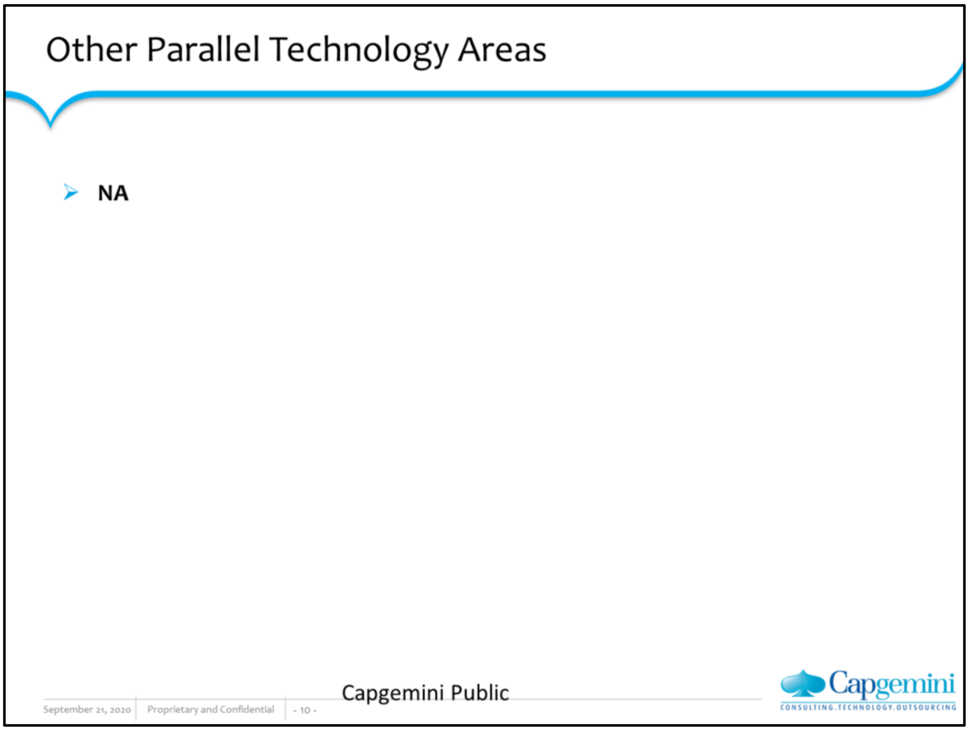
September 21, 2020 | Proprietary and Confidential | - 8 -

Capgemini Public



Next Step Courses (if applicable)

- XSD
- XQuery
- AJAX : Using XML with Java/.Net

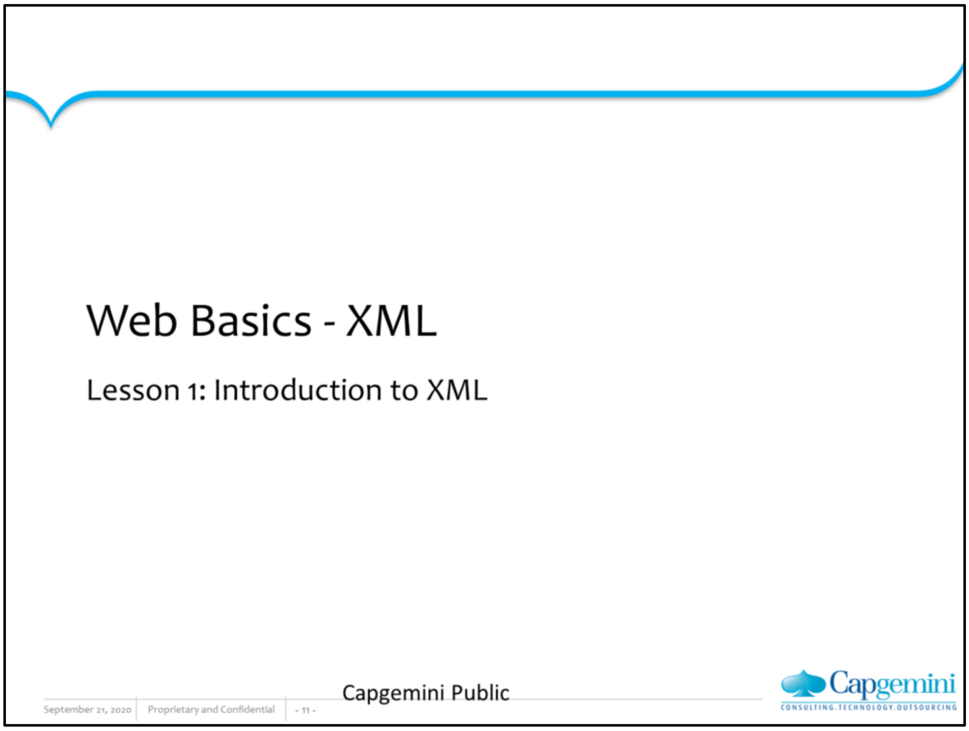


Other Parallel Technology Areas

- NA

September 21, 2020 | Proprietary and Confidential | - 10 -

Capgemini Public

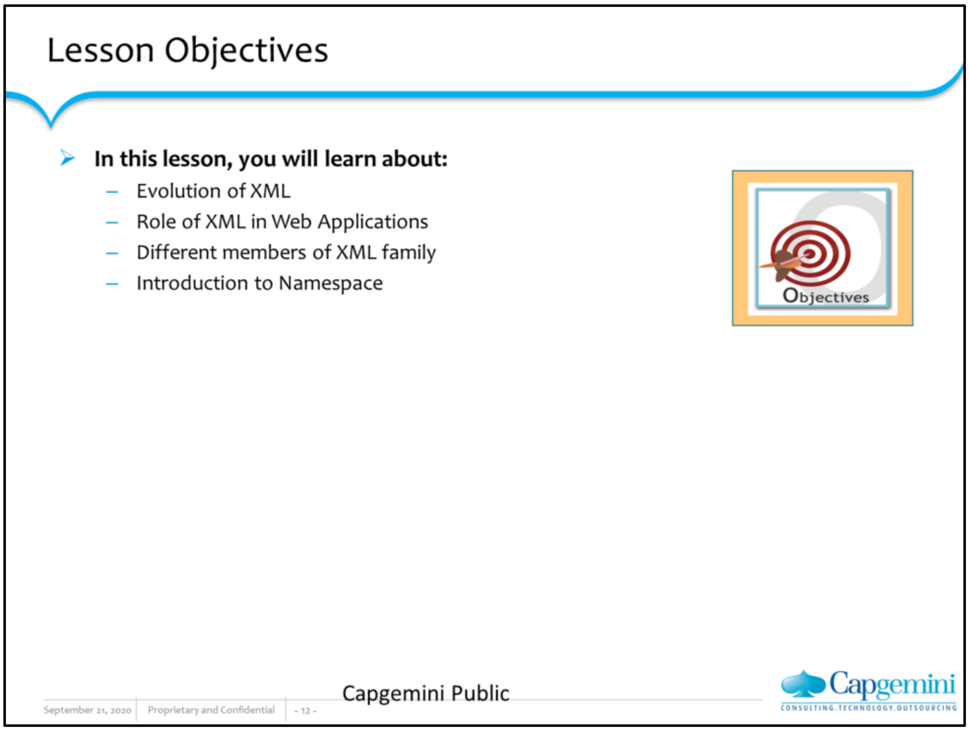


Web Basics - XML

Lesson 1: Introduction to XML

September 21, 2020 | Proprietary and Confidential | - 11 -

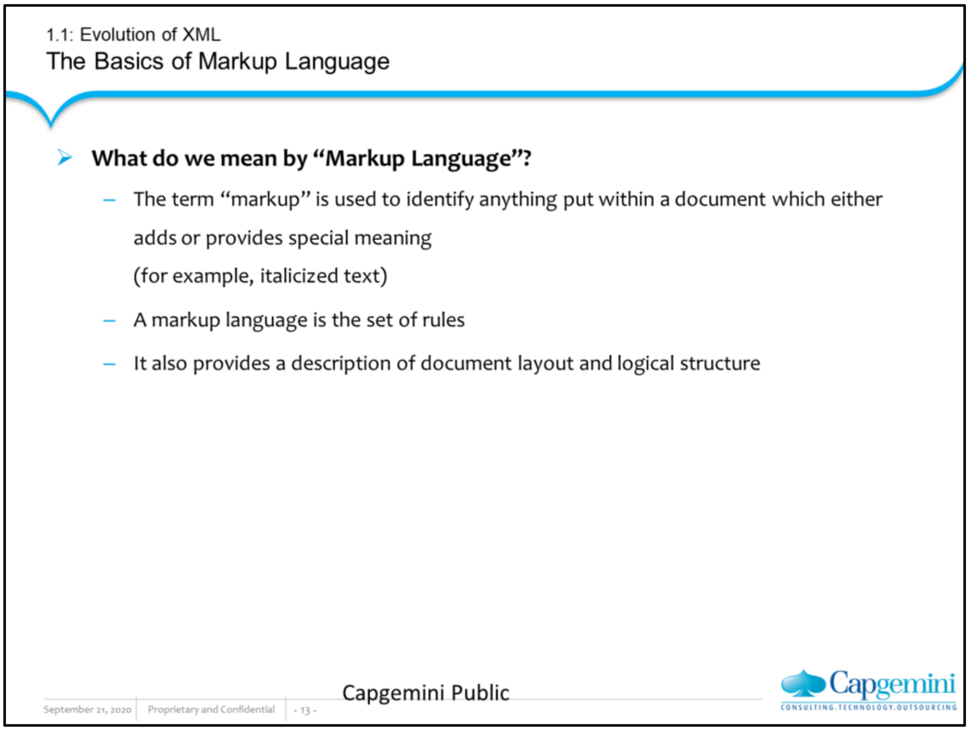
Capgemini Public



Objectives

Lesson Objectives

- **In this lesson, you will learn about:**
  - Evolution of XML
  - Role of XML in Web Applications
  - Different members of XML family
  - Introduction to Namespace



1.1: Evolution of XML

The Basics of Markup Language

- **What do we mean by "Markup Language"?**
  - The term "markup" is used to identify anything put within a document which either adds or provides special meaning
    (for example, italicized text)
  - A markup language is the set of rules
  - It also provides a description of document layout and logical structure

September 21, 2020 | Proprietary and Confidential | - 13 -

Capgemini Public

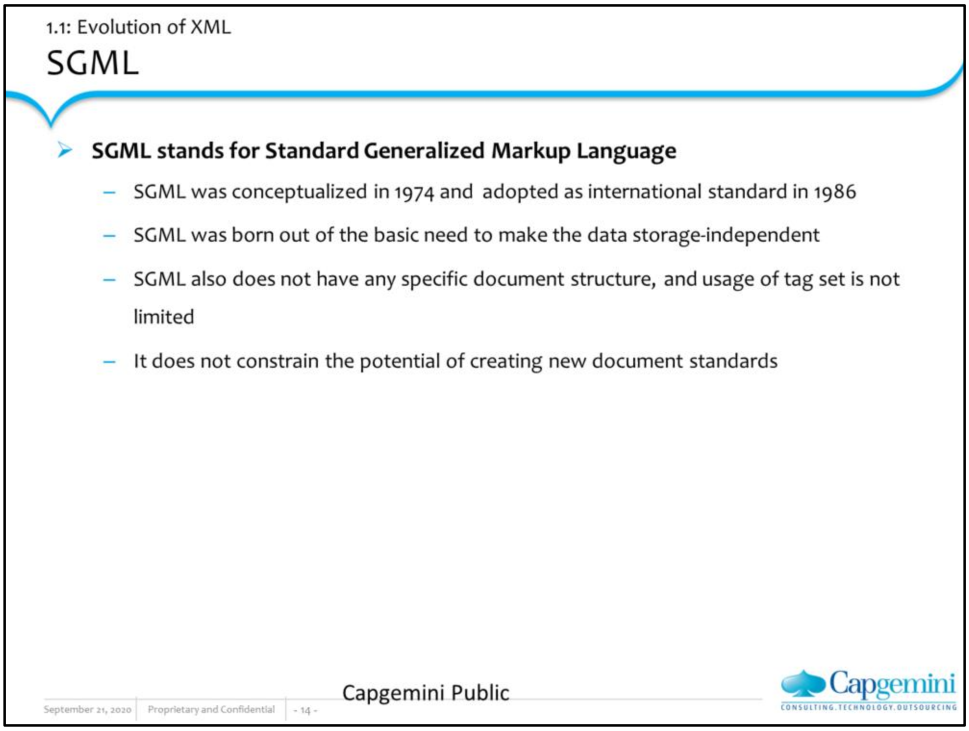
Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

The Basics of Markup Language:

There exist three types of markup:

- Stylistic: It determines how a document is presented (for example, the HTML tags <I> for italics, <B> for bold, and <U> for underline).
- Structural: It determines how the document is to be structured (for example, the HTML tags <P> for paragraph, <SPAN> for creating ad hoc styles in a document, and <DIV> for grouping structures that are aligned in the same way.
- Semantic: It tells about the content of the data (for example, the HTML tags <TITLE> for page title, <HEAD> for page header information, and <SCRIPT>to indicate a JavaScript in a page.)

In XML, the only type of markup that we are concerned with is structural.



1.1: Evolution of XML

SGML

- **SGML stands for Standard Generalized Markup Language**
  - SGML was conceptualized in 1974 and adopted as international standard in 1986
  - SGML was born out of the basic need to make the data storage-independent
  - SGML also does not have any specific document structure, and usage of tag set is not limited
  - It does not constrain the potential of creating new document standards

September 21, 2020 | Proprietary and Confidential | - 14 -

Capgemini Public

SGML:

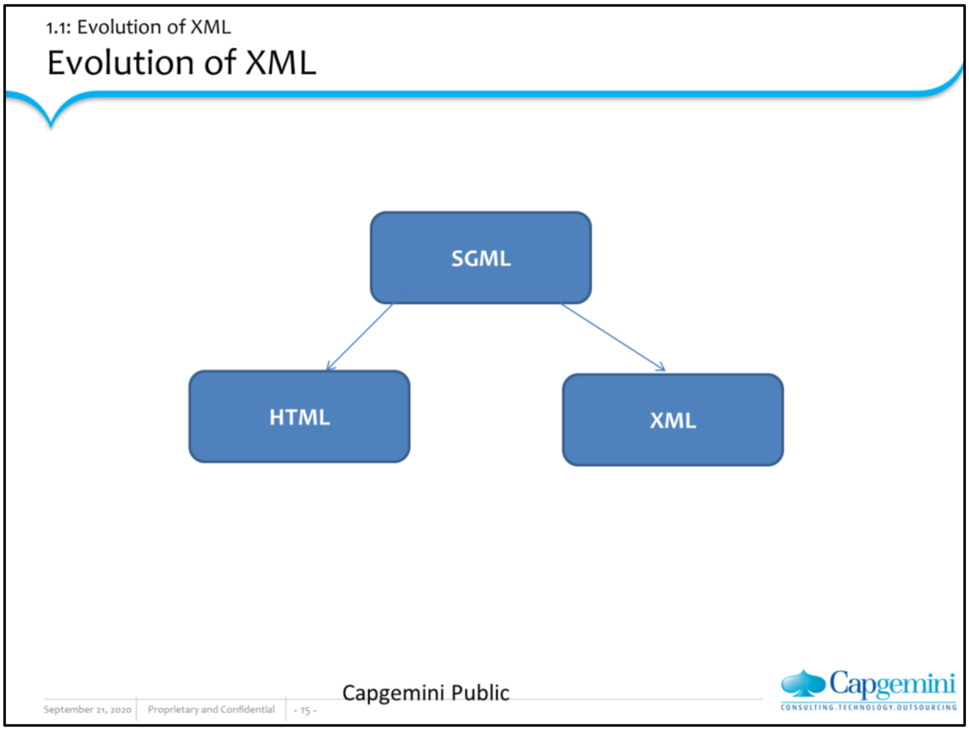
The **Standard Generalized Markup Language (SGML)**, from which XML is derived, has been around in various forms for quite some time now.

SGML may be best described by *what it is not*, as follows:

- SGML does not promote a specific document structure.
- There is not a limited tag set that must be used.
- SGML will not constrain the potential of creating new document standards.

SGML provides the common framework necessary to describe documents and to create new measures of conformance.

- Almost all languages that have been created to manipulate documents can trace at least a portion of their roots back to SGML.
- SGML itself is used by many large organizations, such as the United States Department of Defense, to handle complex electronic document exchanges. Avoiding the presentation features common to other document formats like PDF or even MS Word, SGML concentrates on the structure of the information.
- It does not promote one specific structure. However, it allows for the customized containment of data.

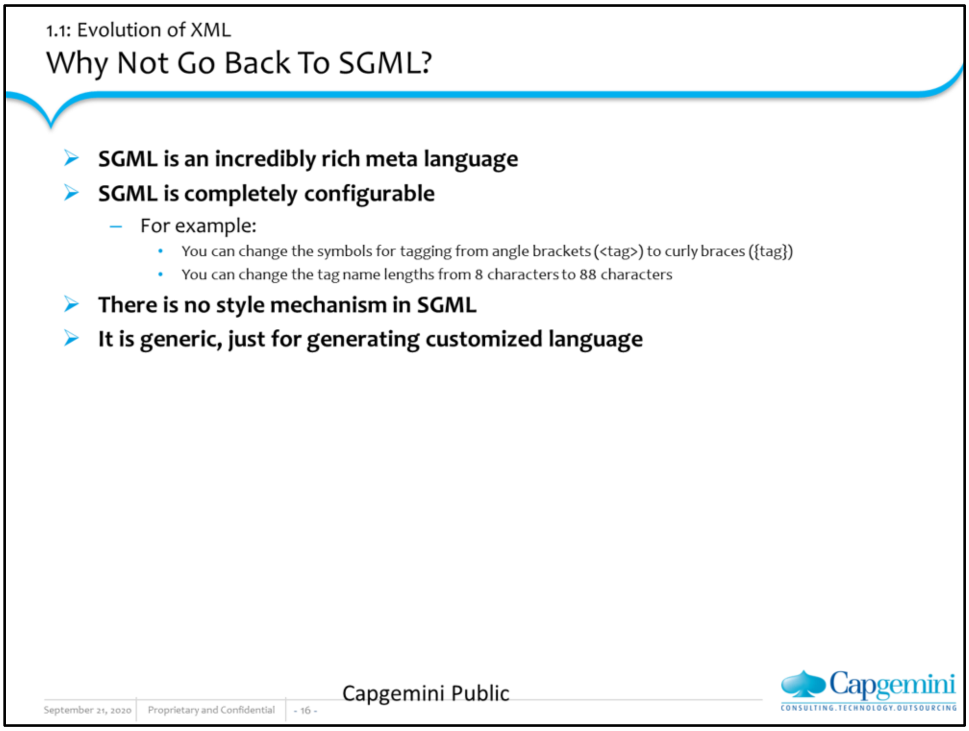


1.1: Evolution of XML

Evolution of XML

September 21, 2020 | Proprietary and Confidential | - 15 -

Capgemini Public



1.1: Evolution of XML

Why Not Go Back To SGML?

- SGML is an incredibly rich meta language
- SGML is completely configurable
  - For example:
    - You can change the symbols for tagging from angle brackets (<tag>) to curly braces ({tag})
    - You can change the tag name lengths from 8 characters to 88 characters
- There is no style mechanism in SGML
- It is generic, just for generating customized language

September 21, 2020 | Proprietary and Confidential | - 16 -

Capgemini Public

Why Not Go Back to SGML?

If HTML is limiting and SGML provides a means of describing document structures, why not simply go back to SGML?

SGML is a language for creating languages. With SGML, you can create tag sets.

Example 1: The Air Transport Association used SGML to create a tag set for aircraft maintenance documentation.

Example 2: The Society of Automotive Engineers used SGML to create a tag set for automotive service manuals.

SGML is an incredibly rich Meta language. It is completely configurable.

Example 1: You can change the symbols for tagging from angle brackets (<tag>) to curly braces ({tag}).

Example 2: You can change the tag name lengths from 8 characters to 88 characters.

All this flexibility takes up computing power on the part of tools that interact with SGML. However, for the web, you need **lightweight tools** and processes. As a subset of SGML, in many ways XML serves as a lightweight, web-compatible version of SGML.

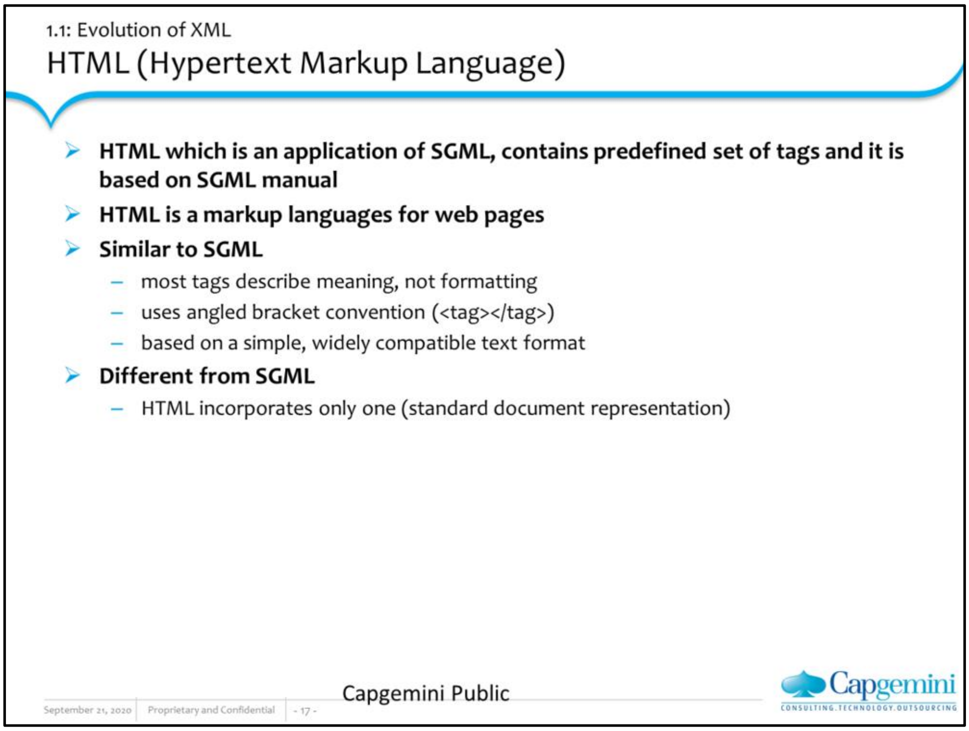
Also SGML has some more limitations

No mainstream browser support

No support for styles

Not much support for datatype

Security limitations



1.1: Evolution of XML

HTML (Hypertext Markup Language)

- **HTML which is an application of SGML, contains predefined set of tags and it is based on SGML manual**
- **HTML is a markup languages for web pages**
- **Similar to SGML**
  - most tags describe meaning, not formatting
  - uses angled bracket convention (<tag></tag>)
  - based on a simple, widely compatible text format
- **Different from SGML**
  - HTML incorporates only one (standard document representation)

September 21, 2020 | Proprietary and Confidential | - 17 -

Capgemini Public

<u>HTML</u>:

The **H**yper **T**ext **M**arkup **L**anguage (HTML), which is an Application from SGML, contains predefined set of tags. HTML is designed for web publishing intended to describe structure of text.

However, HTML is **inflexible** in that it cannot allow domain-specific tag sets to be created and used without formally introducing them into the HTML DTDs.

Limitations of HTML

Browser specific commands for e.g Netscape-specific tags (e.g., <blink>),Microsoft IE tags (e.g., <marquee>)

Limited no of tags.

Some formatting commands not separating content and presentation e.g CENTER

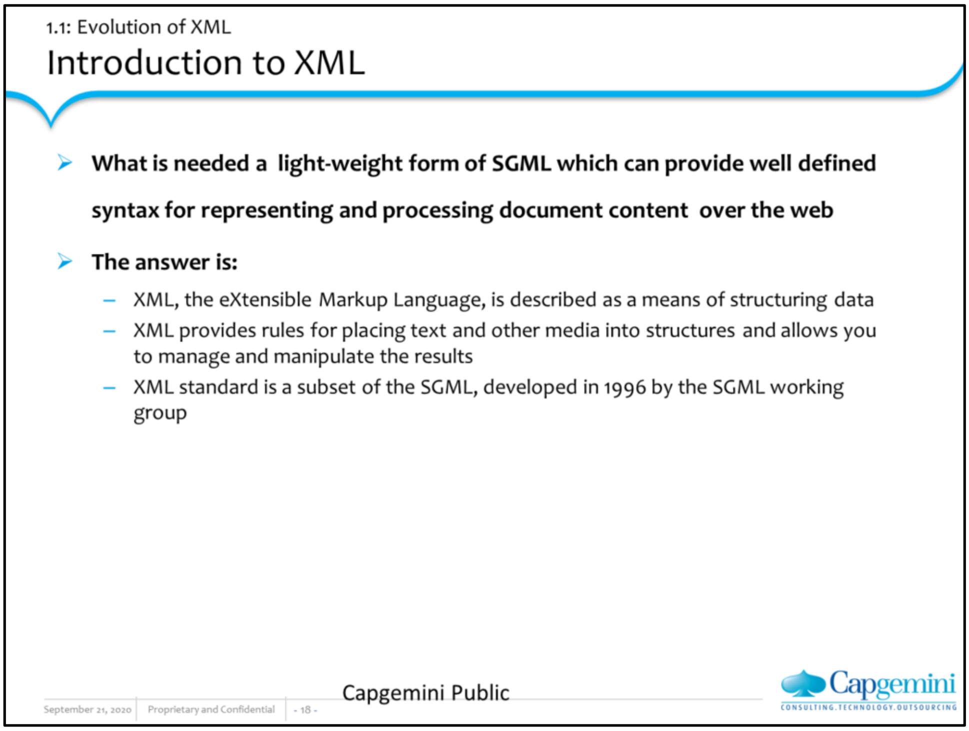
HTML authors code to the browser's standards, not the W3C standards, therefore

Pages look different in different browsers & HTML validation is difficult

HTML talks of how and not what

Introduction of CSS to separate the look of the document to some extent

Introduction of XHTML to ensure standardization



1.1: Evolution of XML

Introduction to XML

- **What is needed a light-weight form of SGML which can provide well defined syntax for representing and processing document content over the web**
- **The answer is:**
  - XML, the eXtensible Markup Language, is described as a means of structuring data
  - XML provides rules for placing text and other media into structures and allows you to manage and manipulate the results
  - XML standard is a subset of the SGML, developed in 1996 by the SGML working group

September 21, 2020 | Proprietary and Confidential | - 18 -

Capgemini Public

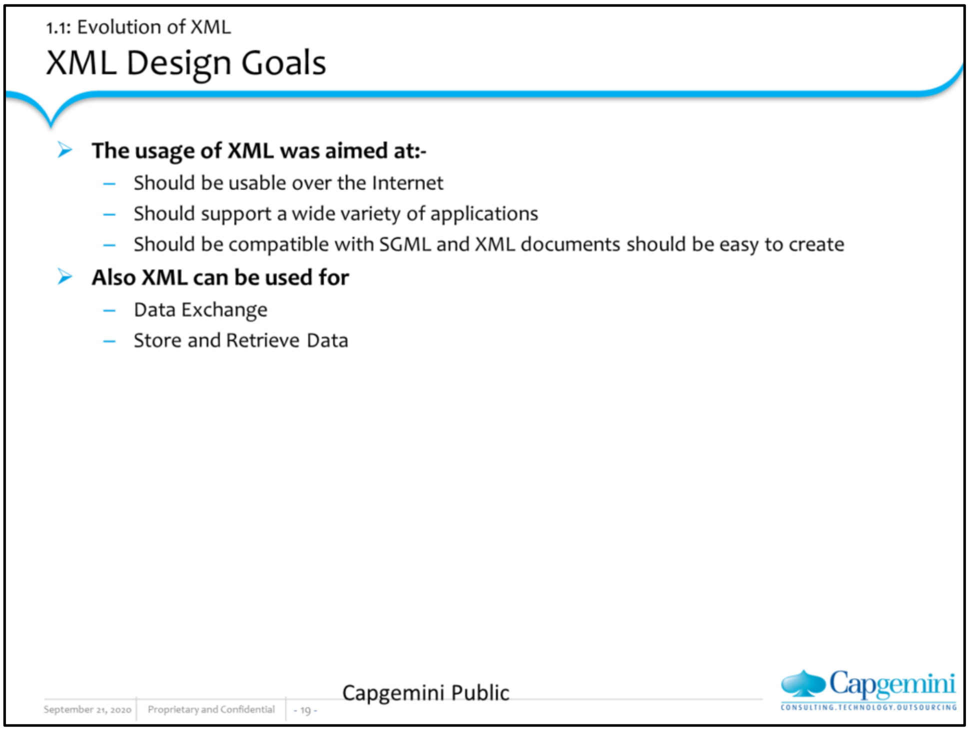
Introduction to SGML and HTML:

XML, the e**X**tensible **M**arkup **L**anguage, is best described as a means of structuring data.

XML provides rules for placing text and other media into structures and allows you to manage and manipulate the results.

The XML standard is a subset of the **Standard Generalized Markup Language (SGML)**, and was developed in 1996 by the **SGML Editorial Review Board** under the auspices of **World Wide Web Consortium (W3C)**.

XML is designed for that express purpose. XML provides a mechanism for the interchange of structured information on the web. This sort of data is required to transform the web from a publishing media to an application processing environment



1.1: Evolution of XML

XML Design Goals

- **The usage of XML was aimed at:-**
  - Should be usable over the Internet
  - Should support a wide variety of applications
  - Should be compatible with SGML and XML documents should be easy to create
- **Also XML can be used for**
  - Data Exchange
  - Store and Retrieve Data

September 21, 2020 | Proprietary and Confidential | - 19 - Capgemini Public

Capgemini CONSULTING.TECHNOLOGY.OUTSOURCING

XML Design Goals:

The design goals for XML were proposed by the World Wide Web Consortium (W3C), and published in January 1998.

The focus of XML is:

To carry the power of SGML through its ability to create user defined information about data, its ability to adapt to specific user needs, and the ability to maintain document changes.

To carry HTML's ease of use by the ability to link, its simplicity in the use of user defined tags, and its portability among different platforms.

Some more design goals are as follows:

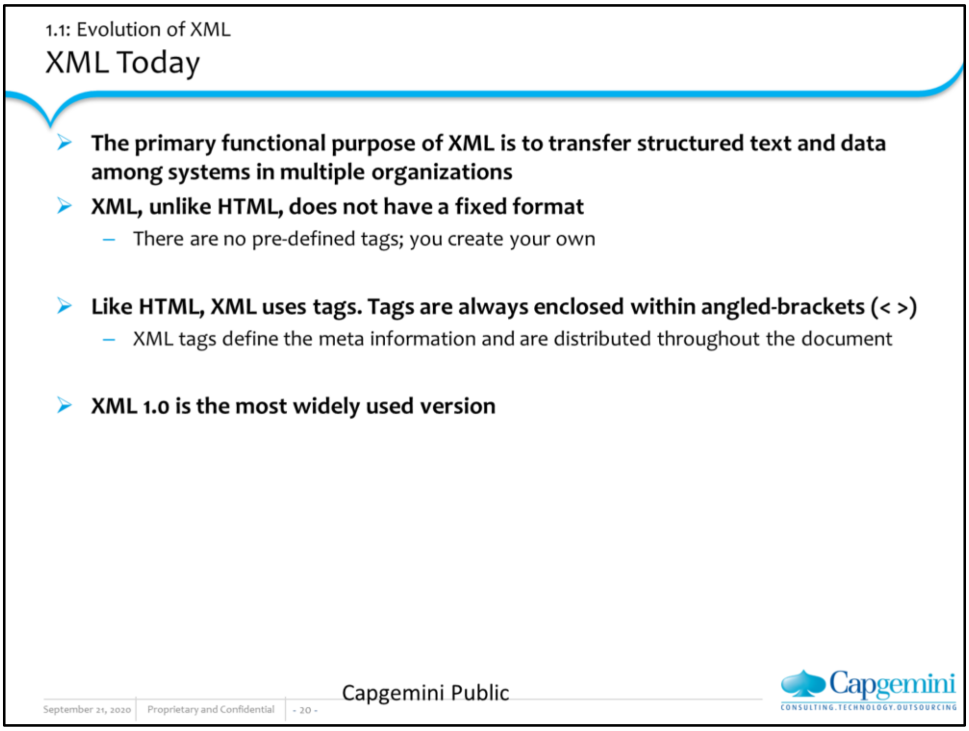
It shall be easy to write programs which process XML documents.

The number of optional features in XML should be kept to the absolute minimum, ideally zero.

XML documents should be human-legible and reasonably clear.

The XML design should be prepared quickly.

The design of XML shall be formal and concise.



1.1: Evolution of XML

XML Today

- **The primary functional purpose of XML is to transfer structured text and data among systems in multiple organizations**
- **XML, unlike HTML, does not have a fixed format**
  - There are no pre-defined tags; you create your own
- **Like HTML, XML uses tags. Tags are always enclosed within angled-brackets (< >)**
  - XML tags define the meta information and are distributed throughout the document
- **XML 1.0 is the most widely used version**

September 21, 2020 | Proprietary and Confidential | - 20 -

Capgemini Public

About XML:
XML versus HTML:

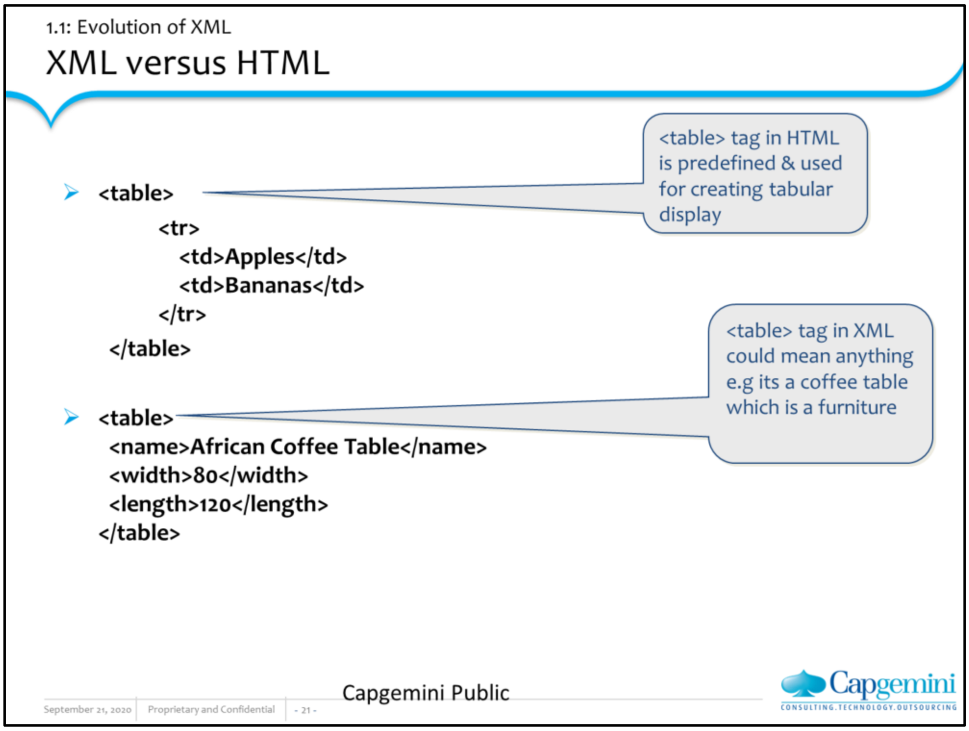
Both are based on SGML – the International Standard for structured information.

In HTML:		In XML:
<p>P200 Laptop	<product>
<br>Friendly Computer Shop		<model>P200 Laptop</model>
<br>$1438		<dealer>Friendly Computer Shop</dealer>
		<price>$1438</price>
		</product>

Both XML and HTML may appear the same in your browser. However, the XML data is smart data. HTML tells how the data should look, but XML tells you what it means.

With XML, your browser knows there is a product, and it knows the model, dealer, and price. From a group of these, it can show you the cheapest product or closest dealer without going back to the server.

Unlike HTML, with XML you create your own tags, so they describe exactly what you need to know. As a result, your client-side applications can access data sources anywhere on the Web, and in any format. New "middle-tier" servers sit between the data sources and the client, translating everything into your own task-specific XML.



1.1: Evolution of XML

XML versus HTML

```
<table>
        <tr>
           <td>Apples</td>
           <td>Bananas</td>
        </tr>
 </table>
```

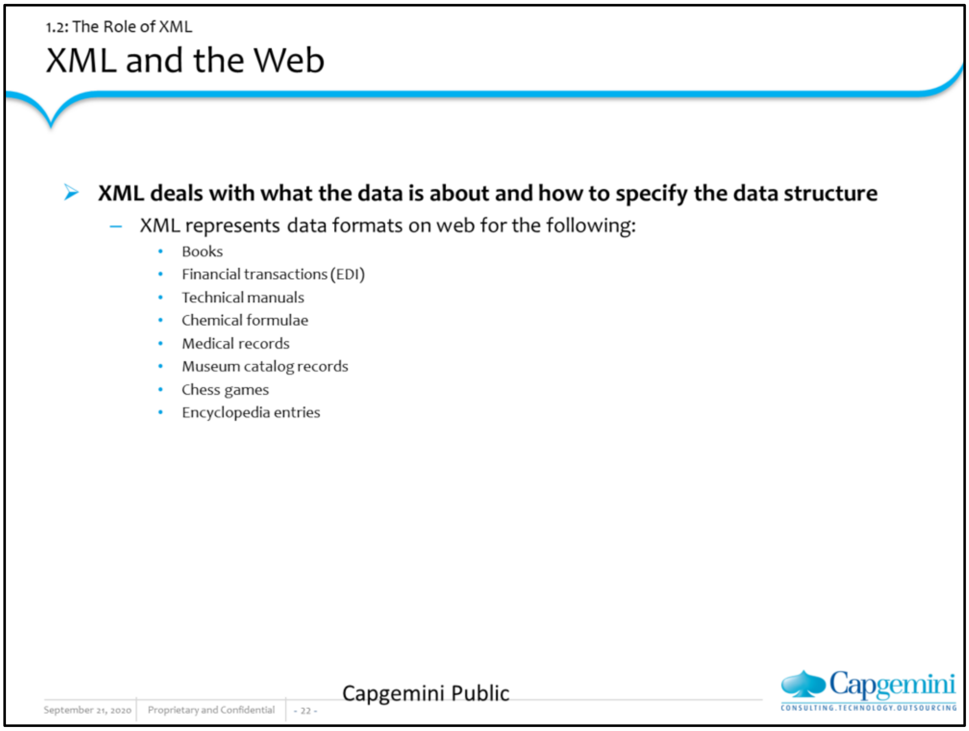
<table> tag in HTML is predefined & used for creating tabular display

```
<table>
 <name>African Coffee Table</name>
 <width>80</width>
 <length>120</length>
</table>
```

<table> tag in XML could mean anything e.g its a coffee table which is a furniture

September 21, 2020 | Proprietary and Confidential | - 21 -

Capgemini Public



1.2: The Role of XML

XML and the Web

- **XML deals with what the data is about and how to specify the data structure**
  - XML represents data formats on web for the following:
    - Books
    - Financial transactions (EDI)
    - Technical manuals
    - Chemical formulae
    - Medical records
    - Museum catalog records
    - Chess games
    - Encyclopedia entries

September 21, 2020 | Proprietary and Confidential | - 22 -

Capgemini Public

The Role of XML:

XML will be most interesting to people and organizations who have:

Information resources that do not fit into the HTML mold, and

Resources that they want to make available over the web

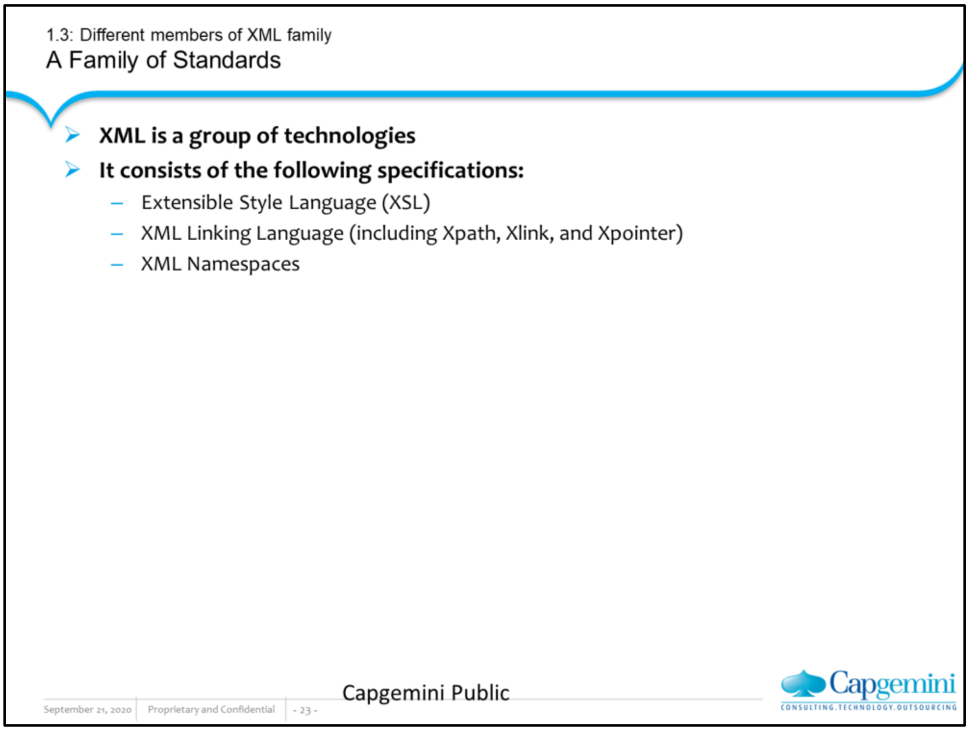
XML was created to structure, store, and transport information. It is just plain text. Software that can handle plain text can also handle XML. However, XML-aware applications can specially handle the XML tags. The functional meaning of the tags depends on the nature of the application.

XML is as important for the web, as HTML was to the foundation of the web.

XML is everywhere. It is the most common tool for data transmissions between all sorts of applications, and is popular in the area of storing and describing information.

Some examples:

Books
Financial transactions (EDI)
Technical manuals
Chemical formulae
Medical records
Museum catalog records
Chess games
Encyclopedia entries



1.3: Different members of XML family

A Family of Standards

- **XML is a group of technologies**
- **It consists of the following specifications:**
  - Extensible Style Language (XSL)
  - XML Linking Language (including Xpath, Xlink, and Xpointer)
  - XML Namespaces

September 21, 2020 | Proprietary and Confidential | - 23 -

Capgemini Public

Introducing XML and its Relatives:

XML is a group of technologies.

It consists of the following specifications:

eXtensible Style Language: XSL works with the XML data in a way similar to the manner in which CSS works with HTML.

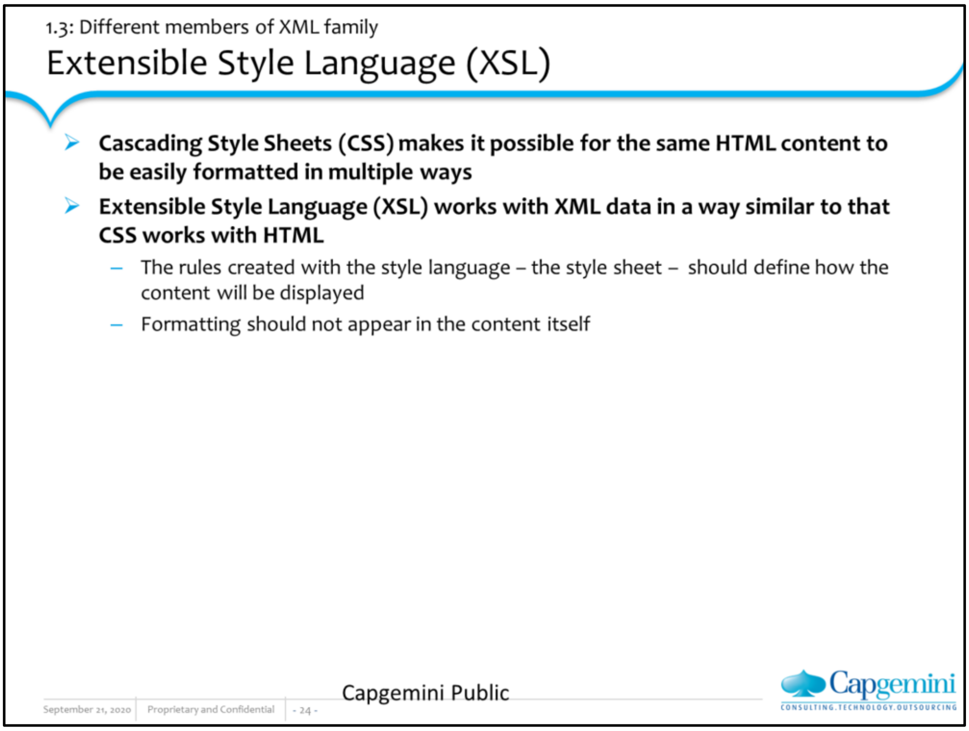
XML Linking: XML linking and addressing mechanisms are specified in three W3C Working Draft documents:

XML Path Language (Xpath): The primary purpose of Xpath is to do the actual addressing of parts rather than the whole XML document.

XML Linking Language (Xlink): It uses XML syntax to create structures to describe both "simple unidirectional links" of today's HTML as well as more "sophisticated multidirectional links". The important part of Xlink is that is defines the relationship between two or more data objects (or portion of objects) as opposed to a whole document.

XML Pointer Language (Xpointer): Xpointer builds on Xpath to support addressing into the internal structures of XML documents. Thus you can use the XML markup to link to specific parts of another document without supplying an ID reference.

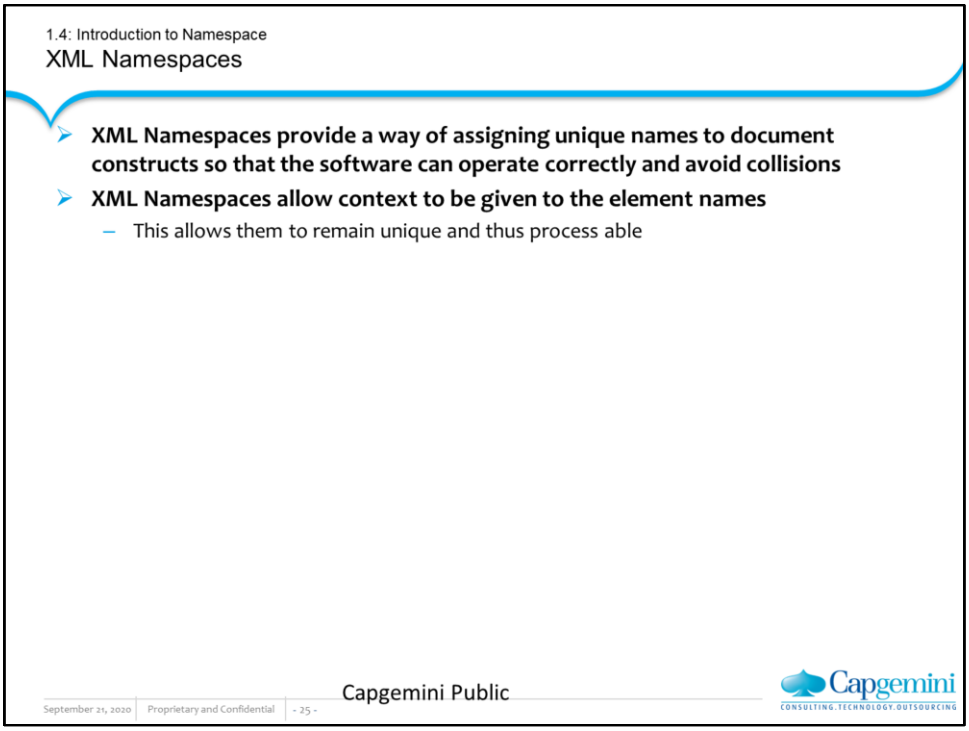
XML Namespaces



1.3: Different members of XML family

Extensible Style Language (XSL)

- **Cascading Style Sheets (CSS) makes it possible for the same HTML content to be easily formatted in multiple ways**
- **Extensible Style Language (XSL) works with XML data in a way similar to that CSS works with HTML**
  - The rules created with the style language – the style sheet – should define how the content will be displayed
  - Formatting should not appear in the content itself



1.4: Introduction to Namespace

XML Namespaces

- **XML Namespaces provide a way of assigning unique names to document constructs so that the software can operate correctly and avoid collisions**
- **XML Namespaces allow context to be given to the element names**
  - This allows them to remain unique and thus process able

September 21, 2020 | Proprietary and Confidential | - 25 -

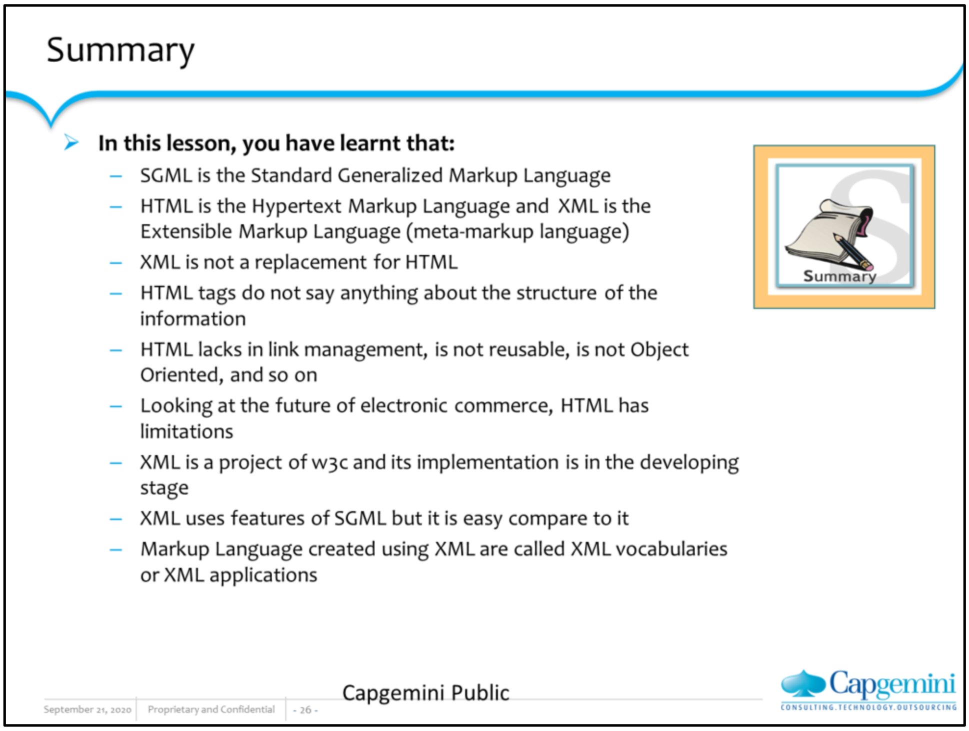
Capgemini Public

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

XML Namespaces:

XML Namespaces are a way of assigning "unique names" to document constructs so that software can operate correctly and avoid collisions.

Namespaces allow context to be given to element names, which allow them to remain unique and thus processable.



Summary

- **In this lesson, you have learnt that:**
  - SGML is the Standard Generalized Markup Language
  - HTML is the Hypertext Markup Language and XML is the Extensible Markup Language (meta-markup language)
  - XML is not a replacement for HTML
  - HTML tags do not say anything about the structure of the information
  - HTML lacks in link management, is not reusable, is not Object Oriented, and so on
  - Looking at the future of electronic commerce, HTML has limitations
  - XML is a project of w3c and its implementation is in the developing stage
  - XML uses features of SGML but it is easy compare to it
  - Markup Language created using XML are called XML vocabularies or XML applications

September 21, 2020 | Proprietary and Confidential | - 26 -

Capgemini Public

Summary:

Why XML?

Many of the most influential companies in the software industry are promoting XML as the next step in the evolution of the web. How can they be so confident about something so new?

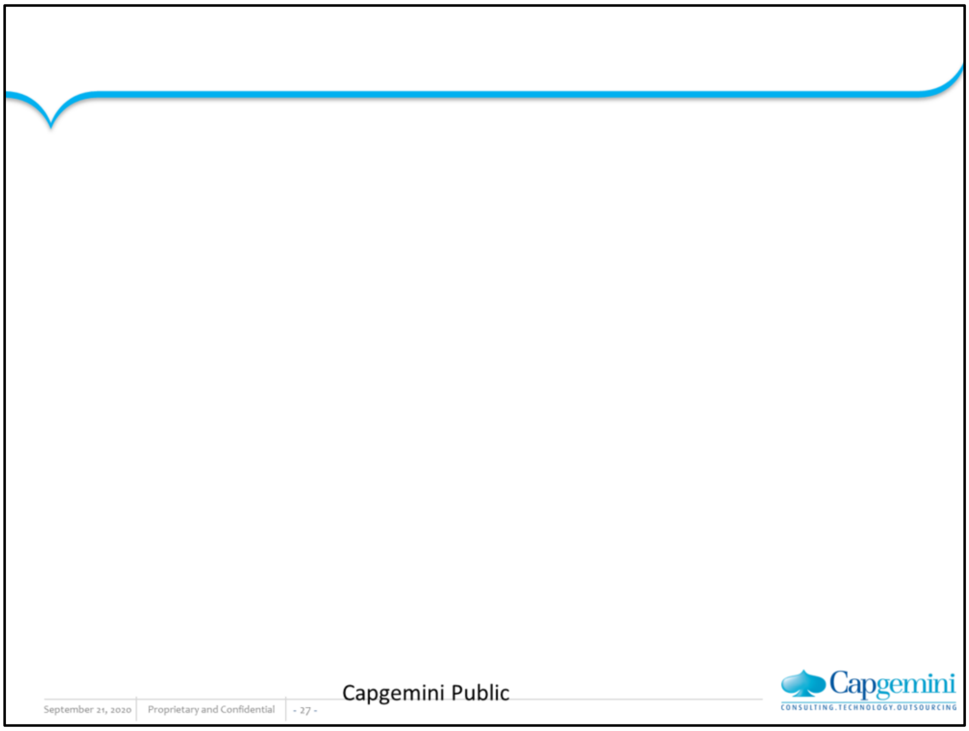
We can all safely bet on XML because the central ideas in this new technology are in fact very old and have been proven correct across several decades and thousands of projects.

Database Publishing:

One particularly popular application of XML will surely be the publishing of databases to the web.

Consider for instance a product database, used by the internal ordering system of a toy manufacturer. The manufacturer might want the database to be available on the web so that potential clients know the toys that are available and their price.

Rather than having someone in the web design department to mark up the data again, they can build a connection between their web server and their database using the features typically built into web servers that allows those sorts of data pipes. The designer can then make the product list beautiful using a style sheet. Pictures of the toys can be supplied by the database. In essence, the web site will be merely a view on the data in the database. As toys get added and removed from the database, they will appear and disappear from the view on the website.



Capgemini Public

September 21, 2020 | Proprietary and Confidential | - 27 -

Capgemini

CONSULTING.TECHNOLOGY.OUTSOURCING

Summary:

Database Publishing (contd.):

XML is also expected to become an important tool for interchange of database information. They are the “documents produced by and for computer software”. Databases have typically interchanged information using simple file formats such as one-record per line with semi-colons between the fields. This is not sufficient for the new object-oriented information being produced by databases. Objects must have internal structure and links between them. XML can represent this using elements and attributes to provide a common format for transferring database records between databases.

Today’s web model is a “client/server” model. Queries from the customer go to the server, and resulting responses are shipped back to the customer for viewing in HTML. Unfortunately, a web server can handle only a limited number of connections at one time.

Today, XML has enabled a new breed of web server software, one that allows the web developer to add a new “middle-tier” server to the web model.

Summary:

Database Publishing (contd.):

In the old web model, the customer using browser such as IE or Netscape on the client interacted directly with data sources on remote servers. The client maintained its connection throughout the interactive session. Each query was sent a response in HTML, which could be directly viewed by the client browser. Maintaining the connection between the client and server was critical.

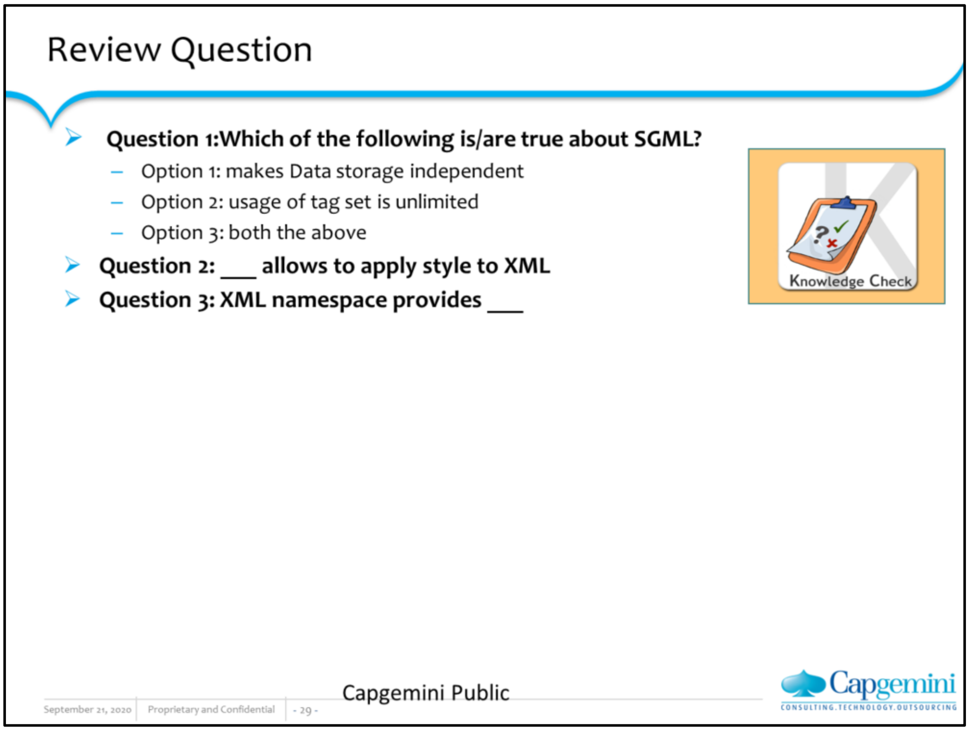
In the new three-tier web model, the information that fits the profile of the customer is retrieved at once from the remote databases by software on the middle tier, either as XML documents or through an ODBC or similar database connection. From that point, continued interaction with the remote databases is no longer required. The connection to the remote servers can be, and is, terminated.

Once all information that fits the customer profile has been assembled by software on the middle tier, it is sent in XML to the client. Now the requirement for further interaction between the client and the middle tier server is eliminated as well.

Rich XML data, directly usable by client applications and scripting languages like JavaScript, has been delivered to the client. The connection between the client and the middle tier server can now be terminated. At this point, all computing becomes client-based, resulting in a much more efficient use of the web and a much more satisfying customer experience.

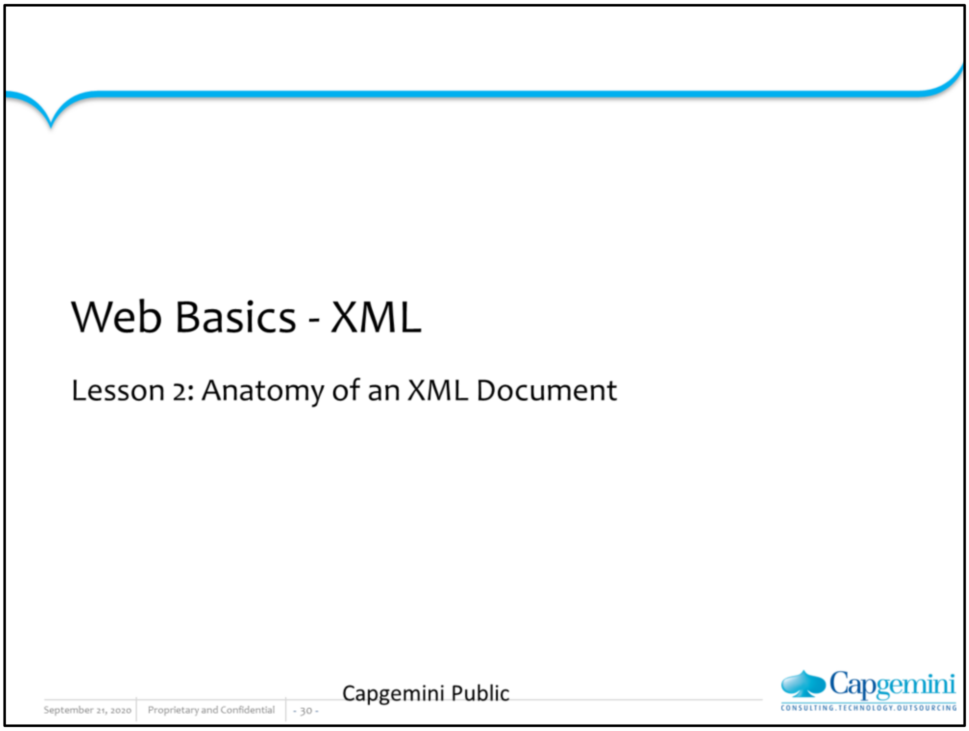
Answers

1. Option 3
2. XSL
3. Unique names



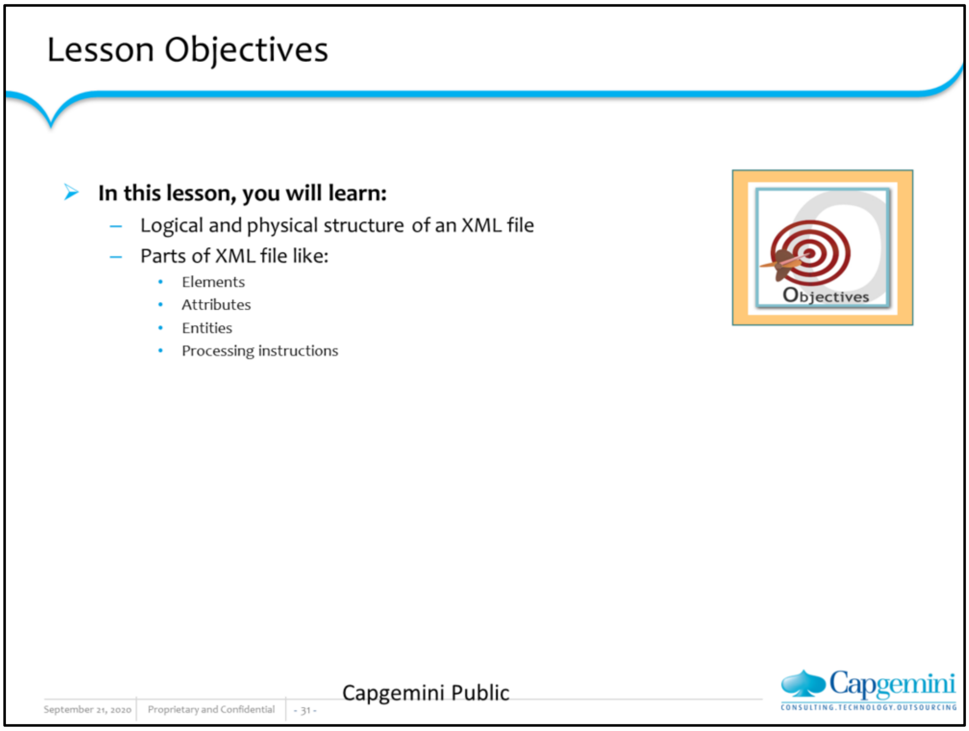
Review Question

- **Question 1:Which of the following is/are true about SGML?**
  - Option 1: makes Data storage independent
  - Option 2: usage of tag set is unlimited
  - Option 3: both the above
- **Question 2: ___ allows to apply style to XML**
- **Question 3: XML namespace provides ___**



Web Basics - XML

Lesson 2: Anatomy of an XML Document



Lesson Objectives

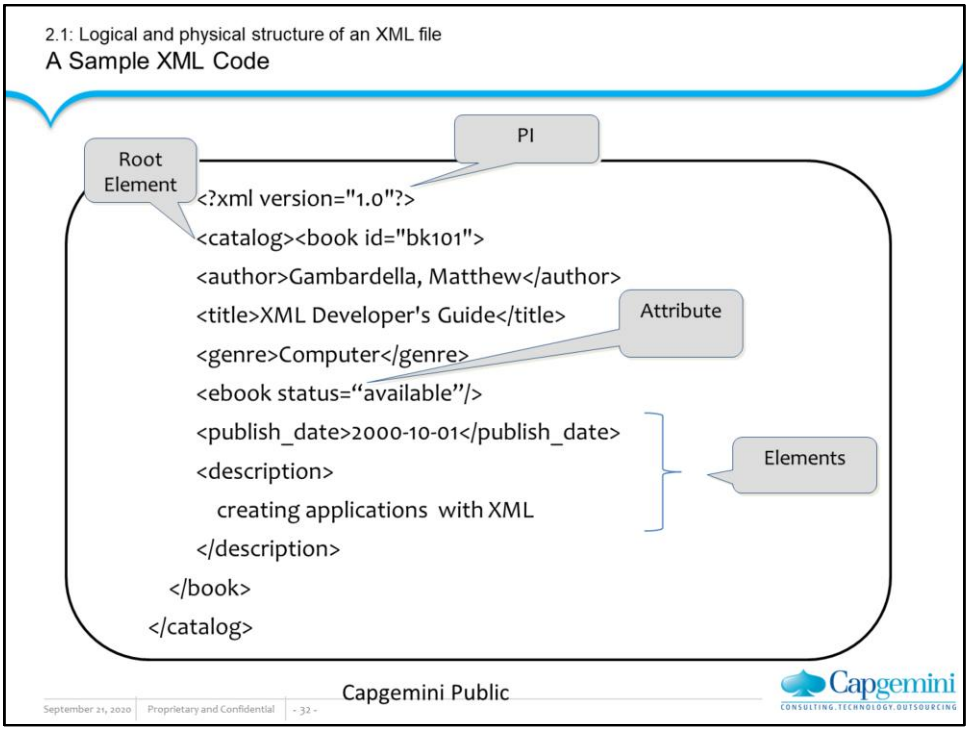
- **In this lesson, you will learn:**
  - Logical and physical structure of an XML file
  - Parts of XML file like:
    - Elements
    - Attributes
    - Entities
    - Processing instructions

Objectives

September 21, 2020 | Proprietary and Confidential | - 31 -

Capgemini Public

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING



2.1: Logical and physical structure of an XML file

A Sample XML Code

```
    <?xml version="1.0"?>
    <catalog><book id="bk101">
    <author>Gambardella, Matthew</author>
    <title>XML Developer's Guide</title>
    <genre>Computer</genre>
    <ebook status=“available”/>
    <publish_date>2000-10-01</publish_date>
    <description>
      creating applications  with XML
    </description>
  </book>
</catalog>
```

September 21, 2020 | Proprietary and Confidential | - 32 -

Capgemini Public

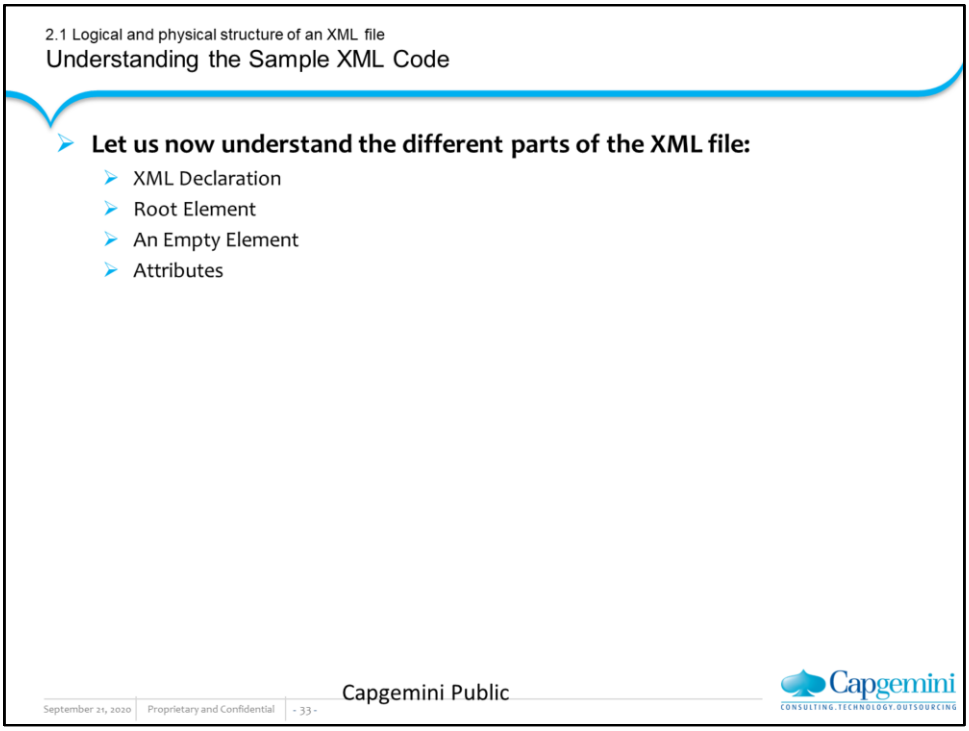
Capgemini CONSULTING.TECHNOLOGY.OUTSOURCING

Elements can be simple, empty, mixed

Simple : <data>value</data>

```
Mixed  :  <data> value
                <sub>sub1</sub1>
                </data>
```

Empty : <data></data>



2.1 Logical and physical structure of an XML file

Understanding the Sample XML Code

- Let us now understand the different parts of the XML file:
  - XML Declaration
  - Root Element
  - An Empty Element
  - Attributes

Capgemini Public

September 21, 2020 | Proprietary and Confidential | - 33 -

Understanding the Sample XML Code:

XML Declaration:

It is a processing instruction (identified by the ? at its start and end).

Root Element:

Each XML document must have only one root element, all the other elements must be completely enclosed in that element.

Line 2 (in example) identifies the start element (the start tag), and line 12 identifies the end of the element (the end tag).

Empty Elements:

Empty elements have no content and are marked up as either of the following:

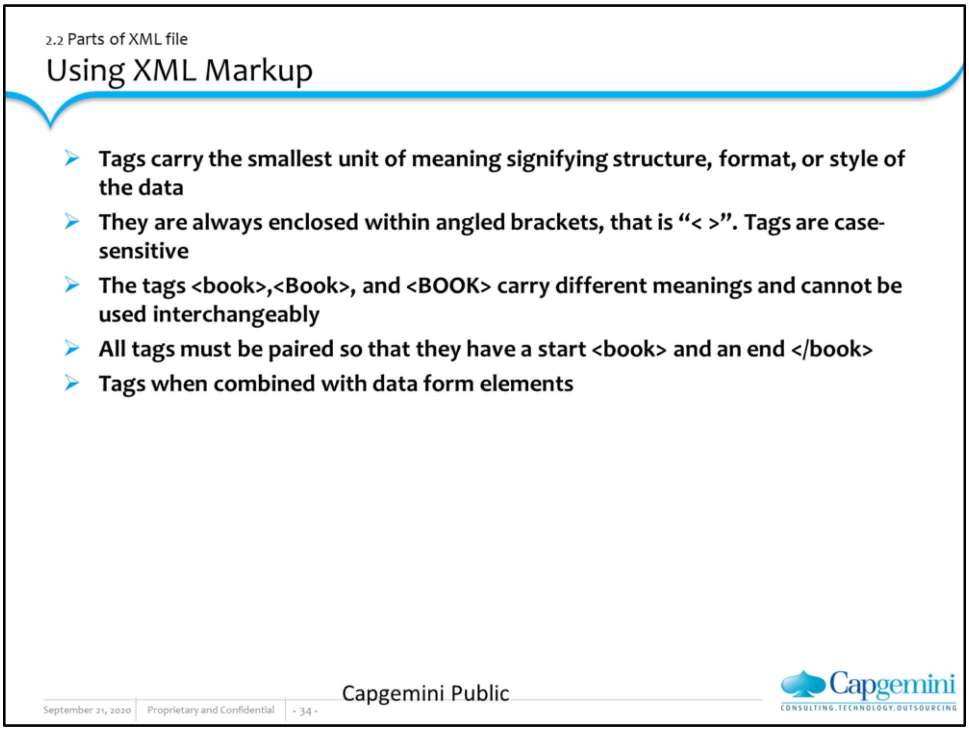
<empty_element/>

<empty_element></empty_element>

Attribute Markup:

Attributes are used to attach information to the information contained in an element. The general form for using an attribute is as follows:

<element-name property=”value”>



2.2 Parts of XML file

Using XML Markup

- **Tags carry the smallest unit of meaning signifying structure, format, or style of the data**
- **They are always enclosed within angled brackets, that is "< >". Tags are case-sensitive**
- **The tags <book>,<Book>, and <BOOK> carry different meanings and cannot be used interchangeably**
- **All tags must be paired so that they have a start <book> and an end </book>**
- **Tags when combined with data form elements**

September 21, 2020 | Proprietary and Confidential | - 34 -

Capgemini Public

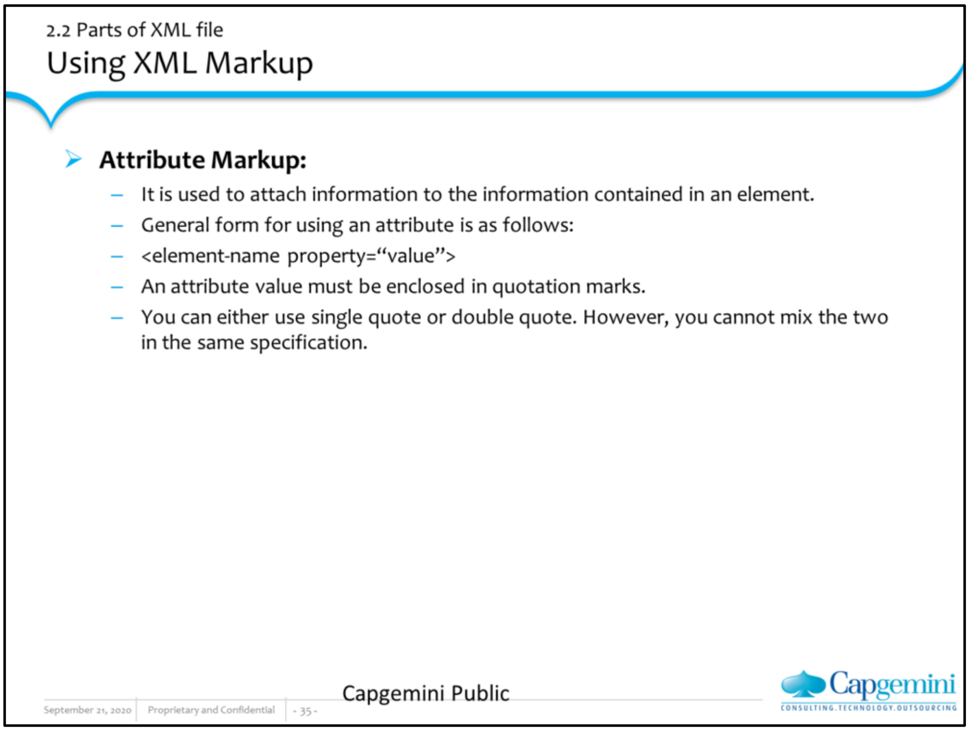
Using XML Markup:

XML is concerned with element markup. Instead of XML's tags being markers that indicate where a style should change or a new line should begin, XML's element markup is composed of three parts:

- The start tag
- The content
- The end tag

Elements:

- Elements contain information or content and can also contain other elements.
- There is one element that contains all the other elements called the "root element".
- Tags show the beginning and end of an element.
- XML documents are divided into containers called "elements". People who are familiar with HTML, know that <p> …. </p>, <form> … </form>, <br> are all elements.



2.2 Parts of XML file

Using XML Markup

- **Attribute Markup:**
  - It is used to attach information to the information contained in an element.
  - General form for using an attribute is as follows:
  - <element-name property=“value”>
  - An attribute value must be enclosed in quotation marks.
  - You can either use single quote or double quote. However, you cannot mix the two in the same specification.

September 21, 2020 | Proprietary and Confidential | - 35 -

Capgemini Public

Using XML Markup:

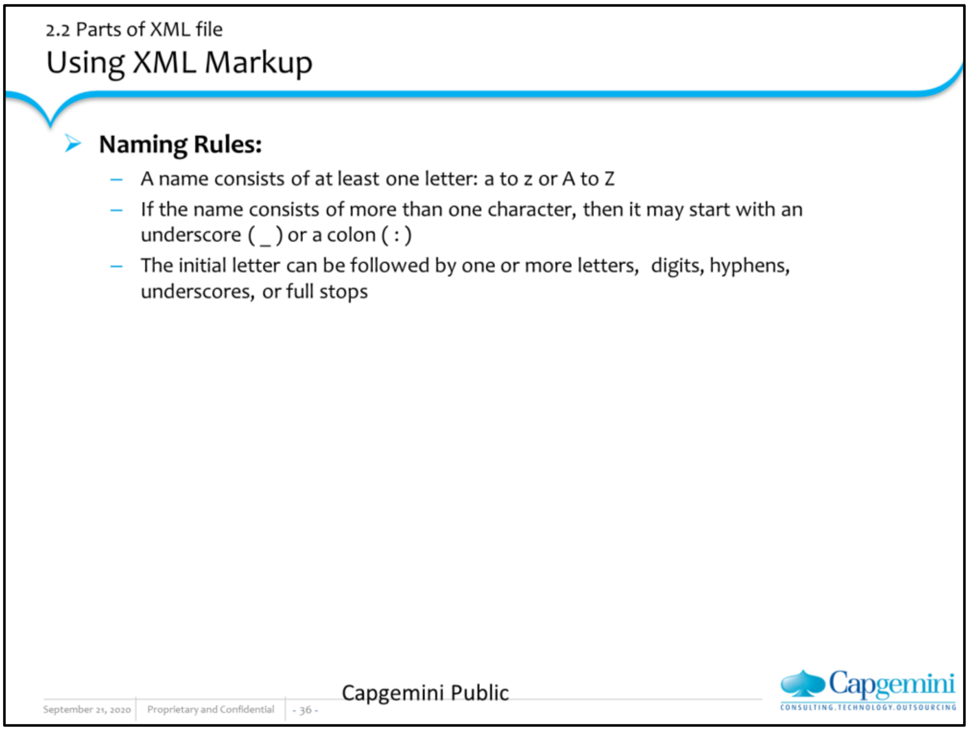
Attributes:

Attributes are element modifiers. They provide additional and more specific information about an element and its content.

Normally in HTML, attributes are used most often to provide the browser with a suggestion for formatting the display of the elements content by a browser.

For example: bgcolor attribute of <body> element or align attribute normally are used with almost all elements.

However, the same is not true with XML. The attributes are used to provide further information about the element itself. This is because the main purpose of XML is to separate markup from display, so you will rarely see formatting attributes in XML DTDs.



2.2 Parts of XML file

Using XML Markup

- **Naming Rules:**
  - A name consists of at least one letter: a to z or A to Z
  - If the name consists of more than one character, then it may start with an underscore (_) or a colon (:)
  - The initial letter can be followed by one or more letters, digits, hyphens, underscores, or full stops

September 21, 2020 | Proprietary and Confidential | - 36 -

Capgemini Public

Capgemini CONSULTING.TECHNOLOGY.OUTSOURCING

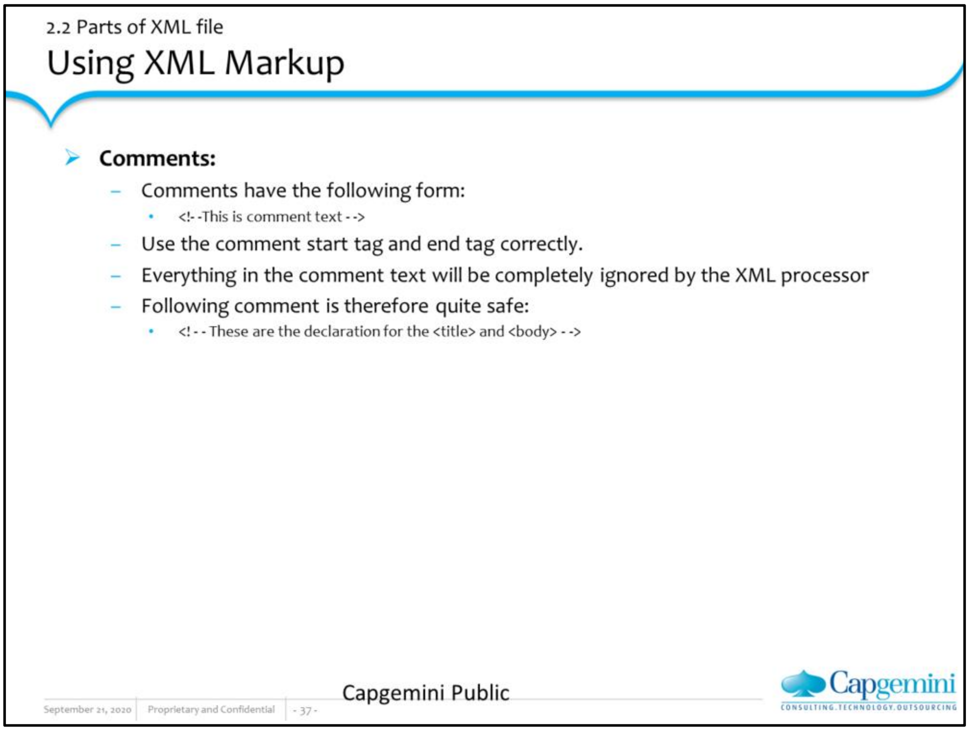
Using XML Markup:

Naming Rules:

A name consists of at least one letter: a to z or A to Z.

If the name consists of more than one character, then it may start with an underscore (_) or a colon (:)

The initial letter can be followed by one or more letters, digits, hyphens, underscores, and full stops.



2.2 Parts of XML file

Using XML Markup

- **Comments:**
  - Comments have the following form:
    - <!- -This is comment text - ->
  - Use the comment start tag and end tag correctly.
  - Everything in the comment text will be completely ignored by the XML processor
  - Following comment is therefore quite safe:
    - <! - - These are the declaration for the <title> and <body> - ->

Using XML Markup:
Comments

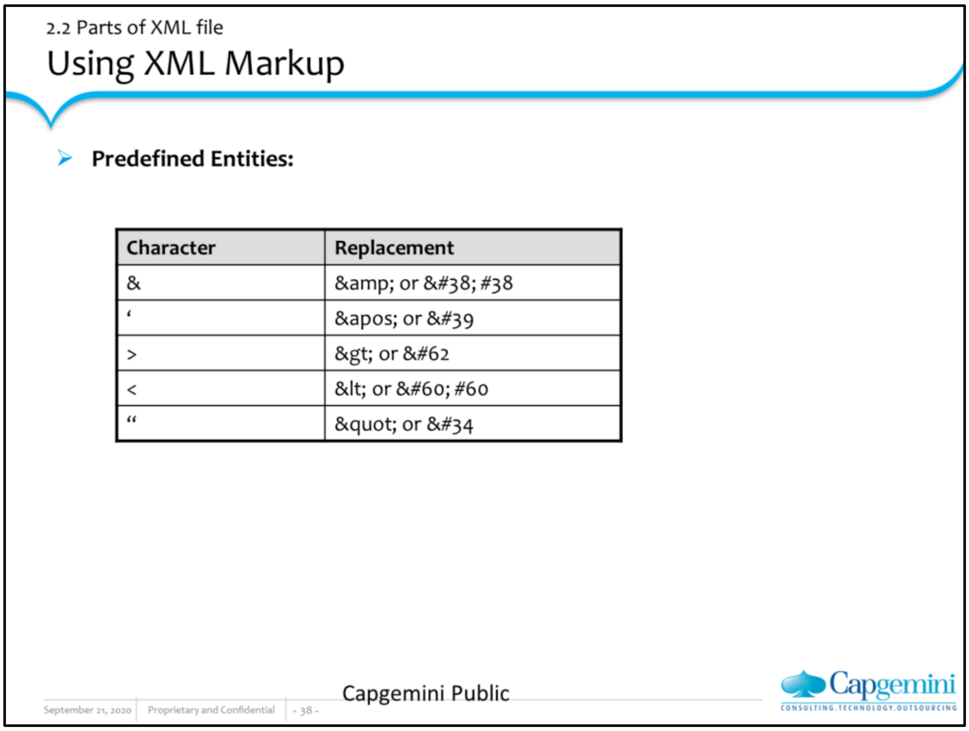
In keeping with the design constraint of keeping XML simple, its comment facilities are also simple. Comments have the following form:

<!- -This is comment text - ->

Provided that you use the comment start tag and end tag correctly, everything in the comment text will be completely ignored by the XML processor. The following comment is therefore quite safe:

<! - - These are the declaration for the <title> and <body> - ->

There is only one restriction on what you can place in your comment text: the string - - is not allowed. This keeps XML backward compatible with SGML.



2.2 Parts of XML file

Using XML Markup

- **Predefined Entities:**

Character	Replacement
&	&amp; or &#38; #38
‘	&apos; or &#39
>	&gt; or &#62
<	&lt; or &#60; #60
“	&quot; or &#34

September 21, 2020 | Proprietary and Confidential | - 38 -

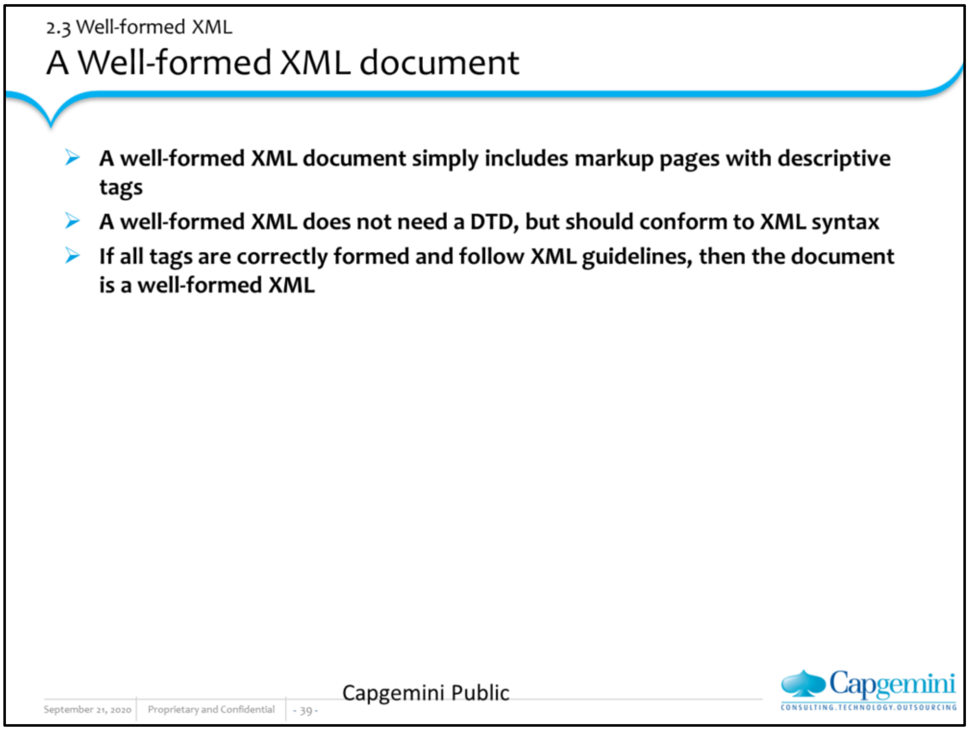
Capgemini Public

Using XML Markup:

Predefined Entities:

The special characters for quote ("), apostrophe ('), less-than (<), greater-than (>), and ampersand (&) are used for punctuation in XML, and are represented with predefined entities: &quot;, &apos;, &lt;, &gt;, and &amp;.

Notice that the semicolon is part of the entity. You cannot use “<“ or “&” in attributes or elements.



2.3 Well-formed XML

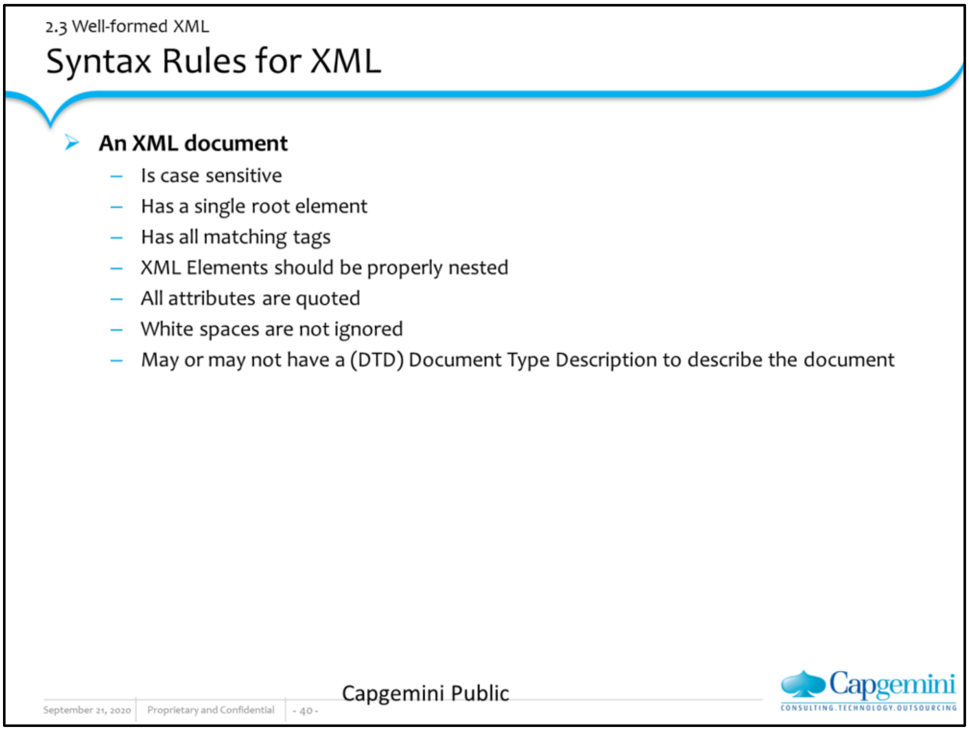
A Well-formed XML document

- **A well-formed XML document simply includes markup pages with descriptive tags**
- **A well-formed XML does not need a DTD, but should conform to XML syntax**
- **If all tags are correctly formed and follow XML guidelines, then the document is a well-formed XML**

September 21, 2020 | Proprietary and Confidential | - 39 -

Capgemini Public

The XML syntax is discussed on the next slide.



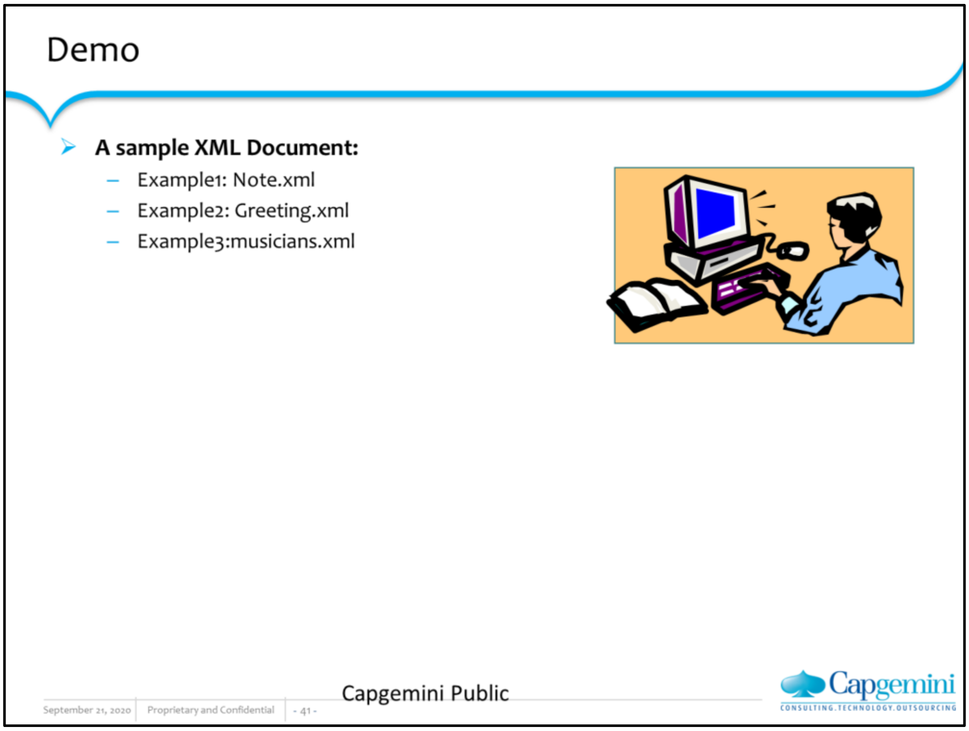
2.3 Well-formed XML

Syntax Rules for XML

- **An XML document**
  - Is case sensitive
  - Has a single root element
  - Has all matching tags
  - XML Elements should be properly nested
  - All attributes are quoted
  - White spaces are not ignored
  - May or may not have a (DTD) Document Type Description to describe the document

September 21, 2020 | Proprietary and Confidential | - 40 -

Capgemini Public



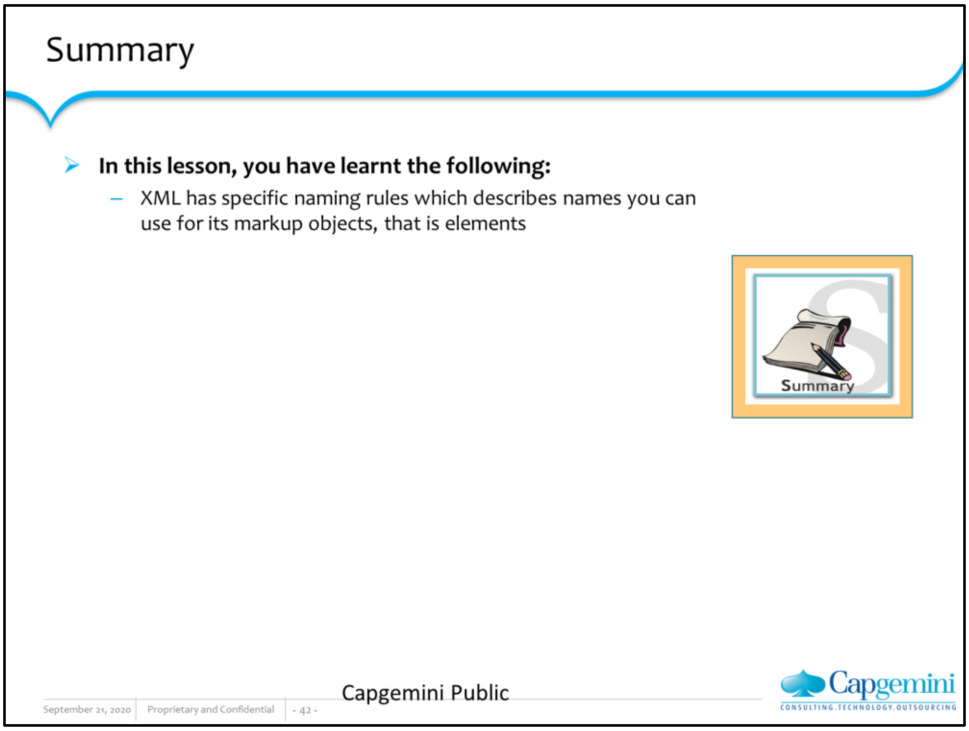
Demo

- **A sample XML Document:**
  - Example1: Note.xml
  - Example2: Greeting.xml
  - Example3:musicians.xml

September 21, 2020 | Proprietary and Confidential | - 41 -

Capgemini Public

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING



Summary

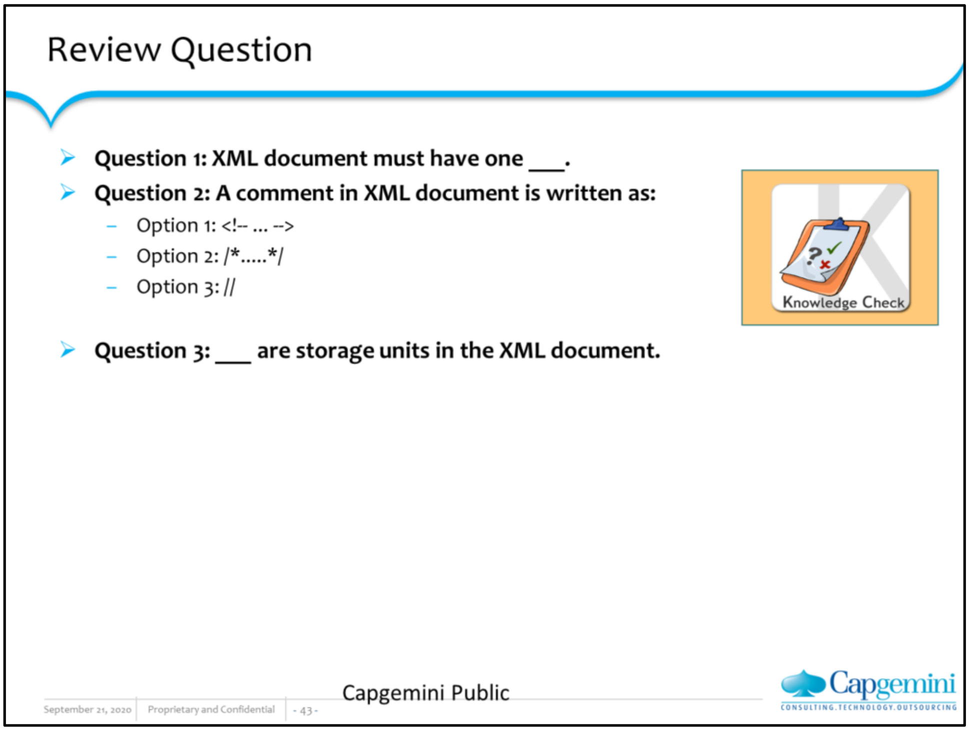
- **In this lesson, you have learnt the following:**
  - XML has specific naming rules which describes names you can use for its markup objects, that is elements

September 21, 2020 | Proprietary and Confidential | - 42 -

Capgemini Public

Answers

1. Root element
2. Option 1
3. Entities



Review Question

- Question 1: XML document must have one ___.
- Question 2: A comment in XML document is written as:
  - Option 1: <!-- … -->
  - Option 2: /*…..*/
  - Option 3: //
- Question 3: ___ are storage units in the XML document.

Knowledge Check

September 21, 2020 | Proprietary and Confidential | - 43 -

Capgemini Public

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING