

**REGULATION 2013**  
**IT6503 – WEB PROGRAMMING**  
**QUESTION BANK**

**UNIT – I**  
**PART – A**

**1.What is Web Browser**

A web browser displays a web document and enables users to access web documents.

When the user clicks a hyperlink, the browser initiates a message to a web server.

This message requests the server to retrieve the requested information and send it back to the web browser through the telecommunications network.

**2.What Web Server**

1. This is a program that waits patiently for the browser to request a web page.

2. The server looks for the requested information, retrieves it and sends it to the browser or sends an error message if the file is not found.

**3.Define Web page, Web site &Web browser?**

A Web page is a document created using HTML.

A Web site is a collection of related pages.web pages and sites can be stored on the hard drive of a local computer or a web server.

A Web browser is a program that displays the web pages it retrieves.

**4.What is HTML (HyperText Markup Language)**

It is the universal language understood by all WWW (World Wide Web) clients.

An HTML document (program) is ASCII text with embedded instructions (markups) which affect the way the text is displayed.

The basic model for HTML execution is to fetch a document by its name (e.g. URL), interpret the HTML and display the document, possibly fetching additional HTML documents in the process.

It can accept user input and/or cause additional HTML documents to be fetched by URL.

Providing safety, platform independence, and the ability to interact with a variety of formats, protocols, tools, and languages makes it a universal language.

**What is Java and Java Script?**

Java is a language for sending applets over the web, so that the computer can execute them.

JavaScript is a language that allows HTML to embed small programs called scripts in web pages. The main purpose of applets and scripts is to speed up web page interactivity.

**5.List down font characteristics permitted in style sheets.**

font-family

font-size

font-weight

font-style

font-variant

**6.What are Style Sheets?**

Style sheets are collections of style information that are applied to plain text. Style information includes font attributes such as type size, special effects (bold,italic,underline),color and alignment. Style sheets also provide broader formatting instructions by specifying values for quantities such as line spacing and left and right margins.

## 7. List down the ways of including style information in a document.

External Styles -Style information is read from a separate file that is specified in the <LINK> tag

Embedded Styles -Style information is defined in the document head using the <STYLE> and </STYLE> tags.

Inline Styles -Style information is placed inside an HTML tag and applies to all content between that tag and its companion closing tag.

## 8. Define cascading.

Cascading refers to a certain set of rules that browsers use, in cascading order, to determine how to use the style information. Such a set of rules is useful in the event of conflicting style information because the rules would give the browser a way to determine which style is given precedence.

## 9. What are the style precedence rules when using multiple approaches?

Inline styles override both linked style sheets and style information stored in the document head with <STYLE> tag. Styles defined in the document head override linked style sheets. Linked style sheets override browser defaults.

## 10. List the goals of SGML.

To manage the flow of millions of pages.

For structuring information exchange

For modeling inter-document linkages

For managing information flows between departments and weapons systems

## 11. What is the role of server?

The server

Manages application tasks

Handles storage

Handles security

Provides scalability

Handles accounting and distribution

## 12. What are the necessities of using HTML forms?

Gathering user information

Conducting Surveys

Interactive services

## 13. What is the usage of CSS?

A simple mechanism for adding style (such as fonts, colors, or spacing) to web documents. Multiple levels of CSS can be used to allow selective overriding of styles.

## 14. What is DOM?

DOM is a W3C supported application program interface (API) that provides a platform and language neutral interface to allow developers to programmatically access and modify the content and structure of a document such as HTML or XML.

## 15. What is HTML?

HTML is a language for describing web pages.

HTML stands for Hyper Text Markup Language

HTML is not a programming language, it is a markup language. A markup language is a set of markup tags

HTML uses markup tags to describe web pages

## 16. How to implement the HTML Links

HTML links are defined with the <a> tag.

```
<a href="http://www.w3schools.com">This is a link</a>
```

## 17. Define HTML Tables

Tables are defined with the <table> tag. A table is divided into rows (with the <tr> tag), and each row is divided into data cells (with the <td> tag). td stands for "table data," and holds the content of a data cell. A <td> tag can contain text, links, images, lists, forms, other tables, etc.

## 18. How to declare the HTML Forms

HTML forms are used to pass data to a server. A form can contain input elements like text fields, checkboxes, radio-buttons, submit buttons and more. A form can also contain select lists, textarea, fieldset, legend, and label elements. The <form> tag is used to create an HTML form: <form> .input elements .</form>

## 19. Compare SGML-based versus XML-based HTML

The XML-based specification is usually called XHTML to distinguish it clearly from the more traditional definition. However, the root element name continues to be 'html' even in the XHTML-specified HTML. The W3C intended XHTML 1.0 to be identical to HTML 4.01 except where limitations of XML over the more complex SGML require workarounds. Because XHTML and HTML are closely related, they are sometimes documented in parallel.

## 20. What is CSS?

Cascading Style Sheets (CSS) is a slightly misleading term, since a website might have only one CSS file (style sheet), or the CSS might be embedded within an HTML file. It is better to think of CSS as a technology (in the singular). CSS is comprised of statements that control the styling of HTML documents. Simply put, an HTML document should convey content. A CSS document should control the styling of that content.

## 21. What are the types of rules?

Type 1: Rules

Statement + statement block X {declaration; declaration;} X {property; value; property: value;}

div > p {font-size: 1em; color #333;}

Type 2: At-Rules at-keyword + identifier + declaration @import "subs.css";

## 22. What is selector and mention its types also

Selectors refer to elements in an HTML document tree

Type of Selector

Universal Selector

Class Selector

ID Selector and Descendant Selector

## 23. Mention all the text properties

Color Direction

Line-height Letter-spacing

Text-align Text-decoration

Text-indent Text-shadow

Text-transform Unicode-bidi

Vertical-align White-space

Word-spacing

## 24. What is the purpose of CSS Box Model and mention its parts also.

The CSS box model is essentially a box that wraps around HTML elements, and it consists of: margins, borders, padding, and the actual content.

The different parts are:

Margin

Border

Padding

Content

## 25. Define the term positioning and mention its types also.

The CSS positioning properties allow you to position an element. It can also place an element behind another, and specify what should happen when an element's content is too big . There are four different positioning methods.

- Static Positioning
- Fixed Positioning
- Relative Positioning
- Absolute Positioning

## 26. What is JavaScript?

JavaScript was designed to add interactivity to HTML pages

JavaScript is a scripting language

A scripting language is a lightweight programming language

JavaScript consists of lines of executable computer code

JavaScript is usually embedded directly into HTML pages

## 26. Are Java and JavaScript the Same?

NO!. Java and JavaScript are two completely different languages in both concept and design. Java (developed by Sun Microsystems) is a powerful and much more complex programming language - in the same category as C and C++.

## 27. How to Put a JavaScript Into an HTML Page

```
<html>
<body>
<script type="text/javascript">
document.write("Hello World!");
</script>
</body>
</html>
```

## 28. What are literals?

Literal values are the ones you type into mathematical or string expressions. For example 23 (an integer), 12.32E23 (a floating point), or 'flopsy the Hamster' (a string). String literals can be enclosed by either single or double quotes. For example: 'literal string' , "literal string", 'literal string with "double quotes" inside'

## PART – B

### 1. Explain the Fundamental HTML Elements

- What is HTML?
- HTML Tags
- HTML Documents
- HTML Elements
- HTML Element Syntax
- Nested HTML Elements

- HTML Document Example
- EMPTY HTML ELEMENTS
- HTML Attributes
- HTML HEADINGS
- HTML Formatting Tags

### 2. Write a short notes on HTML Tables

- HTML Tables and the Border
- Attribute HTML Table Headers

### **3. Write Short notes on HTML LISTS and HTML**

#### **Forms**

HTML LISTS

HTML Unordered Lists

HTML Ordered Lists

HTML Definition Lists

HTML List Tags

HTML FORMS

HTML Forms - The Input Element

Text Fields

Password Field

Radio Buttons

Checkboxes

Checkboxes

Submit Button

HTML Form tags

### **4. Explain about XML and how to Create HTML Documents with the help of XML**

Elements

Element examples

Paragraphs:

Comments:

Attributes

Character and entity references

Data types

Document type declaration

Semantic HTML

Delivery

### **5. Write short notes on CSS.**

Features

Core Syntax At-Rules

CSS1 Selectors

### **6. Explain concept of Cascading and inheritance Concepts in CSS**

Specificity

Rule Cascade

Style Inheritance

### **7. Discuss the text properties of CSS with Suitable Example.**

CSS Font Families Font

Family

Font Style Font Size

Set Font Size With Pixels Set Font

Size With Em All CSS Font

Properties Text Formatting and color

All CSS Text Properties

### **8. Explain the elements of JavaScript with relevant examples**

Basic syntax

Javascript variables and datatypes

Javascript statements

Javascript operators

Javascript literal values

Javascript functions

### **9. Explain about the use of Objects in JavaScript.**

Creating a Custom JavaScript Object Creating  
and Using Object Instances Extending Objects

Built-in objects

JavaScript Boolean Object JavaScript

RegExp Object

### **10. How HTML elements can be changed using JavaScript, the HTML DOM and events.**

Modifying Element Style Change an

HTML Element

Change the Text of an HTML Element - innerHTML Change an

HTML Element Using Events

Change the Text of an Element - with a Function Using  
the Style Object  
Change the font and color of an Element

## UNIT – II PART – A

### 1. Define Java.

Java is an object-oriented language similar to C++, but simplified to eliminate language features that cause common programming errors. Java source code files (files with a .java extension) are compiled into a format called bytecode (files with a .class extension), which can then be executed by a Java interpreter.

### 2. What are the Java Features?

- Platform Independence
- Object Oriented
- Compiler/Interpreter Combo
- Robust
- Automatic Memory Management
- Security
- Dynamic Binding
- Good Performance
- Threading
- Built-in Networking

### 3. What is Data Type ?

Data Types: It has Eight Simple Data types which can be put in four groups, they are:

Integers: **byte**, **short**, **int**, and **long** for whole valued signed numbers.

Floating-point numbers: **float**, and **double** for fractional precision.

Characters: **char** for representing symbols in character set.

Boolean: **boolean** a special type for representing true/false.

### 4. Define Variable

Variables: The basic unit of storage in Java Program. And is defined by the combination of *an identifier, a type, and an optional initializer*. The syntax for declaring a variable is: *type identifier [= value] [, identifier [= value]]...*; e.g.: `int a = 5, b=6;`

### 5. Type Conversions and Casting:

a. Java's Automatic Conversions:

- i. The Two types are compatible.
- ii. The destination type is larger then the source

type. b. Casting Incompatible Types:

- i. Also called as *narrowing conversions*, since we are explicitly making the value narrower so that it will fit into the target type.
- ii. A *cast* is simply a explicit type conversion. The general form is (*target type*) value

e.g. `int a; byte b; b = (byte) a;`

### 6. Define Arrays

a. An array is a group of variables of the same data type and referred to by a common name. An array is contiguous block of memory locations referred by a common name.

E.g. `char[] s;`

`Class[] Obj;`

b. Like all objects we use the *new* keyword to create an array.

`s = new char[10];`

c. Types of Arrays:

i. One Dimensional Array.

`type array_name [];` //type is the datatype of the array.

ii. Multi – Dimensional Array.

`type array_name = new type [rows] [cols];`

### 7. What is Expression

In Java, arithmetic, boolean, and String expressions are written in conventional mathematical infix notation, adapted to the standard computer character set (called ASCII).

## 8. What is Operator

Java provides a rich operator environment. And it can be classified into four groups as:

Arithmetic: Used in mathematical expressions. They are: +, -, \*, /.

Bitwise: It operates on individual bits of integer values. They are: &, |, ~,

Relational: Compares two values and determines relationship between them. ==, !=, <, >, <=, and >=

Logical: It is an easy way to handle multiple conditions. They are: &&, || and, !.

## 9. What are Control Structures

Statements that support repetition and conditional execution are control statements

They are of two type: They are:

Branching Statements: - They are *If* and *Switch* Statements.

Looping Statements: - They are *for*, *while*, and *do-while* loops.

## 10. Define Classes

Java classes contain fields and methods. A field is like a C++ data member, and a method is like a C++ member function.

Each field and method has an access level:

- private: accessible only in this class
- (package): accessible only in this package
- protected: accessible only in this package and in all subclasses of this class
- public: accessible everywhere this class is available

Example:       Class MyClass{  
                  int RollNo; // Member Variable  
                  String Name; // Member Variable  
                  void getDetails(); // Member Function  
                  void dispDetails(); // Member Function }

## 11. Define Object

Objects are key to understanding object-oriented technology.

Object is an instance (or instantiation) of a class.

Three properties characterize objects:

- Identity: the property of an object that distinguishes it from other objects
- State: describes the data stored in the object
- Behavior: describes the methods in the object's interface by which the object can be used

Example:

```
public static void main(String args[])  
{ MyClass c1 = new MyClass(); //c1 is the Object of the Class MyClass  
  MyClass c2; // declare reference to Object  
  c2 = new MyClass(); //Allocate a MyClass Object  
}
```

## 12. Define Method

A method is a set of Java statements which can be included inside a Java class.

They are similar to functions or procedures in other programming languages.

The only required elements of a method declaration are the method's return type, name, a pair of parentheses, (), and a body between braces, {}.

The general form is

```
return-type method-name(parameters-list)  
{ // body of the method }
```

## 13. What is a Constructor?

Constructors have one purpose in life: to create an instance of a class. This can also be called creating an object.

Constructors and methods differ in three aspects of the signature: modifiers, return type, and name.

Like methods, constructors can have any of the access modifiers: public, protected, private, or none.

Constructors cannot be *abstract*, *final*, *native*, *static*, or *synchronized*.



#### 14. What is Garbage Collection?

The Java virtual machine's heap stores all objects created by a running Java application.

Garbage collection is the process of automatically freeing objects that are no longer referenced by the program.

It relieves programmers from the burden of freeing allocated memory.

In addition to freeing unreferenced objects, a garbage collector may also combat heap fragmentation.

Before an object is garbage collected, the runtime system calls its *finalize()* method.

It takes no arguments and returns no results. This method can be overridden to perform some tidying up tasks when an object is garbage collected.

#### 15. What is Method Overloading?

The concept of defining two or more methods within the same class that share the same name, as long as their parameter declarations are different is called as Method Overloading.

Each overloaded method must take a unique list of argument types.

When an overloaded method is called, java uses the type and/or number of arguments to decide which version of the overloaded method to actually call.

#### 16. What is Inheritance?

Inheritance is the capability of a class to use the properties and methods of another class while adding its own functionality.

It has the following advantages:

- you can customize and enhance working classes
- it is easier to reuse code
- you can take a more general class and modify to suit a particular situation

#### 17. What is Method Overriding?

Method Overriding is achieved when a subclass overrides non-static methods defined in the superclass, following which the new method implementation in the subclass that is executed.

#### 18. What is Abstract Class?

An abstract class is a class that is declared abstract—it may or may not include abstract methods.

Abstract classes cannot be instantiated, but they can be subclassed.

An abstract method is a method that is declared without an implementation (without braces, and followed by a semicolon), like this:

```
abstract void moveTo(double deltaX, double deltaY);
```

When an abstract class is subclassed, the subclass usually provides implementations for all of the abstract methods in its parent class. However, if it does not, the subclass must also be declared abstract.

#### 19. What is Package?

A package is a grouping of related types providing access protection and name space management.

Note that types refers to classes, interfaces, enumerations, and annotation types.

The types that are part of the Java platform are members of various packages that bundle classes by function: fundamental classes are in `java.lang`, classes for reading and writing (input and output) are in `java.io`, and so on.

To create a package, you choose a name for the package (naming conventions are discussed in the next section) and put a package statement with that name at the top of every source file that contains the types. E.g. `package graphics;`

#### 20. Define Interface

An interface is a group of related methods with empty bodies.

Implementing an interface allows a class to become more formal about the behavior it promises to provide.

Interfaces form a contract between the class and the outside world, and this contract is enforced at build time by the compiler.

If your class claims to implement an interface, all methods defined by that interface must appear in its source code before the class will successfully compile.

## 21. What is Exception Handling?

An exception is an event that occurs during the execution of a program that disrupts the normal flow of instructions.

When an error occurs within a method, the method creates an object and hands it off to the runtime system. The object, called an exception object, contains information about the error, including its type and the state of the program when the error occurred.

Creating an exception object and handing it to the runtime system is called *throwing an exception*.

The set of possible "somethings" to handle the exception is the ordered list of methods that had been called to get to the method where the error occurred. The list of methods is known as the *call stack*

The runtime system searches the call stack for a method that contains a block of code that can handle the exception. This block of code is called an *exception handler*.

## 22. What is Multi-Threading?

A thread executes a series of instructions. Every line of code that is executed is done so by a thread.

Some threads can run for the entire life of the applet, while others are alive for only a few milliseconds.

The class `java.lang.Thread` is used to create and control threads.

To create a thread, a new instance of this class must be created. However, the thread does not start running right away. `Thread.start()` must be called to actually make the thread run.

There are two ways to create a thread:

Extend the `Thread` class.

- With this technique the new class inherits from the class `Thread`. The thread can start running in the class's `run` method.

Implement the `Runnable` interface.

- This technique is probably more common than extending the `Thread` class. It is not necessary to define a new class to run the thread

## 23. Utility Packages:

The final Java package, `java.util`, contains a collection of utility classes.

The Utility Package of Java consist of the following components:

- Collections framework
- Legacy collection classes
- Event model
- Date and time facilities
- Internationalization
- Miscellaneous utility classes such as string tokenizer, random-number generator and bit
- Array

## 24. What is String Tokenizer ?

`StringTokenizer`: This `StringTokenizer` class is used to convert a `String` of text into its tokens.

## 25. What is Input Stream?

`InputStream`: The abstract class `InputStream` declares methods to read bytes from a particular source. `InputStream` is the superclass of most byte input streams in `java.io`.

## 26. What is Output Stream ?

`OutputStream`: The abstract class `OutputStream` is analogous to `InputStream`; it provides an abstraction for writing bytes to a destination

## 27. Define Character Streams ?

`Character Streams`: The abstract classes for reading and writing streams of characters are `Reader` and `Writer`.

Each supports methods similar to those of its byte stream counterpart—`InputStream` and `OutputStream`, respectively.

`Reader` has a `read` method that returns a `char` as the lowest 16 bits of an `int`. And, `Writer` has methods that write `char` arrays.

The character streams were designed after the byte streams to provide full support for working with Unicode characters, and in the process the contracts of the classes were improved to make them easier to work with.

### **28.What is Inner Class ?**

There are four other types of classes, loosely known as inner classes, that can be defined in a Java program. Used correctly, inner classes are an elegant and powerful feature of the Java language.

### **29.What is Static Member ?**

A static member class is a class (or interface) defined as a static member of another class.

A static method is called a class method, so, by analogy, we could call this type of inner class a "class class," but this terminology would obviously be confusing.

A static member class behaves much like an ordinary top-level class, except that it can access the static members of the class that contains it. Interfaces can be defined as static members of classes.

### **30.Define Anonymous classes?**

An anonymous class is a kind of local class that has no name; it combines the syntax for class definition with the syntax for object instantiation. While a local class definition is a Java statement, an anonymous class definition (and instantiation) is a Java expression, so it can appear as part of a larger expression, such as method invocation.

### **31.How does Java achieve portability?**

Java programs are portable across operating systems and hardware environments. Portability is advantageous because,

You need only one version of your software to serve a broad market. The Internet, in effect, becomes one giant, dynamic library.

You are no longer limited by your particular computer platform.

### **32.What's the difference between an interface and an abstract class?**

An abstract class may contain code in method bodies, which is not allowed in an interface. With abstract classes, you have to inherit your class from it and Java does not allow multiple inheritances. On the other hand, you can implement multiple interfaces in your class.

### **33.How do you know if an explicit object casting is needed?**

If you assign a superclass object to a variable of a subclass's data type, you need to do explicit casting. For example: Object a; Customer b; b = (Customer) a; When you assign a subclass to a variable having a superclass type, the casting is performed automatically.

### **34.What's the difference between constructors and other methods?**

Constructors must have the same name as the class and can not return a value. They are only called once while regular methods could be called many times.

### **35.What is meant by Object Oriented Programming?**

OOP is a method of programming in which programs are organised as cooperative collections of objects. Each object is an instance of a class and each class belong to a hierarchy.

### **36.What is an Instance?**

An instance has state, behaviour and identity. The structure and behaviour of similar classes are defined in their common class. An instance is also called as an object.

### **37.What are the core OOP's concepts?**

Abstraction, Encapsulation, Inheritance and Polymorphism are the core OOP's concepts.

### **38.What is meant by abstraction?**

Abstraction defines the essential characteristics of an object that distinguish it from all other kinds of objects. Abstraction provides crisply-defined conceptual boundaries relative to the perspective of the viewer. Its the process of focussing on the essential characteristics of an object. Abstraction is one of the fundamental elements of the object model.

### **39.What is meant by Encapsulation?**

Encapsulation is the process of compartmentalising the elements of an abstraction that defines the structure and behaviour. Encapsulation helps to separate the contractual interface of an abstraction and implementation.

#### **40. What are Encapsulation, Inheritance and Polymorphism?**

Encapsulation is the mechanism that binds together code and data it manipulates and keeps both safe from outside interference and misuse. Inheritance is the process by which one object acquires the properties of another object. Polymorphism is the feature that allows one interface to be used for general class actions.

#### **41. What are methods and how are they defined?**

Methods are functions that operate on instances of classes in which they are defined. Objects can communicate with each other using methods and can call methods in other classes. Method definition has four parts. They are name of the method, type of object or primitive type the method returns, a list of parameters and the body of the method. A method's signature is a combination of the first three parts mentioned above.

#### **42. What are different types of access modifiers (Access specifiers)?**

Access specifiers are keywords that determine the type of access to the member of a class. These keywords are for allowing privileges to parts of a program such as functions and variables. These are:

public: Any thing declared as public can be accessed from anywhere.

private: Any thing declared as private can't be seen outside of its class.

protected: Any thing declared as protected can be accessed by classes in the same package and subclasses in the other packages.

default modifier : Can be accessed only to classes in the same package.

#### **43. What is an Object and how do you allocate memory to it?**

Object is an instance of a class and it is a software unit that combines a structured set of data with a set of operations for inspecting and manipulating that data. When an object is created using new operator, memory is allocated to it.

#### **44. What is method overloading and method overriding?**

When a method in a class having the same method name with different arguments is said to be method overloading. Method overriding : When a method in a class having the same method name with same arguments is said to be method overriding.

#### **45. What gives java it's "write once and run anywhere" nature?**

All Java programs are compiled into class files that contain bytecodes. These byte codes can be run in any platform and hence java is said to be platform independent.

#### **46. What is a constructor? What is a destructor?**

Constructor is an operation that creates an object and/or initialises its state. Destructor is an operation that frees the state of an object and/or destroys the object itself. In Java, there is no concept of destructors. Its taken care by the JVM.

#### **47. What is the difference between constructor and method?**

Constructor will be automatically invoked when an object is created whereas method has to be called explicitly

#### **48. What is Static member classes?**

A static member class is a static member of a class. Like any other static method, a static member class has access to all static methods of the parent, or top-level, class.

#### **49. What is Garbage Collection and how to call it explicitly?**

When an object is no longer referred to by any variable, java automatically reclaims memory used by that object. This is known as garbage collection. System. gc() method may be used to call it explicitly

#### **50. In Java, How to make an object completely encapsulated?**

All the instance variables should be declared as private and public getter and setter methods should be provided for accessing the instance variables.

#### **51. What is static variable and static method?**

static variable is a class variable which value remains constant for the entire class static method is the one which can be called with the class itself and can hold only the static variables

**52. What is finalize() method?**

finalize () method is used just before an object is destroyed and can be called just prior to garbage collection.

**53. What is the difference between String and StringBuffer?**

- a) String objects are constants and immutable whereas StringBuffer objects are not.
- b) String class supports constant strings whereas StringBuffer class supports growable and modifiable strings.

**54. What is the difference between Array and vector?**

Array is a set of related data type and static whereas vector is a growable array of objects and dynamic

**55. What is the difference between this() and super()?**

this() can be used to invoke a constructor of the same class whereas super() can be used to invoke a super class constructor.

**56. Explain working of Java Virtual Machine (JVM)?**

JVM is an abstract computing machine like any other real computing machine which first converts .java file into .class file by using Compiler (.class is nothing but byte code file.) and Interpreter reads byte codes

**57. What is garbage collection? What is the process that is responsible for doing that in java?**

Reclaiming the unused memory by the invalid objects. Garbage collector is responsible for this process

**58. What is a daemon thread?**

These are the threads which can run without user intervention. The JVM can exit when there are daemon thread by killing them abruptly.

**59. What is the finalize method do?**

Before the invalid objects get garbage collected, the JVM give the user a chance to clean up some resources before it got garbage collected.

**60. What is mutable object and immutable object?**

If a object value is changeable then we can call it as Mutable object. (Ex., StringBuffer,...) If you are not allowed to change the value of an object, it is immutable object. (Ex., String,Integer, Float, ...)

**61. What is the basic difference between string and stringbuffer object?**

String is an immutable object. StringBuffer is a mutable object.

**62. What is the purpose of Void class?**

The Void class is an uninstantiable placeholder class to hold a reference to the Class object representing the primitive Java type void.

**63. What is an Abstract Class?**

Abstract class is a class that has no instances. An abstract class is written with the expectation that its concrete subclasses will add to its structure and behaviour, typically by implementing its abstract operations.

**64. What are inner class and anonymous class?**

Inner class: classes defined in other classes, including those defined in methods are called inner classes. An inner class can have any accessibility including private.

Anonymous class: Anonymous class is a class defined inside a method without a name and is instantiated and declared in the same place and cannot have explicit constructors

**65. What is interface and its use?**

Interface is similar to a class which may contain method's signature only but not bodies and it is a formal set of method and constant declarations that must be defined by the class that implements it. Interfaces are useful for:

- a) Declaring methods that one or more classes are expected to implement
- b) Capturing similarities between unrelated classes without forcing a class relationship.
- c) Determining an object's programming interface without revealing the actual body of the class.

### **66. What are the methods provided by the object class?**

The Object class provides five methods that are critical when writing multithreaded Java programs:

notify

notifyAll

wait (three versions)

### **PART-B**

1. What is inheritance? Explain different types of inheritance supported by java with an example
2. Explain Abstract classes and dynamic binding with an example program
3. What is meant by stream? What are types of stream and classes? Explain with program
4. Write about the properties of Java interface with an example code
5. Explain the InputStream, OutputStream class hierarchy with an example program.
6. Explain the Reader, Writer stream class hierarchy with an example program
7. Discuss on the visibility of base class members in privately and publicly inherited classes
8. What are abstract classes? Give an example (with the program) to illustrate the use of abstract classes
9. Does Java support multiple inheritances? Justify your answer with an example
10. Explain multi threading in Java

**UNIT – III**  
**PART – A**

**1. Java Database Connectivity:**

Java Database Connectivity or in short JDBC is a technology that enables the java program to manipulate data stored into the database.

JDBC is Java application programming interface that allows the Java programmers to access database management system from Java code. It was developed by JavaSoft, a subsidiary of Sun Microsystems. JDBC is consists of four Components: The JDBC API, JDBC Driver Manager, The JDBC Test Suite and JDBC-ODBC Bridge.

JDBC is an API specification developed by Sun Microsystems that defines a uniform interface for accessing various relational databases. JDBC is a core part of the Java platform and is included in the standard JDK distribution.

**2. JDBC Driver Manager:**

The JDBC DriverManager class defines objects which can connect Java applications to a JDBC driver. DriverManager has traditionally been the backbone of the JDBC architecture.

Its main purpose is to provide a means of managing the different types of JDBC database driver.

When opening a connection to a database it is the DriverManager's role to choose the most appropriate driver from the previously loaded drivers.

A Java program that uses the JDBC API loads the specified driver for a particular DBMS before it actually connects to a database. The JDBC DriverManager class then sends all JDBC API calls to the loaded driver.

**3. Remote Method Invocation:**

Remote Method Invocation (RMI) facilitates object function calls between Java Virtual Machines (JVMs).

JVMs can be located on separate computers - yet one JVM can invoke methods belonging to an object stored in another JVM. Methods can even pass objects that a foreign virtual machine has never encountered before, allowing dynamic loading of new classes as required.

The first thing we need to do is to agree upon an interface, an interface is a description of the methods we will allow remote clients to invoke. An interface is a method which contains abstract methods.

**4. What is skeleton and stub? What is the purpose of those?**

Stub is a client side representation of the server, which takes care of communicating with the remote server. Skeleton is the server side representation. But that is no more in use... it is deprecated long before in JDK.

**5. What are the components of HTTP URL?**

The components are host, an optional port, path, filename, section and query string.

**6. Define URL encoding.**

URL encoding involves replacing all unsafe and nonprintable characters with a percent sign (%) followed by two hexadecimal digits corresponding to the character's ASCII value.

**7. Explain about URL Encoding.**

HTTP specification requires that the URL data should be encoded in such a way that it can be used on almost any hardware and software platforms. Information specified in this way is called URL encoded. If parameters are passed as a part of query string or path information, they will take the form of 'Name-Value' pairs. variable=value1&variable2=value2& so on for each variable defined in the form. The variables or name value pairs are separated by '&'. Real ampersand is escaped –that is, encoded as a two-digit hexadecimal value representing the character. Escaped characters are indicated in URL-encoded string by the percent (%) sign. Blank spaces are replaced by '+' sign. Before the script can deal with the data it has to parse and decode it. The script scans through the string looking for an ampersand. When it is found the string is broken from that point. The variable's name is every thing up to the equal sign in the string and the value is every thing after the equal sign. The script continues to parse the original string for the next ampersand, and so on until the original string is exhausted. After the variables are separated, they are decoded as follows.

1. Replace all plus signs with blank spaces.

2. Replace all %## (Percent sign followed by two hexadecimal digits) with the corresponding ASCII character. Separate the name-value pairs from the URL and store the values separately.

### **8.What are the responsibilities of stub?**

A stub for a remote object is the client side proxy for the remote object. A client side stub is responsible for:

- Initiating a call to the remote object
- Marshaling arguments to a marshal stream
- Informing the remote reference layer that the call should be invoked
- Unmarshaling the return value or exception from a marshal stream

### **9.What is the role of skeleton in RMI?**

A skeleton for a remote object is a server side entity that contains a method which dispatches calls to the actual remote object implementation. The skeleton is responsible for

- Unmarshaling arguments from the marshal stream.
- Making the up-call to the actual remote object.
- Marshalling the return value of the call to an exception onto the Marshall stream

### **10.List down the layers of RMI architecture.**

- Stubs/Skeletons
- Remote reference layer
- Transport layer

### **11.What is meant by loop back address?**

A zone that enables the server to direct traffic to itself. The host number is almost always 127.0.0.1.

### **12.What are the issues of next generation IP?**

The issues to be considered in IP next generation are

- Addresses Space Growth
- Support large Global networks
- A clear way of transition from the existing IP to new IP next generation

### **13.What is the difference between TCP and UDP?**

TCP:

- Connection oriented transport protocol
- Sends data as a stream of bytes
- Guarantee of delivery

UDP:

- Connection less protocol
- Datagram service
- No guarantee of delivery.

### **14.What does ICMP provide?**

ICMP provides

- Error messaging
- Demand reply functions

### **15.Define IGMP.**

It is Internet Group Management protocol. It provides

- Broadcasting
- Multicasting

### **16.Give the important terminologies in HTTP Connection**

A transport layer virtual circuit established between two application programs for the purpose of communication.

#### **Message**

The basic unit of HTTP communication, consisting of a structured sequence of octets matching the syntax defined in and transmitted via the connection.

#### **Request**

An HTTP request message.

#### **Response**

An HTTP response message.

### **17.Define the GET() and POST() method**

**GET()**



The GET method means retrieve whatever information (in the form of an entity) is identified by the Request-URI. If the Request-URI refers to a data-producing process, it is the produced data which shall be returned as the entity in the response and not the source text of the process, unless that text happens to be the output of the process.

### **POST()**

The POST method is used to request that the destination server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the RequestLine

### **18.What is the role of server?**

The server

- Manages application tasks

- Handles storage

- Handles security

- Provides scalability

- Handles accounting and distribution

### **19.What is API -Application Program Interface?**

A set of routines, protocols, and tools for building software applications. A good API makes it easier to develop a program by providing all the building blocks. A programmer puts the blocks together. Most operating environments, such as MS- Windows, provide an API so that programmers can write applications consistent with the operating environment. Although APIs are designed for programmers, they are ultimately good for users because they guarantee that all programs using a common API will have similar interfaces. This makes it easier for users to learn new programs.

### **20.Define socket**

The socket is a software abstraction used to represent the terminals of a connection between two machines or processes.

### **21.What are the basic operations of client sockets?**

Connect to a remote machine

- Send data
- Receive data
- Close a connection

### **22.What are the basic operations of Server socket? Bind to a port**

- Listen for incoming data

- Accept connections from remote machines on the bound port

### **23.List all the socket classes in java. Socket**

- ServerSocket

- Datagram Socket

- Multicast Socket

- Secure sockets

### **24.What the Socket Object does?**

Socket object is the java representation of a TCP connection when a socket is created; a connection is opened to the specified destination.

### **25.What is meant by Server Socket?**

ServerSocket represents a listening TCP connection. Once an incoming connection is requested, the ServerSocket object will return a Socket object representing the connection.

### **26.What do you mean by DatagramSocket and DatagramPacket?**

DatagramSocket represents a connectionless datagram socket. This class works with the DatagramPacket class to provide for communication using the UDP protocol.

### **27.Write a note on Connect Exception.**

This exception is raised when a connection is refused at the remote host. (ie, no process is listening on that port).

**28. What is a multicast socket?**

Multicasting sends data from one host to many different hosts, which are in the multicast group.

**29. What is multicast address and the range of address?**

A multicast address is the address of a group of hosts called a multicast group. Multicast addresses are IP addresses in the range 224.0.0.0 to 239.255.255.255

**30. What are the different types of IP addresses?**

Unicast address: It is used for transmitting a message to single destination node

Multicast address: It delivers a message to a group of destination nodes, which are necessarily in the same sub network.

Broadcast address: It transmits a message to all nodes in a sub network.

**31. What is meant by protocol handler?**

Protocol handlers are used to retrieve the web objects using application specific protocols. The protocols are specified in the URL referencing the object.

**32. How are the protocol handlers implemented?**

Four different classes in the java.net package implement the protocol handlers:

URL

URLConnection

URLStreamHandler

URLStreamHandlerFactory

**33. What are the methods for parsing URLs?**

parseURL(URL u, String spec, int start, int limit)- splits the URL into parts

setURL(URL u, String protocol, String host, int port, String file, String ref) –assigns values to the URL's fields.

**34. What do you mean by JDBC?**

JDBC Part of the Java Development Kit which defines an application-programming interface for Java for standard SQL access to databases from Java programs.

**35. Define ODBC.**

It is a standard for accessing different database systems. There are interfaces for Visual Basic, Visual C++, SQL and the ODBC driver pack contains drivers for the Access, Paradox, dBase, Text, Excel and Btrieve databases

**36. What is the difference between node and host?**

A node is any addressable device connected to a network whereas the host is a more specific descriptor that refers to a networked general-purpose computer rather than a single purpose device (such as a printer).

**37. What is the purpose of routers?**

Router operates like electronic postal workers that evaluate and forward packets between networks.

**38. Define protocol.**

A protocol is a formal set of rules that must be followed in order to communicate.

**39. Why are the protocols layered?**

Layering protocols simplifies the task of communicating over the network and it allows for reuse of layers that are not specific to a particular application.

**40. Define port.**

A port is a logical channel to an application running on a host. ie., The applications running on the host machines are uniquely identified by port numbers.

**41. What do you mean by well-known ports?**

Port numbers can range from 1 to 65535; however ports 1 to 1023 are reserved. These reserved ports are referred to as well-known ports because the Internet Assigned Numbers Authority publicly documents the applications that use them.

#### **42.What is meant by Name Resolution?**

Name Resolution is the process of mapping a hostname to its corresponding IP Address. One way to translate a hostname to an IP address is to look it up in a simple text file. The second way is the domain name service, which is a distributed database containing all registered hostnames on the Internet and their IP addresses.

#### **43.Define protocol tunneling.**

Protocol tunneling is the process of encapsulating one protocol within another protocol that operates on the same layer.

#### **44.Define URI, URL, URN.**

1. URI (Uniform Resource Identifier): It identifies an object on the Internet.
2. URL (Uniform Resource Locator): It is a specification for identifying an object such as a file, newsgroup, CGI program or e-mail address by indicating the exact location on the internet.
3. URN (Uniform Resource Name): It is a method for referencing an object without declaring the full path to the object.

#### **PART-B**

1. Explain the JDBC database access in detail
2. Implement a simple client-server program using RMI in java that displays a message
3. Explain in detail the creation, instantiation and usage of java beans objects
4. Explain the steps involed to create JDBC connectivity.list the advantages of JDBC.
5. Write a java program using JDBC connection
6. Explain in deatail about TCP sockets with a program
7. Explain in detail about UDP sockets with a program
8. Explain in detail about servlet database connectivity with an example of student database
9. Explain URL Rewriting and classes

Explain InetAddress class in detail

**UNIT – IV**  
**PART – A**

**1.What is Servlets?**

Servlets are Java technology's answer to CGI programming. They are programs that run on Web server and build Web pages.

Java servlets are more efficient, easier to use, more powerful, more portable, and cheaper than traditional CGI and than many alternative CGI-like technologies.

The Servlet API, contained in the Java package hierarchy javax.servlet, defines the expected interactions of a Web container and a servlet.

**2.What are Servlets?**

A small program that runs on a server, the term usually refers to a Java applet that runs within a Web server environment. This is analogous to a Java applet that runs within a Web browser environment. Java servlets are becoming increasingly popular as an alternative to CGI programs. The biggest difference between the two is that a Java applet is persistent. This means that once it is started, it stays in memory and can fulfill multiple requests. In contrast, a CGI program disappears once it has fulfilled a request. The persistence of Java applets makes them faster because there's no wasted time in setting up and tearing down the process.

**3.What is AWT Package:**

AWT stands for Abstract Window ToolKit.

It is a portable GUI library between Solaris and Windows 95/NT and Mac System 7.X(soon) for stand-alone applications and/or applets.

It provides many classes for programmers to use. It is your connection between your application and the native GUI.

The AWT hides you from the underlying details of the GUI your application will be running on and thus is at very high level of abstraction.

**4.What is Container ?**

Containers:

Containers (Frames, Dialogs, Windows and Panels) can contain components and are themselves components, thus can be added to Containers.

Containers usually handle events that occurred to the Components, although nothing prevents you from handling events in the component.

The method of handling events in the Container (i.e. Frame) is preferred over the latter, since we want to centralize event handling.

**5.Define Component?**

They are generally the stuff that the user interacts with

Components are Buttons, TextAreas, Scrollbars, etc. in other words the visible UI controls that the user interacts with, all of which have been added to a Container.

**6.What is Layout Manager:**

How Components are "laid out" within a Container is described by the LayoutManager class.

Since the LayoutManager class is abstract, we can not use it directly. We must sub-class it and provide our own functionality or use a derived class of LayoutManager (i.e. BorderLayout, CardLayout, GridLayout, etc) already created for us.

**7.What are the Types of Layout?**

Types of Layout available are:

Flow Layout: It is the default layout manager for most components. It lays out horizontally

BorderLayout: Arranges up to five components in five positions: Center, East, West, North, and South.

GridLayout: Place components in a Row vs Column matrix. Components fill slots starting on top row, left to right, then move to next row down.

GridBagLayout: The most powerful layout manager is the GridBagLayout shown below. It is very useful when you have an elaborate interface with lots of components.

GridBagLayout places a component according to the settings in an instance of the helper class GridBagConstraints.

## Event Package:

The java.awt.event package defines classes and interfaces used for event handling in the AWT and Swing. The members of this package fall into three categories:

### Events

The classes with names ending in "Event" represent specific types of events, generated by the AWT or by one of the AWT or Swing components.

### Listeners

The interfaces in this package are all event listeners; their names end with "Listener".

These interfaces define the methods that must be implemented by any object that wants to be notified when a particular event occurs.

Note that there is a Listener interface for each Event class.

### Adapters

Each of the classes with a name ending in "Adapter" provides a no-op implementation for an event listener interface that defines more than one method.

When you are interested in only a single method of an event listener interface, it is easier to subclass an Adapter class than to implement all of the methods of the corresponding Listener interface.

## 8.What is Action Event:

An object of this class represents a high-level action event generated by an AWT component.

Instead of representing a direct user event, such as a mouse or keyboard event, ActionEvent represents some sort of action performed by the user on an AWT component.

The *getID()* method returns the type of action that has occurred.

## 9.What is ActionListener:

This interface defines the method that an object must implement to listen for action events on AWT components.

When an ActionEvent occurs, an AWT component notifies its registered ActionListener objects by invoking their actionPerformed() methods.

The general type is public abstract interface ActionListener extends java.util.EventListener  
{// Public Instance Methods public abstract void actionPerformed (ActionEvent e);}

## 10.What is Painting?

To understand how AWT's painting API works, helps to know what triggers a paint operation in a windowing environment.

In AWT, there are two kinds of painting operations: system-triggered painting, and application triggered painting.

System-triggered Painting

App-triggered Painting

## 11.Servlet containers

A Servlet container is a specialized web server that supports Servlet execution.

It combines the basic functionality of a web server with certain Java/Servlet specific optimizations and extensions – such as an integrated Java runtime environment, and the ability to automatically translate specific URLs into Servlet requests.

## 12.What are Applets?

A program designed to be executed from within another application. Unlike an application, applets cannot be executed directly from the operating system. A well-designed applet can be invoked from many different applications. Web browsers, who are often equipped with Java virtual machines, can interpret applets from Web servers. Because applets are small in files size, cross-platform compatible, and highly secure (can't be used to access users' hard drives), they are ideal for small Internet applications accessible from a browser.

## 13.What do you mean by Server-side?

Occurring on the server side of a client-server system. For example, on the World Wide Web, CGI scripts are server-side applications because they run on the Web server. In contrast, JavaScript scripts are client-side

because they are executed by your browser (the client). Java applets can be either server-side or client-side depending on which computer (the server or the client) executes them.

#### **14. Write a note on Environment variables.**

In CGI, the server prepares the environment variables before it launches the CGI script. These represent the current state of the server that is asking for the information. The environment variables are not set from the command line but are created on the fly, and lasts only until that particular script is finished. Each script gets its own unique set of variables and multiple scripts can be executed at once, each in its own environment.

#### **15. Define JSP.**

Java Server Pages (JSP) are simple technology used to generate dynamic HTML on the server side.

#### **16. Define Directives.**

Directives are JSP elements that provide global information about an entire JSP page

#### **17. Write down the various attributes for the page directives in JSP.**

The page directive defines information that will be globally available for that Java Server Page,

- language
- extends
- import
- session
- buffer
- contenttype

#### **18. Define: Dynamic proxy.**

A dynamic proxy is a class that implements a list of interfaces, which you specify at runtime when you create the proxy. To create a proxy, use the static method `java.lang.reflect.Proxy.newProxyInstance()`. This method takes three arguments:

- The class loader to define the proxy class

- An invocation handler to intercept and handle method calls

- A list of interfaces that the proxy instance implements

#### **19. What is a layout manager and what are different types of layout managers available in java AWT?**

A layout manager is an object that is used to organize components in a container. The different layouts are available are `FlowLayout`, `BorderLayout`, `CardLayout`, `GridLayout` and `GridBagLayout`.

#### **20. How are the elements of different layouts organized?**

`FlowLayout`: The elements of a `FlowLayout` are organized in a top to bottom, left to right fashion.

`BorderLayout`: The elements of a `BorderLayout` are organized at the borders (North, South, East and West) and the center of a container.

`CardLayout`: The elements of a `CardLayout` are stacked, on top of the other, like a deck of cards. `GridLayout`: The elements of a `GridLayout` are of equal size and are laid out using the square of a grid.

`GridBagLayout`: The elements of a `GridBagLayout` are organized according to a grid. However, the elements are of different size and may occupy more than one row or column of the grid. In addition, the rows and columns may have different sizes. The default Layout Manager of `Panel` and `Panel` sub classes is `FlowLayout`.

#### **21. What is source and listener?**

**source** : A source is an object that generates an event. This occurs when the internal state of that object changes in some way.

**listener** : A listener is an object that is notified when an event occurs. It has two major requirements. First, it must have been registered with one or more sources to receive notifications about specific types of events. Second, it must implement methods to receive and process these notifications.

## **PART-B**

1. Explain Java Applets with an example
2. Explain life cycle of an Applet.
3. Write a program to display images using Applet
4. Write an Applet Program for passing values using parameters.
5. Explain in details about Event handling with an example
6. Explain the following in detail
  - a. Graphics and update method
  - b. primitives drawing function with sample input arugements
7. Explain in detail about the AWT and Layout Managers
8. Explain Servlet programming with an example
9. Explain HTTP Request and Respones in details
10. Explain cookies with an example

UNIT – V  
PART – A

### **1. Define XML.**

XML is a meta-markup language that provides a format for describing structured data. This facilitates more structured declarations of content and more meaningful search results across multiple platforms.

### **2. What is XML**

XML stands for EXtensible Markup Language

XML is a markup language much like HTML

XML was designed to carry data, not to display data

XML tags are not predefined. We must define our own

tags XML is designed to be self-descriptive

XML is a W3C Recommendation.

### **3. What is Web directory**

A **web directory** or **link directory** is a directory on the World Wide Web.

It specializes in linking to other web sites and categorizing those links.

A web directory is not a search engine, and does not display lists of web pages based on keywords, instead it lists web sites by category and subcategory.

The categorization is usually based on the whole web site, rather than one page or a set of keywords, and sites are often limited to inclusion in only one or two categories.

Web directories often allow site owners to directly submit their site for inclusion, and have editors review submissions for fitness.

### **4. What is Search Engine**

A **Web search engine** is a search engine designed to search for information on the World Wide Web. Information may consist of web pages, images and other types of files. Some search engines also mine data available in newsgroups, databases, or open directories. Unlike Web directories, which are maintained by human editors, search engines operate algorithmically or are a mixture of algorithmic and human input.

### **5. Define DTD.**

A DTD is a set of rules that specifies how to use XML markup. It contains specifications for each element, including what the element's attributes are, what values the attributes can take on and what elements can be contained in others.

### **6. What are the XML rules for distinguishing between the content of a document and the XML markup element?**

1. The start of XML markup elements is identified by either the less than symbol (<) or the ampersand

(&) character

2. Three other characters, the greater than symbol (>), the apostrophe or single quote (') and the double quotation marks (") are used by XML for markup.

3. To use these special characters as content within your document, you must use the corresponding general XML entity.

### **7. What is global.asa file?**

The global.asa file is a Active Server Application file you can track and manage the application and session events, variables and objects. When you start the application the server will load the global.asa file into memory.

### **8. Define response object and list its methods.**

The response object transmits information from the web server to browser. Methods are:

1. Write

2. BinaryWrite

3. Redirect

4. AppendToLog

5. AddHeader

6. Clear

7. Flush



## 9.How XML differ From HTML?

HTML	XML
HTML only displays and focuses on how data looks	Describes and focuses on the data
HTML is all about displaying information	XML all about information
HTML is not extensible	XML is extensible
HTML tags are predefined user must not invent his tags	XML tags are not predefined user must invent his tags
HTML tags are not case sensitive	XML tags are case sensitive

## 10.How to represent the XML Document?

In any markup language, the first element to appear is called the "root element", which defines what kind of document the file will be. In an HTML file, the <html> tag is the root element. An HTML file will always have the HTML element as the root element, while in an XML file, it can be anything. Eg:

```
<phonebook>
<number>
</number>
<name>
</name>
</phonebook>
```

## 11.What is PCDATA in XML?

Parsed Character Data (PCDATA) is a term used about text data that will be parsed by the XML parser. XML parsers normally parse all the text in an XML document. When an XML element is parsed, the text between the XML tags is also parsed:

```
<message>This text is also parsed</message>
```

## 12.Mention any 3 XML Parsers

SAX (Simple API for XML) Parser  
DOM (Document Object Model) Parser and  
XSLT (XML Style Sheet) Parsers.

## 13.What is the purpose of the XML DTD

The purpose of a DTD is to define the structure of an XML document. It defines the structure with a list of legal elements:

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

## 14.What is XML Prolog

XML file always starts with a Prolog. The minimal prolog contains a declaration that identifies the document as an XML document, like this:

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

The declaration may also contain additional information, like this:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
```

## 15.What is XSL Programming?

XSL (XML Stylesheet) Programming is the Next Generation of the CSS (Cascading Style Sheet Programming). In CSS, users can use certain style tags which the browsers can understand and are predefined by W3 consortium. XSL takes this to one step ahead and users can define any tags in the XML file. XML sheets can help in showing the same data in different formats. <?xml-stylesheet href="doc.xsl" type="text/xsl"?>

## 16.What is XSLT?

XSLT stands for XSL Transformations  
XSLT is the most important part of XSL

XSLT transforms an XML document into another XML document  
XSLT uses XPath to navigate in XML documents  
XSLT is a W3C Recommendation

### 17. What are Web Services?

Web services are application components  
Web services communicate using open protocols  
Web services are self-contained and self-describing  
Web services can be discovered using UDDI  
Web services can be used by other applications  
XML is the basis for Web services

### 18. What are web services platform elements?

SOAP (Simple Object Access Protocol)  
UDDI (Universal Description, Discovery and Integration)  
WSDL (Web Services Description Language)

### 19. What is SOAP?

SOAP is an XML-based protocol to let applications exchange information over HTTP. SOAP is language independent  
SOAP is based on XML  
SOAP is simple and extensible

### 20. What is WSDL?

WSDL is an XML-based language for locating and describing Web services.  
WSDL stands for Web Services Description Language  
WSDL is based on XML  
WSDL is used to describe Web services

### 21. Create a simple Web Service that converts the temperature from Fahrenheit to Celsius, and vice versa:

```
<% @ WebService Language="VBScript" Class="TempConvert" %>
Imports System
Imports System.Web.Services
Public Class TempConvert :Inherits WebService <WebMethod()>
Public Function FahrenheitToCelsius (ByVal Fahrenheit As String) As String
dim fahr fahr=trim(replace(Fahrenheit,",", ".")) if fahr="" or IsNumeric(fahr)=false then return "Error"
return (((fahr) - 32) / 9) * 5 end function <WebMethod()> Public Function CelsiusToFahrenheit (ByVal
Celsius As String) As String dim cel cel=trim(replace(Celsius,",", ".")) if cel="" or IsNumeric(cel)=false
then return "Error" return (((cel) * 9) / 5) + 32 end function end class
```

### 22. Define the XML Schema

The purpose of an XML Schema is to define the legal building blocks of an XML document, just like a DTD.  
Here are some reasons:

XML Schemas are extensible to future additions  
XML Schemas are richer and more powerful than  
DTDs XML Schemas are written in XML

### 23. What is a Simple Element in XML?

A simple element is an XML element that can contain only text. It cannot contain any other elements or attributes

The syntax for defining a simple element is:

```
<xs:element name="xxx" type="yyy"/>
```

### 24. Write the Syntax Rules SOAP

A SOAP message MUST be encoded using XML

- A SOAP message MUST use the SOAP Envelope namespace A
- SOAP message MUST use the SOAP Encoding namespace A
- SOAP message must NOT contain a DTD reference
- A SOAP message must NOT contain XML Processing Instructions

## **PART – B**

### **25.How to transform XML documents to other forms**

JAXP -The SAX API -The DOM API -XML Namespaces -Transforming a DOM Tree to an XML Document - Transforming an XML Document to an HTML Document -XSL Programming-XSL COMPONENT VIEW-XSLT Input Document View

### **26.Explain the main role of XPath in selecting the XML Data**

WHAT IS XPATH-XPath Path Expressions-XPath Standard Functions-XPath is Used inXSLT-XPATH is a W3C Recommendation-XPath Nodes-Selecting Nodes-XPath Axes-XPathAxes-Location Path Expression-Loading the XML Document

### **27.How the template based Transformation achieved with the help of XSLT**

XSL Languages-What is XSLT-XSLT Uses XPath-XSLT – Transformation-Correct StyleSheet Declaration-Create an XSL Style Sheet-Link the XSL Style Sheet to the XML DocumentXSLT Functions

### **28.How to write the web service and java web service client**

JAX RPC technology manages communication between a web service and client. –These are the basic steps for creating the web service and client-A service endpoint interface must conform to a few rules-Building the Service -The compile-service Task -The generate-WSDL Task -Packaging and Deploying the Service -Specifying the Endpoint Address -Deploying the Service

### **29.How to describe the Web services**

Defining Services

Port Types and Operations

Binding It All Together

### **30.How to represent data types in XML Schema**

Built-in datatypes

XML schema

User defined simple types

User defined complex types

### **31.Explain SOAP technology for providing the communication between the data object.**

What is SOAP-SOAP Syntax-Syntax Rules-Skeleton SOAP Message-SOAP Envelope Element-The encodingStyle Attribute-SOAP Header Element-SOAP Body Element-SOAP Fault Element-SOAP Fault Codes-SOAP HTTP Binding-Soap HTTP Binding-A SOAP request-The SOAP response

## **PART-B**

1. Give an XML program for storing book details
2. Explain in detail about Form Navigation with an example
3. Explain in detail to display XML data on browser
4. Explain in detail about various XSL tags with an example
- 5.Explain briefly XSLT with a program
6. Explain in detail about the web service with a clear illustration
- 7.Write a client program to access a web service
- 8.Explain the following in detail
  - a. UDDI
  - b. WSDL
- 9.Explain Java Web Services with an example
- 10.Discuss about web resources

