# SYED AMMAL ENGINEERING COLLEGE

(Approved by the AICTE, New Delhi, Govt. of Tamilnadu and Affiliated to Anna University,Chennai)

Established in 1998 - An ISO 9001:2008 Certified Institution

Computer Science and Engineering

## *IT6801 - SERVICE ORIENTED ARCHITECTURE*

## 2 MARK QUESTION WITH ANSWER

### UNIT-I

**1. What is XML?**

XML is a set of rules for structuring, storing and transferring information. This language is used to describe the data which will be passed from one computer application to another. XML tells a computer what the actual data is, not what it should look like.

**2. What is the main disadvantage of HTML?**

The main disadvantage was that it was not designed to share information between computers, and so XML was developed to overcome this limitation.

**3. What are the uses of XML?**

- Connecting databases to the Web; Exchanging data automatically between different computer applications;
- Moving the processing from a Web server to the local PC;
- Using the same information in many different ways;
- Changing the presentation of information automatically for different viewing devices.

**4. What is the emergence of XML?**

- **XLINK** - a standard designed to hyperlink between XML documents;
- **XML Query -** a language used to query XML documents;
- **XSL -** a style sheet language for XML;
- **Resource Description Framework (RDF) -** a standard for metadata. This will be similar to library cards and should make searching the Web much faster

**5. What are the major XML news formats?**

The major XMLNews formats are XMLNews-Story and XMLNews-Meta.

**6. What are markup and text in an XML document?**

XML documents mix markup and text together into a single file: the markup describes the structure of the document, while the text is the documents content

**7. Write the rules of XML declaration**

- The XML declaration is case sensitive: it may not begin with "<?XML" or any other variant;
- If the XML declaration appears at all, it must be the very first thing in the XML document: not even white space or comments may appear before it; and
- It is legal for a transfer protocol like HTTP to override the encoding value that you put in the XML declaration, so you cannot guarantee that the document will actually use the encoding provided in the XML declaration.

**8. Write the rules of XML element**

Elements may not overlap: an end tag must always have the same name as the most recent unmatched start tag. The following example is not well-formed XML, because "</person>" appears when the most recent unmatched start tag was "<function>":

b. <!-- WRONG! -->

c. **<function><person>**President**</function>** Habibe**</person>**

**9. Write on Attributes**

XML start tags also provide a place to specify **attributes**. An attribute specifies a single property for an element, using a name/value pair. One very well known example of an attribute is *href* in HTML:
<a href=\"http://www.yahoo.com/\">Yahoo!</a>

**10. What are the revolutions of XML?**

1. Data Revolution
2. Architectural Revolution
3. Software Revolution
4.

**11. What is SOA?**

SOA is an architectural style whose goal is to achieve loose coupling among interacting software agents. A service is a unit of work done by a service provider to achieve desired end results for a service consumer. Both provider and consumer are roles played by software agents on behalf of their owners.

**12. Define Stateless service**

Each message that a consumer sends to a provider must contain all necessary information for the provider to process it. This constraint makes a service provider more scalable because the provider does not have to store state information between requests. This is effectively service in mass production since each request can be treated as generic.

**13. Define Stateful service**

Stateful service is difficult to avoid in a number of situations. One situation is to establish a session between a consumer and a provider. A session is typically established for efficiency reasons. For example, sending a security certificate with each request is a serious burden for both any consumer and provider. It is much quicker to replace the certificate with a token shared just between the consumer and provider. Another situation is to provide customized service.

**14. What the constraints introduced by the SOAP web services?**

- A SOAP web service introduces the following constraints:
- Except for binary data attachment, messages must be carried by SOAP.
- The description of a service must be in WSDL.
-

**15. Write on Tags and elements?**

XML tags begin with the less-than character ("<") and end with the greater-than character (">"). You use tags to mark the start and end of **elements**, which are the logical units of information in an XML document, an element consists of a **start tag**, possibly followed by text and other complete elements, followed by an **end tag**.

**16. What are attribute name and attribute value?**

Every attribute assignment consists of two parts: the **attribute name** (for example, *href*), and the **attribute value** (for example, http://www.yahoo.com/). There are a few rules to remember about XML attributes:

1. Attribute names in XML (unlike HTML) are case sensitive: *HREF* and *href* refer to two different XML attributes.
2. You may not provide two values for the same attribute in the same start tag. The following example is not well-formed because the *b* attribute is specified twice:

**17. What are the uses of XML?**

**XML** is used in many aspects of web development, often to simplify data storage and sharing.

**18. What are the various features of XML?**

- Security
- Portability
- Scalability
- Reliability

**19. Different between XML and HTML**

1. XML is not a replacement for HTML.
2. XML and HTML were designed with different goals:
3. XML was designed to transport and store data, with focus on what data is.
    1. HTML was designed to display data, with focus on how data looks.
    2. HTML is about displaying information, while XML is about carrying information.

**20. What are the three waves for XML development?**

- Vertical Industry Vocabularies
- Horizontal Industry Applications
- Protocols

**21. List out the advantages of XML.**

- XML files are human - readable
- Widespread industry support
- Relational Databases
- XML support technologies
- More meaningful searches
- Development of flexible web applications
- Data integration from disparate sources
- Local computation and manipulation of data
- Multiple views of the data
- Granular updates

**22. List out the XML structure.**

- Physical structure
- Logical structure

**23. What is physical structure?**

The physical structure consists of the contents used in an XML document. It holds the actual data to be represented in an XML document. This actual data storage can be called as Entities. These entities are identified by a unique name and may be part of the XML document or external to the document.

An entity is declared in the XML declaration part and referenced in the document element. Once declared in the DTD, an entity can be used anywhere.

**24. List out the Physical structure.**

- Parsed Entity
- Unparsed Entity
- Entity Reference
- Predefines Entities
- Internal and External Entities
- XML Syntax
- Attributes

**25. What is XML declaration?**

It identifies the version of the XML specification to which the document conforms.

**Example:**

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

An XML declaration can also include an

· Encoding Declaration

· Stand-alone Document Declaration

**26. What is Encoding?**

- The encoding declaration decides the encoding scheme. The encoding schemes available are
- UTF-8 and EUC-JP.
- The coding schemes map to different character formats or languages.

**27. What is standalone declaration?**

- The stand-alone document declaration identifies whether any markup declarations exits that are external to the document.
- This declaration can take in values of yes or no.

**28. Define Document Type Declaration**

· The document type declaration consists of the markup codes or the DTD according to which the XML document has to be written.

· The document type declaration can also point to an external file that contains the DTD. The document type declaration follows the XML declaration.

**Example:**

<?xml version="1.0"?>

<!DOCTYPE lib SYSTEM "lib.dtd">

**29. List out the various logical structure of an XML document.**

The various logical structures of an XML document are:

· Elements

· Attributes

· Entities

**30. Define Elements**

Element are the primary means for describing data in XML. The rules for composing elements are

- Flexible
- Allowing different combinations of text content, attributes and other elements.

**UNIT-II**

1. **What is structured information?**

Structured information contains both content (words, pictures, etc.) and some indication of what role that content plays (for example, content in a section heading has a different meaning from content in a footnote, which means something different than content in a figure caption or content in a database table, etc.). Almost all documents have some structure

2. **Define Name space**

An **XML namespace** is identified by a URI reference; element and attribute names may be placed in an XML namespace using the mechanisms described in this specification.

3. **Define expanded name space**

E**xpanded name** is a pair consisting of a namespace name and a local name. Definition: For a name *N* in a namespace identified by a URI *I*, the **namespace name** is *I*. For a name *N* that is not in a namespace, the **namespace name** has no value. Definition: In either case the **local name** is*N*. It is this combination of the universally managed IRI namespace with the vocabulary\'s local names that is effective in avoiding name clashes.

4. **Define qualified name space**

A **qualified name** is a name subject to namespace interpretation. ] In documents conforming to this specification, element and attribute names appear as qualified names. Syntactically, they are either prefixed names or unprefixed names

5. **Define Name space prefix**

If the attribute name matches Prefixed Att Name, then the NC Name gives the namespace prefix, used to associate element and attribute names with the namespace name in the attribute value in the scope of the element to which the declaration is attached. In such declarations, the name space name may not be empty.

6. **Write on Declaring name space**

A namespace (or more precisely, a namespace binding) is **declared** using a family of reserved attributes. Such an attribute\'s name must either be **xmlns** or begin **xmlns:**. These attributes, like any other XML attributes, may be provided directly or by default.

7. **Define XML schema**

An XML Schema consists of components such as type definitions and element declarations. These can be used to assess the validity of well-formed element and attribute information items (as defined in [XML-Infoset]), and furthermore may specify augmentations to those items and their descendants.

**8. Define Schema component**

Schema component is the generic term for the building blocks that comprise the abstract data model of the schema. [Definition:] An **XML Schema** is a set of schema components.

9. **What is schema validity assessment?**

Schema-validity assessment has two aspects:

- Determining local schema-validity, that is whether an element or attribute information item satisfies the constraints embodied in the relevant components of an XML Schema;
- Synthesizing an overall validation outcome for the item, combining local schema-validity with the results of schema-validity assessments of its descendants, if any, and adding appropriate augmentations to the infoset to record this outcome.

**10. Define valid**

The word **valid** and its derivatives are used to refer to clause 1 above, the determination of local schema-validity

**11. Define assessment**

The word **assessment** is used to refer to the overall process of local validation, schema validity assessment and infoset augmentation

**12. Define primary component**

The primary components, which may must have names are as follows:

· Simple type definitions
· Complex type definitions
· Attribute declarations
· Element declarations

13. **Define secondary component**

The secondary components, which must have names, are as follows:

· Attribute group definitions
· Identity-constraint definitions
· Model group definitions
· Notation declarations

14. **Define Target Namespace**

Several kinds of component have a **target namespace**, which is either ·absent· or a namespace name, also as defined by [XML-Namespaces]. The ·target namespace· serves to identify the name space within which the association between the component and its name exists. In the case of declarations, this in turn determines the namespace name of, for example, the element information items it may validate.

15. **Define Helper component**

The helper components provide small parts of other components;

- Annotations
- Model groups
- Particles
- Wildcards
- Attribute Uses

16. **What is XSL?**

Extensible Style sheet Language (XSL) provides facilities to access and manipulate the data in XML documents. XSL is itself an XML dialect and provides two distinct and useful mechanisms for handling and manipulating XML documents. Many of the same constructs are shared between the two mechanisms, but each plays a distinct role. One is concerned with formatting data, and the other is concerned with data transformation. When XSL is used as a formatting language, the style sheets consist of formatting objects that prepare an XML document for presentation, usually in a browser.

17. **What are the different template patterns?**

- Match template patterns
- XSLT Patterns
-

18. **Write on Template rule body**

A template rule body can consist of:

- More detailed selection or match conditions and other logic
- A specific type of action or actions to be performed
- Text that becomes part of the results along with the selected target XML document\'s content

**19. What are the various data types available in XML SCHEMA?**

The various data types in XML schema are

§ String

§ Integer

§ Float etc.

**20. List out the type of Namespace.**

- Default Namespace
- Explicit Namespace
-

**21. Define Infoset**

The XML Infoset is an abstract Data Model describing the information available from an XML document. For many applications, this way of looking at an XML document is more useful than having to analyze and interpret XML syntax. DOM describes an API through which the information in an XML Infoset

(i.e., the information available from a specific XML document) can be accessed from different programming languages.

**22. What is mean by RDF?**

- RDF stands for **R**esource **D**escription **F**ramework
- RDF is a framework for describing resources on the web
- RDF provides a model for data, and a syntax so that independent parties can exchange and use it
- RDF is designed to be read and understood by computers
- RDF is not designed for being displayed to people
- RDF is written in XML
- RDF is a part of the W3C\'s Semantic Web Activity
- RDF is a W3C Recommendation
-

**23. Define XLINK.**

XLink defines a standard way of creating hyperlinks in XML documents. XPointer allows the hyperlinks to point to more specific parts (fragments) in the XML document.

· XLink is short for the XML Linking Language
· XLink is a language for creating hyperlinks in XML documents
· XLink is similar to HTML links - but it is a lot more powerful
· ANY element in an XML document can behave as an XLink
· XLink supports simple links (like HTML) and extended links (for linking multiple resources together)
· With XLink, the links can be defined outside of the linked files
· XLink is a W3C Recommendation

**24. Define Voice XML.**

**Voice XML** (**VXML**) is the W3C\'s standard XML format for specifying interactive voice dialogues between a human and a computer. It allows voice applications to be developed and deployed in an analogous way to HTML for visual applications. Just as HTML documents are interpreted by a visual web browser, VoiceXML documents are interpreted by a voice browser. A common architecture is to deploy banks of voice browsers attached to the public switched telephone network (PSTN) so that users can use a telephone to interact with voice applications

**25. List out the three part of XSL.**

· XSLT - a language for transforming XML documents
· XPath - a language for navigating in XML documents
· XSL-FO - a language for formatting XML documents

**26. What is mean by Uniform Resource Identifier (URI)?**

A **Uniform Resource Identifier** (URI) is a string of characters which identifies an Internet Resource. The most common URI is the **Uniform Resource Locator** (URL) which identifies an Internet domain address. Another, not so common type of URI is the **Universal Resource Name**(URN). In our examples we will only use URLs.

**27. List out the Schema Components.**

· Primary Components

· Secondary Components
· Simple type definitions
· Complex type definitions
· Attribute declarations
· Element declarations
· Attribute group definitions
· Identity-constraint definitions
· Model group definitions & Notation declarations

**28. Define Schema-validity.**

Schema-validity assessment has two aspects:

· Determining local schema-validity, that is whether an element or attribute information item satisfies the constraints embodied in the relevant components of an XML Schema;

· Synthesizing an overall validation outcome for the item, combining local schema-validity with the results of schema-validity assessments of its descendants, if any, and adding appropriate augmentations to the infoset to record this outcome.

**29. Define Schema.**

XML Schemas are extensible, because they are written in XML. With an extensible Schema definition you can:

· Reuse your Schema in other Schemas
· Create your own data types derived from the standard types
· Reference multiple schemas in the same document

**30. List out the type of DTD**

· Internal Subset DTD & External Subset DTD.

**UNIT-III**

1. **What is Service Oriented Architecture?**
Service oriented architecture is essentially a collection of services. These services communicate with each other. The communication can involve either simple data passing or it could involve two or more services coordinating some activity.

2. **Define Contemporary SOA.**
Contemporary SOA represents an architecture that promotes service orientation through the use of web services.

3. **List out some characteristics of Contemporary SOA.**
Some of the characteristics of contemporary SOA are:-
   i. Contemporary SOA is at the core of the service oriented platform.
   ii. Contemporary SOA increases quality of service.
   iii. Contemporary SOA is fundamentally autonomous.
   iv. Contemporary SOA is based on open standards.
   v. Contemporary SOA supports vendor diversity.
   vi. Contemporary SOA fosters intrinsic interoperability.

vii. Contemporary SOA promotes discovery.
viii. Contemporary SOA promotes federation.
ix. Contemporary SOA promotes architectural composability.
x. Contemporary SOA fosters inherent reusability.

**4. What are the benefits of SOA?**

The benefits of SOA are:

i. Improved integration and intrinsic interoperability
ii. Inherent reuse
iii. Streamlined architectures and solutions
iv. Leveraging the legacy investment
v. Establishing standardized XML data representation
vi. Focused investment on communications infrastructure
vii. "Best-of-breed" alternatives
viii. Organizational agility

**5. What are the common pitfalls of adopting SOA?**

The common pitfalls of adopting SOA are:

i. Building service oriented architectures like traditional distributed architectures
ii. Not standardizing SOA
iii. Not creating a transition plan
iv. Not starting with an XML foundation architecture
v. Not understanding SOA performance requirements
vi. Not understanding web services security
vii. Not keeping in touch with product platforms and standards development

**6. What are the requirements is needed to fill QoS gaps between contemporary and Primitive SOA?**

Contemporary SOA is striving to fill the QoS gaps of the primitive SOA model with the following requirements:

i. Security (contents and access)
ii. Reliability (message guaranteed delivery)
iii. Appropriate performance
iv. Protecting business integrity
v. Executing exception logic in case of failure

**7. What is Autonomous Principle?**

Autonomous Principle represents the ability of a service to carry out its logic independently of outside influences.

**8. List out the different levels of Autonomy.**

Different levels of Autonomy are:

i. Runtime autonomy
ii. Design time autonomy

**9. What is Runtime Autonomy?**

Runtime Autonomy represents the amount of control a service has over its execution environment at runtime.

**10. What is Design time autonomy?**

Design time autonomy represents the amount of governance control a service owner has over the service design.

**11. Expand UDDI.**

UDDI stands for Universal Description Discovery and Integration.

**12. What are the design characteristics required to facilitate interoperability in contemporary SOA?**

The design characteristics required to facilitate interoperability are:

i. Standardization
ii. Scalability
iii. Behavioral predictability
iv. Reliability

**13. How is loose coupling concept achieved in SOA?**

The loose coupling concept is achieved by implementing standardized service abstraction layers when service orientation principles are applied to both business modeling and technical design.

**14. What is referred as Organizational Agility?**

Organizational Agility refers to efficiency with which an organization can respond to change.

**15. What is Architecture?**

Architecture refers a systematic arrangement of computerized automation technological solutions.

**16. What is application architecture?**

Application architecture is a template for all others which specifically explained the technology, boundaries, rules, limitations, and design characteristics that apply to all solutions based on this template.

**17. What is enterprise architecture?**

Enterprise architecture is a creation of master specification when numerous, disparate and integrate application architectures exist within an organization.

**18. What is Single-tier client-server architecture?**

Single-tier client-server architecture is an environment in which bulky mainframe back-ends server served the thin clients.

**19. List out the primary characteristics of the two tier client server architecture?**

The primary characteristics of the two tier client server architectures is given below which is compared to SOA

i. Application logic
ii. Application processing
iii. Technology
iv. Security
v. Administration

**20. What is multi-tier client-server architectures?**

Multi-tier architecture (often referred to as n-tier architecture) is a client-server architecture in which the presentation, the application processing, and the data management are logically separate processes.

**21. List out the types of communications of mainframe systems?**

The different types of communications of mainframe systems are:

i. Synchronous communication
ii. Asynchronous communication

**22. Define synchronous communication.**

Synchronous communication allows the client and server to wait for each other to transfer the message. That is, the client will not continue until the server has received the message.

**23. Define asynchronous communication.**

Asynchronous communication allows the server to continuously receive messages from the client without waiting for the server to respond.

**24. List out the types of service autonomy?**

The different types of service autonomy are:

i. Service-level autonomy
ii. Pure autonomy
iii.

**25. What are the key benefits of service reuse?**

The key benefits of service reuse are:

i. Accommodate future requirements with less development effort
ii. Reduce the need for creating wrapper services
iii. Reduction of cost by not just avoiding duplication of code
iv. Reducing risks by reusing well-tested code and runtime environments

**26. State Separation of concerns.**

"Separation of concerns" is an established software engineering theory based on the idea of breaking down a large problem into a series of individual concerns.

**27. What are the parts of automation logic?**

The four identified parts of automation logic related to different sized units of logic as follows:

i. messages = units of communication
ii. operations = units of work
iii. services = units of processing logic (collections of units of work)
iv. processes = units of automation logic (coordinated aggregation units of work)

**28. What are the issues that are raised in the client-server and the distributed Internet architecture?**

The issues that are raised in the client-server and the distributed Internet architecture comparisons are discussed in a comparison between multi-tier client-server and SOA.

i. Application logic
ii. Application processing
iii. Technology
iv. Security
v. Administration

**29. What is the use of RPC?**

Client-server remote procedure call (RPC) connection is used for remote communication between components residing on client workstations and servers.

**30. Write down the advantage of RPC?**

Advantages of RPC are:

i. Better load balancing:
   More evenly distributed processing (e.g., application logic distributed between several servers)
ii. More scalable:
   Only servers experiencing high demand need be upgraded

## UNIT-IV

**1. Define Web services**

A Web service (also Web Service, Web service) is defined by the W3C as a software system designed to support interoperable machine-to-machine .Web Services is the umbrella term of group of loosely related Web-based resources and components that may be used by other Web.

**2. Define SOA**

A service-oriented architecture is essentially a collection of services. These services communicate with each other. The communication can involve either simple data passing or it could involve two or more services coordinating some activity. Some means of connecting services to each other is needed. Service-oriented architectures are not a new thing. The first service-oriented architecture for many people in the past was with the use DCOM or Object Request Brokers (ORBs) based on the CORBA specification. For more on DCOM and CORBA.

**3. List out the characteristic of Web services.**

· XML based everywhere
· Message-based
· Programming language independent
· Could be dynamically located
· Could be dynamically assembled or aggregated
· Accessed over the internet
· Loosely coupled
· Based on industry standards
· Are platform neutral
· Are accessible in a standard way
· Are accessible in an interoperable way
· Use simple and ubiquitous plumbing
· Are relatively cheap
· Simplify enterprise integration

**4. What are the uses of Web services?**

· Interoperable – Connect across heterogeneous networks using ubiquitous web-based standards
· Economical – Recycle components, no installation and tight integration of software
· Automatic – No human intervention required even for highly complex transactions
· Accessible – Legacy assets & internal apps are exposed and accessible on the web.
· Available – Services on any device, anywhere, anytime
· Scalable – No limits on scope of applications and amount of heterogeneous applications

**5. What are the three roles of Web service?**

The three role of web service are
· Client
· Service
· Broker.

**6. Define client**

A client is any computer that accesses functions from one or more other computing nodes on the network. Typical clients include desktop computers, Web browsers, Java applets, and mobile devices. A client process makes a request for a computing service and receives results for that request.

**7. Define Service**

A service is a computing process that receives and responds to requests and returns a set of results.

**8. Define Broker**

A broker is essentially a service metadata portal for registering and discovering services. Any network client can search the portal for an appropriate service.

**9. What are the standard protocols used in web service?**

The standard protocols used in web service

· WSDL

· UDDL

**10. Define WSDL**

WSDL stands for Web Services Description Language. WSDL is a document written in XML. The document describes a Web service. It specifies the location of the service and the operations (or methods) the service exposes.

**11. Define UDDI**

Universal Description, Discovery and Integration (UDDI) is a platform-independent, XML- based registry for businesses worldwide to list themselves on the Internet. UDDI is an open industry initiative, sponsored by OASIS, enabling businesses to publish service listings and discover each other and define how the services or software applications interact over the Internet.

**12. What are the three components used in UDDI?**

· White Pages — address, contact, and known identifiers;

· Yellow Pages — industrial categorizations based on standard taxonomies

· Green Pages — technical information about services exposed by the business

**13. List out the UDDI register**

· Public Register

· Private Register

**14. What are the major elements used in WSDL?**

The major elements used in WSDL are

· PortType

· Message

· Types

· Binding

**15. Define the structure of WSDL**

<definitions>
<types>
definition of types.....
</types>
<message>
definition of a message....
</message>
<portType>
definition of a port.....

</portType>
<binding>
definition of a binding....
</binding>
</definitions>

**16. Define WSDL Messages**

The <message> element defines the data elements of an operation. Each message can consist of one or more parts. The parts can be compared to the parameters of a function call in a traditional programming language.

**17. Define WSDL Type**

The <types> element defines the data type that are used by the web service. For maximum platform neutrality, WSDL uses XML Schema syntax to define data types.

**18. Define WSDL Binding**

The <binding> element defines the message format and protocol details for each port.

**19. Define ebXML**

Electronic Business using eXtensible Markup Language, commonly known as e-business XML, or ebXML is a family of XML based standards sponsored by OASIS and UN/CEFACT whose mission is to provide an open, XML-based infrastructure that enables the global use of electronic business information in an interoperable, secure, and consistent manner by all trading partners.

**20. List out the layer of data specification in ebXML**

· Business processes,
· Collaboration protocol agreements,
· Core data components,
· Messaging & Registries and repositories

**21. What is XPointer?**

XPointer is set of recommendations developed by the W3C. The core recommendations are the XPointer Framework which provides an extensible addressing behavior for fragment identifiers in XML media types. XPointer gains its extensibility through the XPointer Framework, which identifies the syntax and processing architecture for XPointer expressions and through an extensible set of XPointer addressing schemes. These schemes, e.g., element() or xpointer(), are actually QNames. The xmlns() scheme makes it possible for an XPointer to declare namespace bindings and thereby use third-party schemes as readily as W3C defined XPointer schemes.

**22. Where can qualified names appear?**

Qualified names can appear anywhere an element type or attribute name can appear: in start and end tags, as the document element type, and in element type and attribute declarations in the DTD

**23. Define .Net**

Microsoft .NET initiative has its origins in the increasing importance of the Web in almost all areas of application development. Previous development tools, exemplified by Visual Studio version

6.0, were designed for the needs of a decade ago, when the ruling paradigm was applications that were stand-alone or were distributed over a local area network (LAN). As the need for Web-related capabilities grew, ad hoc solutions were crafted as enhancements to existing tools. Because the Web capabilities were not built into the development tools from the beginning, however, there were inevitable problems with deployment, maintenance, and efficiency.

**24. Define .NET framework**

The .NET Framework provides a comprehensive set of classes that are designed for just about any programming task you can imagine. From the very beginning, the Framework was designed to integrate Web-related programming functionality.

**25. What is the .NET architecture includes several technology components.**

· Development Tools
· Specialized servers
· Web services
· Devices

**26. Define J2EE**

J2EE is a standard for building robust enterprise applications based on an evolving vision of application-server technology centered on the java programming language.

**27. What are the technical architecture consist in ebXML technology?**

· Messaging
· Business processes
· Trading partner profile and agreements
· Registers and repositories
· Core components

**28. Define Web Services Caveats**

1. Different implementations may not work together
2. SOAP messages on port 80 may bypass firewalls
3. Transactions must be specified outside the web services framework
4. Change Management is not addresses

**29. List out the Web Services Caveats**

· Maturity
· Security
· Transactions
· Configuration Management

**30. List out the .NET Pros**

· It offers multiple language support.
· It has a rich set of libraries, a la JVM.
· It\'s open-standard friendly (e.g., HTTP and XML) -- it may even become a standard itself.
· Its code is compiled natively, regardless of language or deployment (Web or desktop).

## UNIT-V

**1. What are the basic security requirements for e-business?**

The basic security requirements for e-business are

· Confidentiality

· Authentication

· Data integrity

2. **Define Confidentiality**

Ensuring that information is not made available or disclosed to unauthorized individuals.

**3. Define Authentication**

· Ability to determine that the message really comes from the listed sender.

· Non repudiation-preventing the origination of the document from denying having sent it.

**4. Define Integrity**

Ensuring that information is not tampered in transit.

**5. List out the type of cryptography**

· Single key cryptography

· Public key cryptography

**6. What do you mean by Single key Cryptography?**

· A single key is used for both encryption and decryption.

· The key must be known to both sender and receiver

· The difficulty in this approach is the distribution of the key

· Example DES-Data Encryption Standard

· Single key systems are effective for secure communication between ATM machines and server

· However it does not scale upto web, where ecommerce depends on individuals just showing to do business.

**7. What do you mean by Public key Cryptographic?**

· Enables secure communication without having to exchange secret key

· It uses mathematical formula to generate two separate, but related key

· One key is open to public view and the other private, known only to one individual.

**8. Define Signatures**

Defines both syntax and rules for processing XML digital signature. It defines a series of XML elements for describing details of the signature.

· Signed info-holds the information that is actually

· Canonicalization method-algorithm used to canonicalize the signed info.

· Signature method-algorithm used to convert the canonicalized signed info into the signature value

· Combination of digest algorithm

· Key dependent algorithm
· Reference –includes the method used to compute the digital hash and the identified data object the signature is later checked via reference and signature validation.

**9. What are the key to be use in validate the signature?**
· Transforms
· Digest method
· Digest value

**10. Define Transforms**
Optional ordered list of processing steps applied to the resources content before the digest was computed.

**11. Define Digest method**
Algorithm applied to data after transforms is applied to yield the digest Value.

**12. Define Digest value**
Holds the value computed based on the data being signed.
· Encoding scheme-are used to represent characters
· Line breaks
· Attribute values are normalized
· Double quotes for attribute values
· Special character in attribute values and character content
· Entity references
· Default attributes
· XML and DTD declarations
· White space outside document element
· White space in start and end elements
· Empty elements
· Namespace declaration
· Ordering of ns declaration and attributes

**14. List out the XML security technology.**
· XML digital signature
· XML encryption
· XML key management services

**15. What is XML encryption?**
· An important issue not addressed by SSL is encrypting part of the data being exchanged
· Enables to overcome it by enabling encrypting part of the data.
· It can also handle both XML and non XML data
· Does not support encryption of attributes sample file to be encrypted

**16. List out the steps in XML encryption.**
· selecting the XML to be encrypted
· converting into canonical form
· encrypting the resulting canonical form with public key

· sending the encrypted XML

**17. Define XML Digital Signature.**

Xml digital signature defines both syntax and rules for processing. It defines a series of XML elements for describing details of the signature.

· Signed info-holds the information that is actually
· Canonicalization method-algorithm used to canonicalize the signed info.
· Signature method-algorithm used to convert the canonicalized signed info into the signature value
· Combination of digest algorithm
· Key dependent algorithm
· Reference –includes the method used to compute the digital hash and the identified data object the signature is later checked via reference and signature validation
· Key info-indicates the key to be used to validate the signature

XKMS specs are made up of two specs.

· XKRSS-reg.service spec-registration of public key
· XKISS-info.service spec-retrieval of information based on key information

**18. Define XKMS**

Allows management of PK1 by abstracting the complexity of managing the Pk1 from client applications to a trusted third party. Trusted third party hosts the XKMS service while providing a PK1 interface to Client application. This allows a client application to access PK1 features, thereby reducing the client applications complexity.

**19. List out the specification of XKMS.**

· XKRSS
· XKISS

**20. What are the three pillars of secure E-Commerce?**

· Confidentiality
· Authentication
· Data integrity

Unit-I

Define XML
List out the rules of xml structure.
Identify what is well formed and valid document.