SEVENTH/FIFTH SEMESTER EXAMINATION, 2008-2009

INTRODUCTION TO WEB TECHNOLOGY

Time – 3 hours **Total Marks – 100**

Note: Attempt all questions.

Q. 1. Attempt any four parts: (5×4= 20)

Q.(a). Describe the role of W3C. Also give the names of primary hosts for W3C.

Ans. W3C Stands for the World Wide Web Consortium.

W3C is working to make the Web accessible to all users (despite differences in culture, education, ability, resources, and physical limitations).

Major W3C Achievements

Below you will find some of W3C's most important achievements. Obviously, this type of list is subjective and does not represent every aspect of W3C work. For a complete view of W3C work, please consult the list of W3C Activities and the index of W3C's technical reports.

First W3C Recommendation published is Portable Network Graphics (PNG) 1.0. In the mid-'90s, more industrial and academic users were discovering the Web and its graphics capabilities. W3C developed Portable Network Graphics (PNG) to provide a cross-platform alternative to the graphics formats most prevalent at that time, some of which had raised some patent licensing concerns.

Separating content from structure, CSS Level 1 is published. Cascading Style Sheets (CSS) is a simple mechanism for adding style (e.g. fonts, colors, spacing) to Web documents. CSS Level 2 (1998) included further and more sophisticated features.

Web Accessibility Initiative launched. W3C's Web Accessibility Initiative (WAI) guidelines for Web content, user agents, and authoring tools would become very popular among the Web community. WAI, in coordination with organizations around the world, pursues accessibility of the Web through four primary areas of work: technology, tools, education and outreach, and research and development.

W3C launches Web Services Activity. Subsuming the XML Protocol Activity and extending its scope, Web services provide a standard means of interoperating between different software applications, running on a variety of platforms and/or frameworks.

W3C adopts royalty-free Patent Policy. The W3C Patent Policy governs the handling of patents in the process of producing Web standards, and explicitly encourages the development of open standards.

RDF and OWL make a strong foundation for Semantic Web applications. RDF and OWL are Semantic Web standards that provide a framework for asset management, enterprise integration and the sharing and reuse of data on the Web. Respectively, they deliver structured descriptions and Web-based ontologies.

W3C gives voice to the Web with VoiceXML 2.0. Voice interaction can escape the physical limitations of keypads and displays as mobile devices become even smaller. The goal of

www.btechsamplepapers.in

VoiceXML 2.0 is to bring the advantages of Web-based development and content delivery to interactive voice response applications.

W3C describes principles of Web architecture. W3C's Technical Architecture Group (TAG) publishes "Architecture of the World Wide Web," a description of the principles that make the Web we know work, and work well. This condensed assessment of fifteen years of observations about the Web authored by many of those who designed the core Web standards is a valuable foundation on which to design future Web standards.

W3C Launches Group Linking Medical Industry with Semantic Web. W3C launched an Interest Group to connect medical industry verticals with Semantic Web experts in an effort to improve collaboration, research and development, and innovation adoption in the health care and life science industries. The first of its kind for W3C, Semantic Web for Health Care and Life Sciences Interest Group (HCLSIG) deploys standardized Semantic Web specifications into specific services defined by a user community.

Q.1(b). Explain the various protocols governing web.

Ans. Below is a list of protocols used for the world wide web.

ARP	Address Resolution Protocol
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Service
DSN	Data Source Name
FTP	File Transfer Protocol
HTTP	Hypertext Transfer Protocol
IMAP	Internet Message Access Protocol
ICMP	Internet Control Message Protocol
IDRP	ICMP Router-Discovery Protocol
IP	Internet Protocol
IRC	Internet Relay Chat Protocol
POP3	Post Office Protocol version 3
PAR	Positive Acknowledgment and Retransmission
RLOGIN	Remote Login
SMTP	Simple Mail Transfer Protocol
SSL	Secure Sockets Layer
SSH	Secure Shell
TCP	Transmission Control Protocol
TELNET	TCP/IP Terminal Emulation Protocol
UPD	User Datagram Protocol
UPS	Uninterruptible Power Supply

Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems. Its use for retrieving inter-linked resources led to the establishment of the World Wide Web.

HTTP development was coordinated by the World Wide Web Consortium and the Internet Engineering Task Force (IETF), culminating in the publication of a series of Requests for Comments (RFCs), most notably RFC 2616 (June 1999), which defines HTTP/1.1, the version of HTTP in common use

The **Transmission Control Protocol (TCP)** is one of the core protocols of the Internet Protocol Suite. TCP is one of the two original components, with Internet Protocol (IP), of the suite, so that the entire suite is commonly referred to as TCP/IP. Whereas IP handles lower-level transmissions from computer to computer as a message makes its way across the Internet, TCP operates at a higher level, concerned only with the two end systems, for example, a Web browser and a Web server. In particular, TCP provides reliable, ordered delivery of a stream of bytes from a program on one computer to another program on another computer. Besides the Web, other common applications of TCP

www.btechsamplepapers.in

include e-mail and file transfer. Among its other management tasks, TCP controls message size, the rate at which messages are exchanged, and network traffic congestion.

The **Internet Protocol** (IP) is a protocol used for communicating data across a packet-switched internetwork using the Internet Protocol Suite, also referred to as TCP/IP.

IP is the primary protocol in the Internet Layer of the Internet Protocol Suite and has the task of delivering distinguished protocol datagrams (packets) from the source host to the destination host solely based on their addresses. For this purpose the Internet Protocol defines addressing methods and structures for datagram encapsulation. The first major version of addressing structure, now referred to as Internet Protocol Version 4 (IPv4) is still the dominant protocol of the Internet, although the successor, Internet Protocol Version 6 (IPv6) is being deployed actively worldwide.

The **User Datagram Protocol** (UDP) is one of the core members of the Internet Protocol Suite, the set of network protocols used for the Internet. With UDP, computer applications can send messages, in this case referred to as datagrams, to other hosts on an Internet Protocol (IP) network without requiring prior communications to set up special transmission channels or data paths. UDP is sometimes called the Universal Datagram Protocol. The protocol was designed by David P. Reed in 1980 and formally defined in RFC 768.

Q.1(c). Discuss various factors which are to be addressed while designing websites for individual.

Ans. Factors involved in designing a website, with suggestions.

Feature	Suggestions and Comments
CODING	
No frames	frames are deprecated by W3C (ie not supported in future) - avoid them
Browser friendly 	use cross-platform HTML check rendering in text-only browser avoid 'deprecated' tags look to the future, but be sympathetic to legacy browsers
Search Engine friendly	META TAGS for description, keywords proprietary tags for intranet robot use keywords in title, headings, filename, link-text
Clean coding	remove empty, repeated, non standard (proprietary) tags group tag rather than individual tags 'pretty printing' for ease of maintenance validate HTML and CSS - author often overlooks mistakes
Links checked	regularly perform an internal link check (eg use Xenu) link-rot check external urls
Organised	content directories, naming conventions plan for future & expansion directories for ubiquitously used graphics and files
Tools	converters can introduce errors - check HTML editors may 'correct' code - not always what you want!
Tables	browsers take time to compose and render over-complex tables careful nesting of tables (max 3) always use end-tags for table elements
Maintainability	coding must be clear and easy to maintain conflict of min filesize V ease of reading and layout of code CSS is an attempt to separate content from layout, to ease maintenance

www.btechsamplepapers.in

GRAPHICS	
Patterned backgrounds	rarely work with text foreground - text clarity is lost use to increase flexibility of highly graphical layout use CSS coding for graphical border
Colour backgrounds	use table cell colours rather than images
Re-use images	stored in cache hence no download time after first access
Optimised graphics min filesize	Reduce graphical filesize
appropriate format	.gif for logos / diagrams / transparent background .jpg for photos
palette	Use web-safe colours for bgcolor, font, gif palette
Tools	PaintShop Pro, DCEnhancer, Xara Webstyle
MULTI-MEDIA	
Audio	keep to minimum, reduce quality to 11KHz, use MPEG3 or RealAudio use WAV for embedded sound

Dos and don'ts for achieving usability in design

- Do:

- Use ALT tags for all graphics, especially navigation graphics.
- Use black text on white background whenever possible for optimal legibility.
- Use either plain-color backgrounds or extremely subtle background patterns.
- Make sure text is in a printable color (not white).
- Place navigation in a consistent location on each page of your website.
- Use a familiar location for navigation bars.
- Keep the design from scrolling horizontally.
- Use one axis of symmetry for centered text on a page.
- Encourage scrolling by splitting an image at the fold.

- Don't:

- Allow ALT tags to get clipped (especially an issue for small, fixed width images).
- Display static text in blue or underlined.
- Use **boldface or ALL CAPS for long pieces of text. These slow down reading.**
- Leave too much white space—reduces scannability.
- Make the user scroll to find critical information, especially transaction buttons and navigation links.
- Use horizontal rules to separate chunks of content.
- Alternate too frequently between centered text and left-aligned text. Most text should be left-aligned.
- Fix pages at larger than 800 x 600 pixels. Larger pages may force users to scroll horizontally.

Q.1(d). Write a short note on cyber laws.

Ans. Cyberlaw or, less colloquially, Internet law, is a term that encapsulates the legal issues related to use of communicative, transactional, and distributive aspects of networked information devices and technologies. It is less a distinct field of law in the way that property or contract are, as it is a domain covering many areas of law and regulation. Some leading topics include intellectual property, privacy, freedom of expression, and jurisdiction.

Cyber terrorists usually use the computer as a tool, target, or both for their unlawful act either to gain information which can result in heavy loss/damage to the owner of that intangible sensitive information. Internet is one of the means by which the offenders can gain such price sensitive information of companies, firms, individuals, banks, intellectual property crimes (such as stealing new product plans, its description, market programme plans, list of customers etc.), selling illegal articles, pornography etc. this is done through many methods such as phishing, spoofing, pharming, internet phising, wire transfer etc. and use it to their own advantage without the consent of the individual.

(A) Computer Viruses: Viruses are used by Hackers to infect the user's computer and damage data saved on the computer by use of

www.btechsamplepapers.in

"payload" in viruses which carries damaging code. Person would be liable under I.T Act only when the consent of the owner is not taken before inserting virus in his system. The contradiction here is that though certain viruses causes temporary interruption by showing messages on the screen of the user but still it's not punishable under Information Technology Act 2000 as it doesn't cause tangible damage. But, it must be made punishable as it would fall under the ambit of 'unauthorised access' though doesn't cause any damage. Harmless viruses would also fall under the expression used in the provision "to unsurp the normal operation of the computer, system or network". This ambiguity needs reconsideration.

(B) Phishing: By using e-mail messages which completely resembles the original mail messages of customers, hackers can ask for verification of certain information, like account numbers or passwords etc. here customer might not have knowledge that the e-mail messages are deceiving and would fail to identify the originality of the messages, this results in huge financial loss when the hackers use that information for fraudulent acts like withdrawing money from customers account without him having knowledge of it.

(C) Spoofing: This is carried on by use of deceiving Websites or e-mails. These sources mimic the original websites so well by use of logos, names, graphics and even the code of real bank's site.

(D) Phone Phishing: Is done by use of in-voice messages by the hackers where the customers are asked to reveal their account identification, and passwords to file a complaint for any problems regarding their accounts with banks etc.

(E) Internet Pharming: Hacker here aims at redirecting the website used by the customer to another bogus website by hijacking the victim's DNS server (they are computers responsible for resolving internet names into real addresses - "signposts of internet), and changing his I.P address to fake website by manipulating DNS server. This redirects user's original website to a false misleading website to gain unauthorised information.

(F) Risk Posed On Banks And Other Institutions: Wire transfer is the way of transferring money from one account another or transferring cash at cash office. This is most convenient way of transfer of cash by customers and money laundering by cyber terrorists. There are many guidelines issued by Reserve Bank of India (RBI) in this regard, one of which is KYC (Know Your Customer) norms of 2002. Main objective of which is to:

(1) Ensure appropriate customer identification, and

(2) Monitor the transaction of suspicious nature and report it to appropriate authority every day bases.

(G) Publishing Pornographic Material In Electronic Form: Section 67 of the Information Technology Act, 2000 in parallel to Section 292 of Indian Penal Code, 1860 makes publication and transmission of any material in electronic that's lascivious or appeals to the prurient interest a crime, and punishable with imprisonment which may extend to 5 years and fine of 1 lakh rupees and subsequent offence with an imprisonment extending to 10 years and fine of 2 lakhs.

(H) Investment Newsletter: We usually get newsletter providing us free information recommending that investment in which field would be profitable. These may sometimes be a fraud and may cause us huge loss if relied upon. False information can be spread by this method about any company and can cause huge inconvenience or loss through junk mails online.

Sides of INDIAN Cyber Law or IT Act of INDIA

1. Data Protection and Privacy law
2. Electronic and Digital Signature law
3. Computer crime law
4. Telecommunication law
5. Intellectual Property law

www.rajeshsamorepapers.in

Q.1(e). What are the qualities of a good team members for a web project?

Ans. A Web Designer is one who is engaged in setting up websites and web pages, to be accessed through the Internet and World Wide Web by millions of end-users.

Web Designers are basically the people who plan, layout, and create the millions of websites being stored on web servers around the world. Thus, Web Designers need to understand how the Internet and World Wide Web operates, and how to use and apply the various computer and web languages and codes required to make it all happen.

Web Designers should be detail-oriented, well-organized, and methodical, since much of their work involves creating, adjusting, re-creating, re-adjusting of website and web page codes and commands.

To be a good web designer, you need to be a good designer. Here are the top 5 things you need to do.

1. Know your customer's requirements
2. Communicate with the customer often to give status and to find out if requirements have changed. Work in an iterative fashion so that you can show new features on a regular basis (like daily).
3. Be organized. If you are not naturally organized, force yourself to file things so that they can be found later. Start taking notes in meetings (especially on the phone) and write up meeting minutes for all meetings. Refer back to them when you create tasks and risk lists.
4. Create and maintain task lists and risk lists.
5. Stay up on technical concepts.

Qualities of Good Web Design Companies

It may take some work to research web designers, but finding a quality web design company is well worth your time and effort. You do not want to waste your time and money creating a site that is not effective, nor to you want to work with a web design company that cannot meet deadlines or does not produce quality work.

When looking for a web design company, there are a few things that will be helpful to consider before making your final choice:

How long they've been in the business– Because web design has evolved into such a popular trade, new companies are popping up everywhere. Seeking out a reputable company that has been in business for a substantial amount of time is probably your best bet because web designers who do not produce quality work usually do not last.

Former clients– A graphic design company that has worked with high-profile clients can most likely be counted on to produce quality work. Do not, however, simply overlook a company because it is new. You may find that a start-up web design company can provide you with exactly what you want, and may even cost less than established companies because they are new and trying to bring in business.

Client feedback– Some site design companies will post feedback from previous clients on their website. This can be helpful in allowing you to assess what other clients felt about their services, and whether or not you think they will be right for you. If a web design company does not offer client reviews, you can also try an online search to see if any other information about the company surfaces. Lastly, look at the web design company's own site and see if they have experience with ecommerce web design. If a web design company has a poorly created site, it might be an indication of how your site will turn out.

Timeliness and reliability– Launching your website may be time-sensitive, so you will want to hire a web design company who can meet deadlines and complete work in a timely manner. The best way to find out about the punctuality of a company is to contact their previous clients, who can be found in their portfolio. If a company's portfolio is unavailable to you, you can always enter their name into a search engine to see if you can find any reviews.

Q.1(f). Define the following with suitable example

(i) Web applications (ii) Target users.

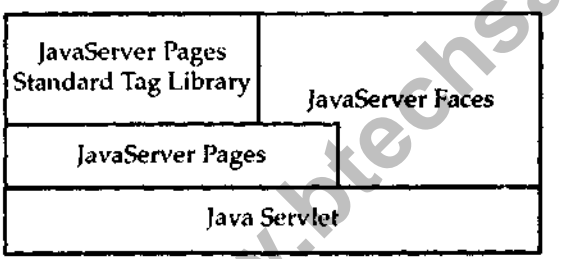
Ans. GETTING STARTED WITH WEB APPLICATIONS

www.btechsamplepapers.in

A web application is a dynamic extension of a web or application server. There are two types of web applications:

- **Presentation-oriented:** A presentation-oriented web application generates interactive web pages containing various types of markup language (HTML, XML, and so on) and dynamic content in response to requests. Chapters 11 through 22 cover how to develop presentation-oriented web applications.
- **Service-oriented:** A service-oriented web application implements the endpoint of a web service. Presentation-oriented applications are often clients of service-oriented web applications.

Web components are supported by the services of a runtime platform called a web container. A web container provides services such as request dispatching, security, concurrency, and life-cycle management. It also gives web components access to APIs such as naming, transactions, and email.



Java Web Application Technologies

Certain aspects of web application behavior can be configured when the application is installed, or deployed, to the web container. The configuration information is maintained in a text file in XML format called a web application deployment descriptor (DD). A DD must conform to the schema described in the Java Servlet Specification.

Most web applications use the HTTP protocol and support for HTTP is a major aspect of web components. For a brief summary of HTTP protocol features see Appendix C.

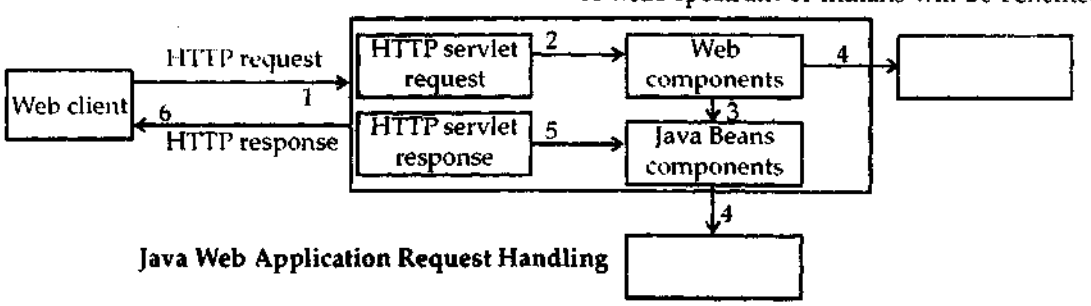
(I) WEB APPLICATION LIFE CYCLE

A web application consists of web components, static resource files such as images, and helper classes and libraries. The web container provides many supporting services that enhance the capabilities of web components and make them easier to develop. However, because a web application must take these services into account, the process for creating and running a web application is different from that of traditional stand-alone Java classes. The process for creating, deploying, and executing a web application can be summarized as follows:

1. Develop the web component code.
2. Develop the web application deployment descriptor.
3. Compile the web application components and helper classes referenced by the components.
4. Optionally package the application into a deployable unit.
5. Deploy the application into a web container.
   Access a URL that references the web application.

(II) TARGET USERS

A wide spectrum of Indians will be benefited:



Java Web Application Request Handling

www.btechsamplepapers.in

General house hold applications; complimentary to the purification technologies.

People in potentially hazardous areas.

Water quality monitoring agencies.

Pollution monitoring industries.

Medical practitioners and patients in economically backward areas.

Agriculture.

Food processing.

Beverages.

Fisheries.

Q. 2. Attempt any four parts: (5×4= 20)

Q.(a). What do you mean by DOM? Design DOM for a university. (Make assumption yourself).

Ans. The Document Object Model (DOM) is a cross-platform and language-independent convention for representing and interacting with objects in HTML, XHTML and XML documents. Objects under the DOM (also sometimes called "Elements") may be specified and addressed according to the syntax and rules of the programming language used to manipulate them. The rules for programming and interacting with the DOM are specified in the DOM Application Programming Interface (API).

(Document Object Model)

Historical Background

DOM is a standard defined by the W3C, just like XML

DOM was not designed specifically for Java technology (unlike SAX)

DOM is cross-platform and crosslanguage

Uses OMG's IDL to define interfaces

IDL to language binding

DOM Characteristics

Access XML document as a tree structure

Composed of mostly element nodes and text nodes

Can "walk" the tree back and forth

Larger memory requirements

Fairly heavyweight to load and store

Use it when for walking and modifying the tree

DOM in Action

Parser

Creates

Tree

Input

XMl

Document

DOM Tree and Nodes

XML document is represented as a tree

A tree is made of nodes

There are 12 different node types

Nodes may contain other nodes

(depending on node types)

uparent node contains child nodes

Q.2(b). Why does FORM an important tag in web applications? Explain attributes ACTION and METHOD.

Ans. The HTML Form Tag

The HTML form tag marks the boundaries of a data input form in a web page. You can have more than one HTML forms in a single page. However, do not nest HTML forms (Don't put one form in another form)!

The general syntax of the form tag is given below:

<form action="server-script-url-here" method="GET or POST" >...Input elements go here...

</form>

Following are the attributes you can have in the form tag:

Action

The HTML form action attribute specifies the URL to submit the data entered in the HTML form. This will normally be a server side script written in a scripting language like Perl or PHP.

Example 1 **Absolute** URL:

<form action="http://someserver/cgi-bin/handle-data.pl">

Absolute URL is required when the form handling is performed at some other website.

Normally, if the form handling script is in the same website, you can give the relative URL:

www.btechsamplepapers.in

Example 2 Relative URL: <form action="/cgi-bin/handle-data.pl">

Learn about dynamically changing action field in the article: Switching form action field dynamically

Method

Method of sending data to the action URL. The value is either GET or POST. In the GET method, the form data is sent as part of the URI. like:handle-data.pl?name - john&email-john@server.com

GET method is suitable for small forms like a search form. For larger forms, use the POST method.

Default is the GET method.

Q.2(c). Define CSS. Write a CSS rule that makes all the text 2.5 times larger than the base font of the system.

Ans. Cascading Style Sheets

Cascading Style Sheets (CSS) is a style sheet language used to describe the presentation (that is, the look and formatting) of a document written in a markup language. Its most common application is to style web pages written in HTML and XHTML, but the language can be applied to any kind of XML document, including SVG and XUL.

CSS is designed primarily to enable the separation of document content (written in HTML or a similar markup language) from document presentation, including elements such as the colors, fonts, and layout. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for tableless web design). CSS can also allow the same markup page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. While the author of a document typically links that document to a CSS stylesheet, readers can use a different stylesheet, perhaps one on their own computer, to override the one the author has specified.

CSS specifies a priority scheme to determine which style rules apply if more than one rule matches against a particular element. In this so-called cascade, priorities or weights are calculated and assigned to rules, so that the results are predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998).

Use of CSS

Prior to CSS, nearly all of the presentational attributes of HTML documents were contained within the HTML markup; all font colors, background styles, element alignments, borders and sizes had to be explicitly described, often repeatedly, within the HTML. CSS allows authors to move much of that information to a separate stylesheet resulting in considerably simpler HTML markup.

Headings (h1 elements), sub-headings (h2), sub-sub-headings (h3), etc., are defined structurally using HTML. In print and on the screen, choice of font, size, color and emphasis for these elements is presentational.

Basic Syntax

Rules

Selectors

Any HTML element is a possible CSS1 selector. The selector is simply the element that is linked to a particular style. For example, the selector in

P { text-indent: 3em }

is P.

Class Selectors

A simple selector can have different classes, thus allowing the same element to have different styles. For example, an author may wish to display code in a different color depending on

www.btechsamplepapers.in

its language:

code.html { color: #191970 }

code.css { color: #4b0082 }

The above example has created two classes, css and html for use with HTML's CODE element. The CLASS attribute is used in HTML to indicate the class of an element, e.g.,

<P CLASS=warning>Only one class is allowed per selector.

For example, code.html.proprietary is invalid.</p>

Classes may also be declared without an associated element:

.note { font-size: small }

In this case, the note class may be used with any element.

A good practice is to name classes according to their function rather than their appearance. The note class in the above example could have been named small, but this name would become meaningless if the author decided to change the style of the class so that it no longer had a small font size.

ID Selectors

ID selectors are individually assigned for the purpose of defining on a per-element basis. This selector type should only be used sparingly due to its inherent limitations. An ID selector is assigned by using the indicator "#" to precede a name. For example, an ID selector could be assigned as such:

#svp940 { text-indent: 3em }

This would be referenced in HTML by the ID attribute:

<P ID=svp940>Text indented 3em</P>

Contextual Selectors

Contextual selectors are merely strings of two or more simple selectors separated by white space. These selectors can be assigned normal properties and, due to the rules of cascading order, they will take precedence over simple selectors. For example, the contextual selector in

P EM { background: yellow }

is P EM. This rule says that emphasized text within a paragraph should have a yellow background; emphasized text in a heading would be unaffected.

Q.2(d). Differentiate between HTML and XML.

Ans. What is XML?

- XML stands for Extensible Markup Language
- XML is a markup language much like HTML
- XML was designed to carry data,
- XML tags are not predefined.
- XML is designed to be self-descriptive.
- XML is a W3C Recommendation

What is HTML?

HTML - short for Hypertext Markup Language is the predominant markup language for the creation of web pages. It provides a means to describe the structure of text-based information in a document by denoting certain text as headings, paragraphs, lists, and to supplement that text with interactive forms, embedded images, and other objects.

- Publish online documents with headings, text, tables, lists, photos,
- Retrieve online information via hypertext links, at the click of a button.
- Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products.
- Include spread-sheets, video clips, sound clips, and other applications directly in their documents.

The Difference between XML and HTML

Web developers may notice some difference between HTML and XML, which is due to the fact that they are both derived from XML document is provided or displayed, enables document authors to focus on creating documents that contain only relevant semantic information for their particular problem domain. This is a gentle introduction to XML

www.btechsamplepapers.in

and using XML with Java. We see what makes XML tick and the advantages of XML. Also we have programs to show how to use Excess parser for parsing and printing an XML document, to generate XML from subjective data.

The most significant difference between HTML and XML is that HTML has predefined elements and attributes whose behavior is well specified, while XML does not. Instead, document authors can create their own XML words that are specific to their application or business needs. XML vocabularies currently exist for a large number of industries and applications from financial filings.

Another difference between HTML and XML is that XML introduces the concept of regular ness. The regular ness rules of XML remove some of ambiguity inherent in processing markup languages like HTML by enforcing rules such as mandating that all attribute values must be in quotes.

Q.2(e). Write a Java script that finds the smallest of given N integers.

Ans.

```
/// <summary>
/// method to find the largest and smallest number
/// in an integer array
/// </summary>
public static void FindLargestAndSmallest()
{
    int arraySize;
    bool isNum;
    int largestNum;
    int smallestNum;
    int[] numArray = new int[50];
    //ask the user for the size of their array
    Console.WriteLine("Enter the size of Array");
    //read in the value
    string sizeString = Console.ReadLine();
    //we will now use TryParse to get the numeric value entered
    isNum = Int32.TryParse(sizeString, out arraySize);
    //now we will determine if the value is numeric
    if (isNum)
    {
        //then entered a numeric value so now we need to
        //ask for the values they want in their array
        Console.WriteLine("Enter array value:");
        for (int i = 0; i < arraySize; i++)
        {
            //int variable to hold the our value from TryParse
            int temp;
            //read in each value and add it to our array if it's
            //a numeric value
            string arrayValue = Console.ReadLine();
            isNum = Int32.TryParse(arrayValue, out temp);
            //now do our check
            if (isNum)
            {
                //value is numeric soa dd to our array
                numArray[i] = temp;
            }
            else
            {
                //value isnt numeric
                Console.WriteLine("Array values must be numeric!");
                break;
            }
        }
        Console.Write("\r\n");
        //now we need to set the values of our largest
        //and smallest number variables
        largestNum = numArray[0];
        smallestNum = numArray[0];
        //now loop through the length of our array
        for (int i = 1; i < arraySize; i++)
        {
            //check if the current array index is larger
```

www.btechsamplepapers.in

```
        //than the largest number variable
        if (numArray[i] > largestNum)
      {
          //if it is then that is the largest number
          //(for this iteration)
          largestNum = numArray[i];
      }
       //otherwise check to see if this array index
        //is smaller than the smallest number
        else if (numArray[i] < smallestNum)
      {
           //that is our smallest number (for this
iteration)
           smallestNum = numArray[i];
      }
    }
     //now print out the results
      Console.WriteLine("Largest value is: {0}",
largestNum);
     Console.WriteLine("Smallest value is: {0}",
smallestNum);
  }
  else
  {
      Console.WriteLine("Array size must be
numeric!");
  }
}
```

Q.2(f). Explain the term SAX with suitable example.

Ans. DOM parser is one which makes the entire XML passed as a tree Structure and will have it in memory. Any modification can be done to the XML.

SAX parser is one which triggers predefined events when the parser encounters the tags in XML. Eventdriven parser. Entire XML will not be stored in memory. Bit faster than DOM. NO modifications can be done to the XML.

SAX Presentation

The SAX (SAdt eXplain) system generates hypertext descriptions of conceptual models designed with the SADT methodology.

Background

During information system development, software engineering methodologies are used for analysis and design. Conceptual modelling with some formalized or semi-formalized graphic language is frequently used in the analysis phase. The validation of these conceptual models is crucial: by ensuring the correctness and quality of the model at an early stage, time and money can be saved. The validation task might be difficult if the domain expert does not know the formal language used by the analyst. Thus the analyst should provide the domain expert with additional and comprehensible documentation: textual natural-language descriptions of the model are suitable for this purpose. The motivation behind SAX is that valuable analyst time could be saved if this documentation is produced automatically. The automatically generated documentation can also be used by the analyst to debug in the model. SAX generates hypertext descriptions of conceptual models designed with the Structured Analysis and Design Technique (SADT, developed by Softech Inc.).

System Properties

SAX is implemented in Prolog and runs in the Windows environment. The system generates model descriptions within a few seconds.

SAX is characterized by a hybrid approach to text generation. A classical NLG approach was chosen for the planning of the global structure of the text using a schema-like representation, while templates were used at the sentence level. The flexibility of traditional static templates was enhanced by developing a new template formalism. See SAX publications for more details.

Q. 3. Answer any two parts: (10×2=20)

(a) (i). What do you mean by Java Bean? Give the advantages of Java Beans.

(ii) Give various steps to develop a simple Bean with example.

www.btechsamplepapers.in

Ans. A Java Bean is a software component that has been designed to be reusable in a variety of different environments. There is no restriction on the capability of a Bean. It may perform a simple function, such as checking the spelling of a document, or a complex function, such as forecasting the performance of a stock portfolio. A Bean may be visible to an end user. One example of this is a button on a graphical user interface. A Bean may also be invisible to a user. Software to decode a stream of multimedia information in real time is an example of this type of building block. Finally, a Bean may be designed to work autonomously on a user's workstation or to work in cooperation with a set of other distributed components. Software to generate a pie chart from a set of data points is an example of a Bean that can execute locally. However, a Bean that provides real-time price information from a stock or commodities exchange would need to work in cooperation with other distributed software to obtain its data.

We will see shortly what specific changes a software developer must make to a class so that it is usable as a Java Bean. However, one of the goals of the Java designers was to make it easy to use this technology. Therefore, the code changes are minimal.

Advantages of Java Beans

A software component architecture provides standard mechanisms to deal with software building blocks. The following list enumerates some of the specific benefits that Java technology provides for a component developer:

- A Bean obtains all the benefits of Java's "write-once, run-anywhere" paradigm.
- The properties, events, and methods of a Bean that are exposed to an application builder tool can be controlled.
- A Bean may be designed to operate correctly in different locales, which makes it useful in global markets.
- Auxiliary software can be provided to help a person configure a Bean. This software is only needed when the design-time parameters for that component are being set. It does not need to be included in the run-time environment.
- The configuration settings of a Bean can be saved in persistent storage and restored at a later time.
- A Bean may register to receive events from other objects and can generate events that are sent to other objects.

Q.3(b) (i). What is servlet? Give its applications. Also explain servlet API.

(ii). Write short note on JSDK.

Ans. Life of a JSP file.

Servlets are Java programming language objects that dynamically process requests and construct responses. The Java Servlet API allows a software developer to add dynamic content to a Web server using the Java platform. The generated content is commonly HTML, but may be other data such as XML. Servlets are the Java counterpart to non-Java dynamic Web content technologies such as PHP, CGI and ASP.NET. Servlets can maintain state across many server transactions by using HTTP cookies, session variables or URL rewriting.

The Servlet API, contained in the Java package hierarchy javax.servlet, defines the expected interactions of a Web container and a servlet. A Web container is essentially the component of a Web server that interacts with the servlets. The Web container is responsible for managing the lifecycle of servlets, mapping a URL to a particular servlet and ensuring that the URL requester has the correct access rights.

A Servlet is an object that receives a request and generates a response based on that request. The basic servlet package defines Java objects to represent servlet requests and responses, as well as objects to reflect the servlet's

www.btechsamplepapers.in

configuration parameters and execution environment. The package javax.servlet.http defines HTTP-specific subclasses of the generic servlet elements, including session management objects that track multiple requests and responses between the Web server and a client. Servlets may be packaged in a WAR file as a Web application.

Servlets can be generated automatically by JavaServer Pages (JSP) compiler, or alternately by template engines such as WebMacro. Often servlets are used in conjunction with JSPs in a pattern called "Model 2", which is a flavor of the model-view-controller pattern.

Lifecycle of a Servlet

The Servlet lifecycle consists of the following steps:

1. The Servlet class is loaded by the container during start-up.

2. The container calls the init() method. This method initializes the servlet and must be called before the servlet can service any requests. In the entire life of a servlet, the init() method is called only once.

3. After initialization, the servlet can service client requests. Each request is serviced in its own separate thread. The container calls the service() method of the servlet for every request. The service() method determines the kind of request being made and dispatches it to an appropriate method to handle the request. The developer of the servlet must provide an implementation for these methods. If a request for a method that is not implemented by the servlet is made, the method of the parent class is called, typically resulting in an error being returned to the requester.

4. Finally, the container calls the destroy() method that takes the servlet out of service. The destroy() method like init() is called only once in the lifecycle of a Servlet.

Here is a simple servlet that just generates HTML

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class HelloWorld extends HttpServlet {
  public void doGet(HttpServletRequest
request, HttpServletResponse response)
      throws ServletException, IOException {
    PrintWriter out = response.getWriter();
   out.println("<!DOCTYPE HTML PUBLIC \"-/
/W3C//DTD HTML 4.0 " +
  "Transitional//EN\">\n" +
  "<html>\n" +
  "<head><title>Hello          WWW</title></
head>\n" +
          "<body>\n" +
           "<h1>Hello WWW</h1>\n" +
          "</body></html>");
 }
}
```

(II) WRITE SHORT NOTE ON JSDK?

Java Servelet Development kit

This tutorial has been written for use with JSDK2.0/2.1 and/or Apache JServ. Both have now been officially superseded by JSDK2.3 and Apache Tomcat respectively. However Tomcat has a number of serious problems which need to be addressed before i'm prepared to recommend it, these are as follows (Warning rant mode=true) :

- Performance, Tomcat is generally slower than JServ
- Complexity, Tomcat is considerably more difficult to setup and use.
- Tomcat can't be easily de-integrated. Tomcat can act as both a servlet engine, JSP engine and a http webserver along with all sorts of other features. From a development point of view this is quite handy, but if your trying to set up a live webserver, you generally want to strip out everything that isn't

www.btechsamplepapers.in

needed. I suspect this probably can be done, but I can't find anything in the docs which explains how it could be done.

- Tomcat uses XML Configuration files. Although this is more 'standards compliant', the use of XML in this case has made configuration files longer and more difficult to read. The style used for JServ/ Servletrunner is much cleaner and simpler and therefore less prone to error.
- Related to the above, Tomcat cannot handle JServ configuration files, so for me to switch my own systems would require a lot of work re-writing configuration files. Work on making Tomcat a drop in replacement for JServ has been promised, but I haven't seen anything yet.
- Tomcat seems to require a special build process using ANT. You can't simply drop class files into a directory and use them.
- Documentation is poor. I have difficulty understanding the docs and I'm a specialist in Java and Web technologies.

One of these days I may try to set up an open source project to produce a servlet only JSDK 2.2 (or better) api compatible engine with direct JServ compatibility. The aim would be to produce a servlet engine that can communicate with apache using the ajpv21 protocol and reads JServ configuration files. I'm currently investigating two ways of achieving this :

Q.3(c) (i). How does a web server link physically on the Internet? Explain with the help of block diagram.

(ii). Define firewall. What are the common functions of firewall?

Ans. The term web server can mean one of two things:

1. A computer program that is responsible for accepting HTTP requests from clients (user agents such as web browsers), and serving them HTTP responses along with optional data contents, which usually are web pages such as HTML documents and linked objects (images, etc.).
2. A computer that runs a computer program as described above.

Common features

A standard 19" Rack of servers as seen from the front.

Although web server programs differ in detail, they all share some basic common features.

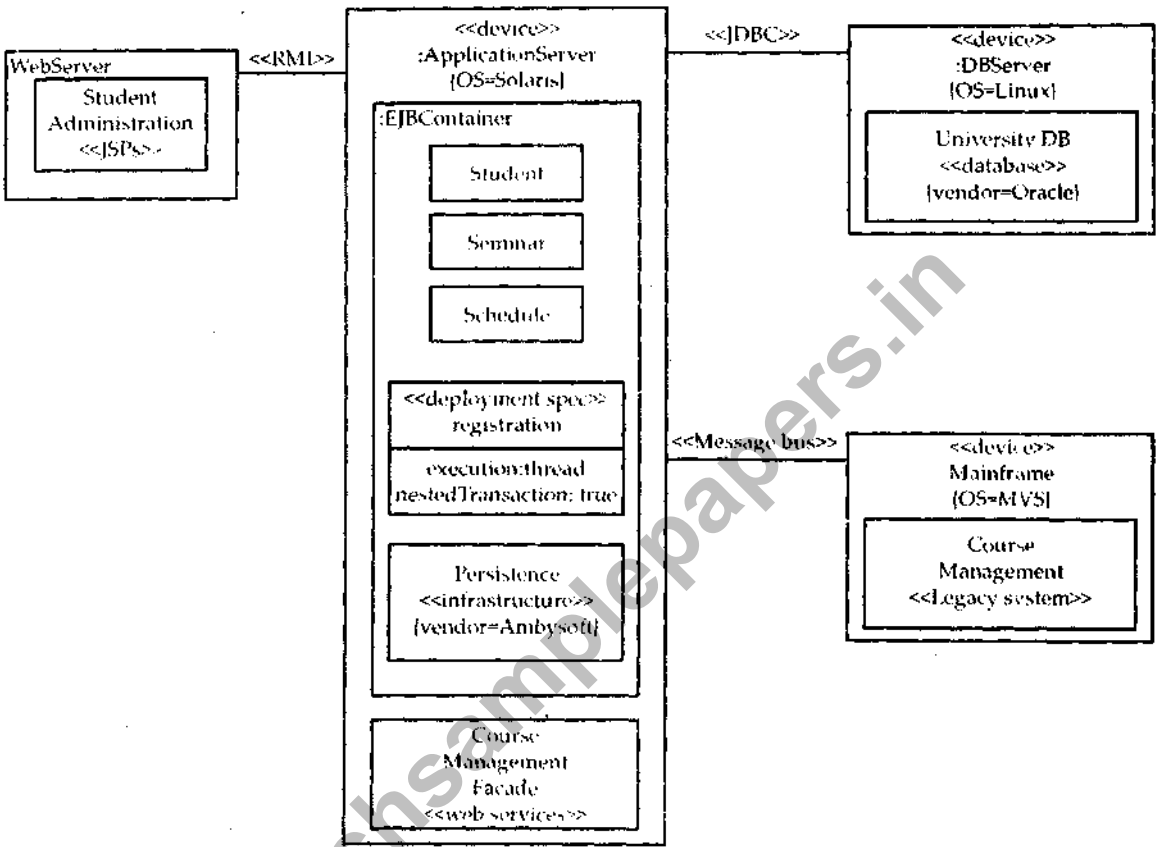
1. HTTP: every web server program operates by accepting HTTP requests from the client, and providing an HTTP response to the client. The HTTP response usually consists of an HTML or XHTML document, but can also be a raw file, an image, or some other type of document (defined by MIME-types). If some error is found in client request or while trying to serve it, a web server has to send an error response which may include some custom HTML or text messages to better explain the problem to end users.

2. Logging: usually web servers have also the capability of logging some detailed information, about client requests and server responses, to log files; this allows the webmaster to collect statistics by running log analyzers on these files.

In practice many web servers implement the following features also:

1. Authentication, optional authorization request (request of user name and password) before allowing access to some or all kind of resources.
2. Handling of static content (file content recorded in server's filesystem(s)) and dynamic content by supporting one or more related interfaces (SSI, CGI, SCGI, FastCGI, JSP,ColdFusion, PHP, ASP, ASP.NET, Server API such as NSAPI, ISAPI, etc.).
3. HTTPS support (by SSL or TLS) to allow secure (encrypted) connections to the server on the standard port 443 instead of usual port 80.
4. Content compression (i.e. by gzip encoding) to reduce the size of the responses (to lower bandwidth usage, etc.).

www.btechsamplepapers.in



5. Virtual hosting to serve many web sites using one IP address.

6. Large file support to be able to serve files whose size is greater than 2 GB on 32 bit OS.

7. Bandwidth throttling to limit the speed of responses in order to not saturate the network and to be able to serve more clients.

(II) DEFINE FIREWALL .WHAT ARE THE COMMON FUNCTION OF FIREWALL?

An example of a user interface for a firewall (Gufw)

A firewall is a part of a computer system or network that is designed to block unauthorized access while permitting authorized communications. It is a device or set of devices configured to permit, deny, encrypt, decrypt, or proxy all (in and out) co mputer traffic between different security domains based upon a set of rules and other criteria.

Firewalls can be implemented in either hardware or software, or a combination of both. Firewalls are frequently used to prevent unauthorized Internet users from accessing private networks connected to the Internet, especially intranets. All messages entering or leaving the intranet pass through the firewall, which examines each message and blocks those that do not meet the specified security criteria.

There are several types of firewall techniques:

1. Packet filter: Looks at each packet entering or leaving the network and accepts or rejects it based on user-defined rules. Packet filtering is fairly effective and transparent to users, but it is difficult to configure. In addition, it is susceptible to IP spoofing.
2. Application gateway: Applies security mechanisms to specific applications, such as FTP and Telnet servers. This is very effective,

www.btechsamplepapers.in

but can impose a performance degradation.

3. Circuit-level gateway: Applies security mechanisms when a TCP or UDP connection is established. Once the connection has been made, packets can flow between the hosts without further checking.

Programming element	Description
Firewall Create Rule	This function creates a new firewall rule.
Firewall Delete Rule	This function deletes a firewall rule.
Firewall Enable	This function enables or disables the firewall.
Firewall Enable Rule	This function enables or disables a firewall rule.
Firewall Get Rules	This function retrieves information about firewall rules.
Firewall Refresh	This function reloads the firewall run-time image settings from the registry, with the exception of rules and interfaces.
Firewall Set Interface	This function specifies that traffic through a network interface should be filtered by the firewall.
IsFirewall Enabled	This function determines whether the firewall is enabled.
IsInterface Firewalled	This function indicates whether traffic through an interface is filtered by the firewall.

Proxy server: Intercepts all messages entering and leaving the network. The proxy server effectively hides the true network addresses

The following table shows the firewall functions with a description of the purpose of each.

Q. 4. Answer any two parts: **(10×2= 20)**

(a). Develop an HTML and Java script document to evaluate the roots of the quadratic equation. Explain each step clearly.

Ans.

```
var a,b,c,x1,x2;
function getValues() {
  signB = (document.form.firstSign.value == "+")
? 1 : -1;
  signC = (document.form.secondSign.value ==
"+") ? 1 : -1;
  a = document.form.x2Coef.value;
  b = document.form.xCoef.value * signB;
  c = document.form.endValue.value * signC;
}
function solveForX() {
  x1 = ((-1*b) + Math.sqrt((b*b) - 4*a*c))/ (2*a);
  x2 = ((-1*b) - Math.sqrt((b*b) - 4*a*c))/ (2*a);
  if(document.form.round.checked){
  x1 = Math.round(x1*1000)/1000;
  x2 = Math.round(x2*1000)/1000;
  }
}
function factorEq() {
  if( (Boolean(x1) != false) && (Boolean(x2) !=
false)) {
    x1Print = ((x1 * -1) > 0) ? "+" + (x1*-1) : x1 * -1;
    x2Print = ((x2 * -1) > 0) ? "+" + (x2*-1) : x2 * -1;
    aPrint = (a > 1 || a < 0) ? a : "";
  document.getElementById('testing').innerHTML
= "<font color=\"red\">The Solutions Are:</
font><br>x1 = " + x1 + "; x2 = " + x2 + "<br><font
color=\"red\"> The Equation Is:</font><br> " +
aPrint + "(x " + x1Print + ") (x " + x2Print + ")";
  } else {
  document.getElementById('testing').innerHTML
= "<font color=\"red\"><b>YOUR INPUT
PRODUCED AN ERROR:</b></font><br> You
have entered a non-integer into one of the fields
above, or the solution(s) for your equation
is(are) an imaginary(s) number!";
  }
}
function SolveEq() {
  getValues();
  solveForX();
```

www.btechsamplepapers.in

```
 factorEq();
}
<script type="text/javascript" src="quad
Equation.js"></script>
<div align="center">
<form name="form">
  <input type="text" name="x2Coef" size="2"
value="1"><b> X<sup>2</sup> </b>
    <select name="firstSign">
      <option value="+">+</option>
      <option value="-">-</option>
    </select>
  <input typle="text" name="xCoef" size="2"
value="1"><b> X</b>
    <select name="secondSign">
      <option value="+">+</option>
      <option value="-">-</option>
    </select>
     <input type="text" name="endValue"
size="2"><br>
  Round to the nearest thousandths?<input
type="checkbox" name="round">Yes<br><br>
     <input type="button" value="Solve!"
onClick="SolveEq()">
</form>
<p id="testing" style="width: 400px; height:
100px; border-width: 2px; border-style: groove;
border-color: orange; font-size: 16px; color: blue;
vertical-align: center">The Solution Will
Appear Here!</p>
</div>
<p><div align="center">
<font face="arial, helvetica" size"-2">Free
JavaScripts provided<br>
by <a href="http://javascriptsource.com">The
JavaScript Source</a></font>
</div><p>
```

Q.4(b) (i). Describe JSP architecture in brief.

(ii). Write short note on Error handling and debugging.

Ans. JSPs are built on top of SUN Microsystems' servlet technology. JSPs are essential an HTML page with special JSP tags embedded. These JSP tags can contain Java code. The JSP file extension is .jsp rather than .htm or .html. The JSP engine parses the .jsp and creates a Java servlet source file. It then compiles the source file into a class file,this is done the first time and this why the JSP is probably slower the first time it is accessed. Any time after this the special compiled servlet is executed and is therefore returns faster.

Steps required for a JSP request:

1. The user goes to a web site made using JSP. The user goes to a JSP page (ending with .jsp). The web browser makes the request via the Internet.
2. The JSP request gets sent to the Web server.
3. The Web server recognises that the file required is special (.jsp),therefore passes the JSP file to the JSP Servlet Engine.
4. If the JSP file has been called the first time,the JSP file is parsed,otherwise go to step 7.
5. The next step is to generate a special Servlet from the JSP file. All the HTMl required is converted to println statements.
6. The Servlet source code is compiled into a class.
7. The Servlet is instantiated,calling the init and service methods.
8. HTML from the Servlet output is sent via the Internet.
9. HTML results are displayed on the user's web browser

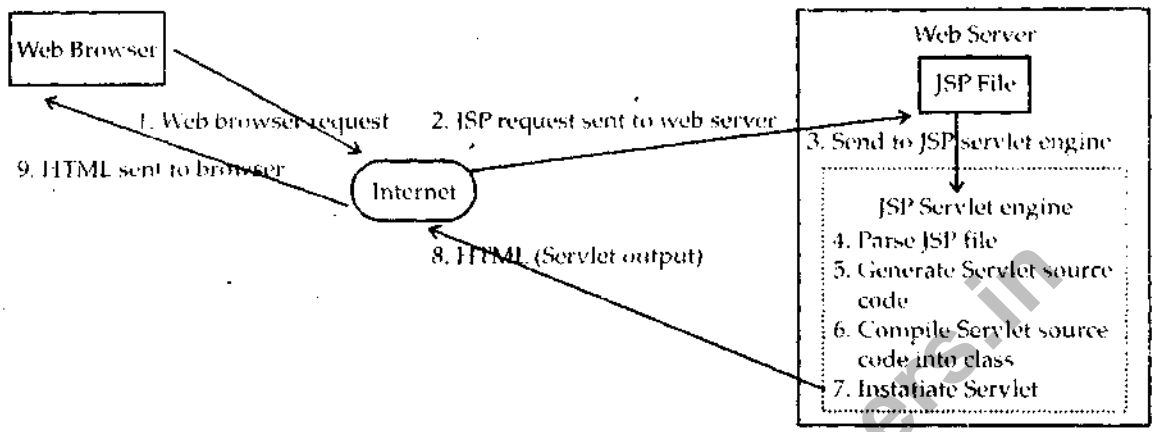
JAVA ERROR HANDLING

Exceptions

All Java errors are handled as exceptions. Exceptions extend the base class, Exception. You might have seen the try and catch block used. This is a way to "catch" exceptions - this essentially allowing you to handle various unexpected errors gracefully.

Life isn't perfect. Much as we'd like to think otherwise, sometimes we encounter exceptional behavior. Things like missing files, badly formatted data to be parsed, and operations on null things are just a few of the many exceptions you may meet in your programming career. When these things happen, the JVM throws a temper tantrum, or (more technically) throws

www.btechsamplepapers.in



an exception.

How do you know if an exception could be thrown? The hard way is when the compiler complains that you didn't properly handle the method. The easy way involves the API documentation. The standard API documentation on the Java site begins a method's documentation with its header: the access level, return type, method name, parameters, miscellaneous modifiers, and the declared exceptions. For example, the readLine method in class java.io.BufferedReader throws IOException. Its header looks like this:

public String readLine() throws IOException

If a method might throw an exception, it has to declare that exception in its header using the throws statement. (Note: just because a method declares an exception doesn't mean that it will throw it, just that it might.) If you use a risky method, you have to tell the compiler what to do in case something bad happens. This is called catching the exception.

Try and Catch

Try and catch blocks are really simple. The basic syntax follows. For this example, we'll try to read a file that doesn't exist. (The details of file I/O comes later in this tutorial: the important part is just that this code is risky: if the file "myfile.txt" doesn't exist, the FileReader constructor throws a FileNotFound Exception.)

```
System.out.println("Ready? Go!");
try
{
   System.out.println("I think I can...");
     FileReader reader = new FileReader
("myfile.txt");
   System.out.println("I knew I could!");
}
catch(FileNotFoundException ex)
{
   System.out.println("File not found.");
}
System.out.println("All done.");
```

Q.4(c) Explain briefly the following:

(i). Tomcat servers

(ii). Sharing data between JSP pages.

Ans. TOMCAT SERVERS

Tomcat server is a open source project of Java Servlet and JSP technologies developed under the Jakarta project at the Apache Software Foundation.

The most popluar tomcat server versions are: Tomcat 4.1.31, Tomcat 5.0.29 and Tomcat 6.0.11.

Tomcat4 server working with jdk 1.4, while Tomcat5 & Tomcat6 require jdk 1.5 or jdk 1.6.

What is a Tomcat Valve?

A Tomcat valve - a new technology introduced with Tomcat 4 which allows you to associate an instance of a Java class with a particular Catalina container.

This configuration allows the named class to act as a preprocessor of each request. These classes are called valves, and they must implement the org.apache.catalina.Valve

www.btechsamplepapers.in

interface or extend the org.apache.catalina.valves.ValveBase class. Valves are proprietary to Tomcat and cannot, at this time, be used in a different servlet/JSP container. At this writing, Tomcat comes configured with four valves:

- Access Log
- Remote Address Filter
- Remote Host Filter
- Request Dumper

Each of these valves (and their available attributes) are described as follows.

The Access Log Valve

The first of the Tomcat prepackaged valves is the Access Log valve: org.apache.catalina.valves.AccessLogValve. It creates log files to track client access information.

Some of the content that it tracks includes page hit counts, user session activity, user authentication information, and much more. The Access Log valve can be associated with an engine, host, or context container.

The following code snippet is an example entry using the org.apache.catalina.valves.AccessLogValve:

```
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="logs" prefix="localhost_access_log." suffix=".txt" pattern="common"/>
```

This code snippet states that the log files will be placed in the <CATALINA_HOME>/logs directory, prepended with the value localhost_access_log., and appended with the .txt suffix.

Installation of Apache & Tomcat Server

What is the Apache Web Server?

The Apache Web Server Project is a collaborative open source development effort with the explicit goal of creating a commercial-quality HTTP server. The original code was based upon the httpd 1.3 product developed by Rob McCool at the National Center for Supercomputing Applications (NCSA).

The project began in February 1995 and was made publicly available in April 1995, with a 0.6.2 release.

The Apache server is a jointly supervised product, managed by a group of volunteers known as the Apache Group. This group is located around the world, using the Internet to correspond, plan, design, and develop the application.

What is the Tomcat Manager Web application?

The Tomcat Manager Web application is packaged with the Tomcat server. It is installed in the context path of /manager and provides the basic functionality to manage Web applications running in the Tomcat server.

Some of the provided functionality includes the ability to install, start, stop, remove, and report on Web applications

Gaining Access to the Manager Web Application

Before you can use the Manager, you must set up a new user with the appropriate privileges to access the /manager application. If you look at the TOMCAT_HOME/webapps/manager/web.xml file in the /manager application, you'll notice a security constraint similar to the following code snippet:

```
<! - Define a Security Constraint on this Application -->
<security-constraint>
<web-resource-collection>
<web-resource-name>Entire Application</web-resource-name>
<url-pattern>/*</url-pattern>
</web-resource-collection>
<auth-constraint>
<!-- NOTE: This role is not present in the default users file -->
<role-name>manager</role-name>
</auth-constraint> </security-constraint>
```

The <security-constraint/> element defined in this code snippet secures the entire /manager Web application, providing access to only those users who have a defined role of manager.

It does this with essentially two sub-elements in the security constraint. The first sub-element, <web-resource-collection>, defines the resource that is protected by this constraint

The definition is made with the <url pattern>

www.techsamplepapers.in

sub-element. When you look at the previous snippet, you'll also notice that this sub-element is defined as follows:

<url-pattern>/*</url-pattern>

The value of the <url-pattern> sub-element uses a wildcard *, which protects all URLs within the /manager application with this security constraint. The second sub-element defines the role that has access to the protected resource.

It does this by using the <role-name> sub-element, which is listed in the following code snippet: <role-name>manager</role-name>

The value of this sub-element states that only users with a role of manager can access the resource protected by this

Q. 5. Describe any two of the following with suitable example : (10×2= 20)

Q.(a). Accessing a database from a JSP page.

Ans. This page will give you a short overview of Struts Framework and its main components. After finishing reading this, continue on to the tutorial to create a simple Struts-based Web application.

The Struts Framework is a standard for developing well-architected Web applications. It has the following features:

- Open source
- Based on the Model-View-Controller (MVC) design paradigm, distinctly separating all three levels:
  - Model: application state
  - View: presentation of data (JSP, HTML)
  - Controller: routing of the application flow
- Implements the JSP Model 2 Architecture
- Stores application routing information and request mapping in a single core file, struts-config.xml

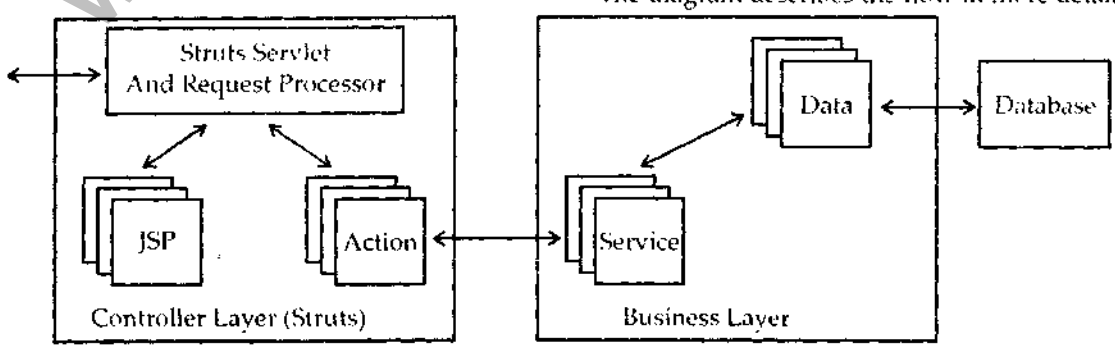
The Struts Framework, itself, only fills in the View and Controller layers. The Model layer is left to the developer.

Architecture Overview

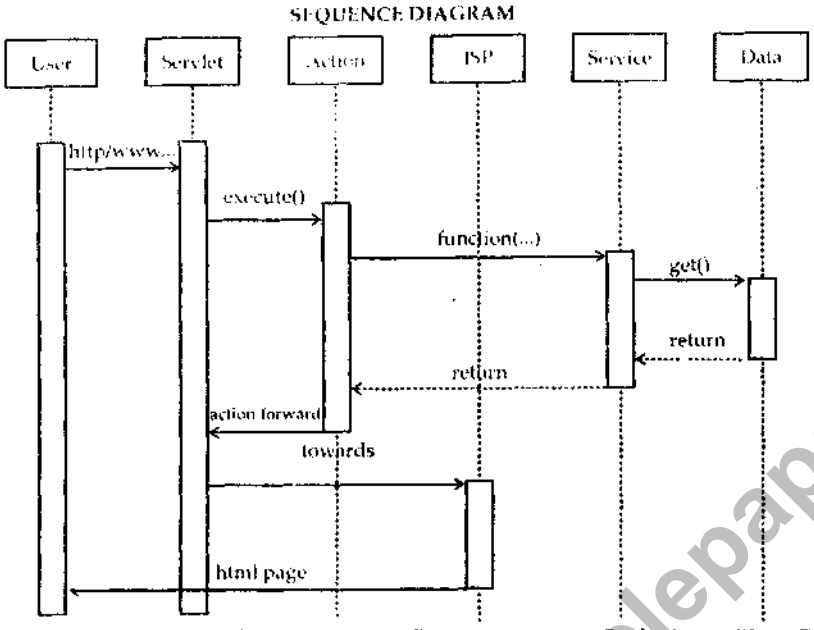
All incoming requests are intercepted by the Struts servlet controller. The Struts Configuration file struts-config.xml is used by the controller to determine the routing of the flow. This flows consists of an alternation between two transitions:

From View to Action	A user clicks on a link or submits a form on an HTML or JSP page. The controller receives the request, looks up the mapping for this request, and forwards it to an action. The action in turn calls a Model layer (Business layer) service or function
From Action to View	After the call to an underlying function or service returns to the action class, the action forwards to a resource in the View layer and a page is displayed in a web browser.

The diagram describes the flow in more detail:



www.btechsamplepapers.in



1. User clicks on a link in an HTML page.
2. Servlet controller receives the request, looks up mapping information in struts-config.xml, and routes to an action.
3. Action makes a call to a Model layer service.
4. Service makes a call to the Data layer (database) and the requested data is returned.
5. Service returns to the action.
6. Action forwards to a View resource (JSP page)
7. Servlet looks up the mapping for the requested resource and forwards to the appropriate JSP page.
8. JSP file is invoked and sent to the browser as HTML.
9. User is presented with a new HTML page in a web browser.

Struts Components

The Controller

This receives all incoming requests. Its primary function is the mapping of a request URI to an action class selecting the proper application module. It's provided by the framework.

The struts-config.xml File

This file contains all of the routing and configuration information for the Struts application. This XML file needs to be in the WEB-INF directory of the application.

Action Classes

It's the developer's responsibility to create these classes. They act as bridges between user-invoked URIs and business services. Actions process a request and return an ActionForward object that identifies the next component to invoke. They're part of the Controller layer, not the Model layer.

View Resources

View resources consist of Java Server Pages, HTML pages, JavaScript and Stylesheet files, Resource bundles, JavaBeans, and Struts JSP tags.

ActionForms

These greatly simplify user form validation by capturing user data from the HTTP request. They act as a "firewall" between forms (Web pages) and the application (actions). These components allow the validation of user input before proceeding to an Action. If the input is invalid, a page with an error can be displayed.

Model Components

The Struts Framework has no built-in support for the Model layer. Struts supports any model components:

- JavaBeans
- EJB
- CORBA
- JDO
- any other

Unlike the stored procedure and functions, which have to be called explicitly, the database triggers are fires (executed) or called implicitly whenever the table is affected by any of the above said DML operations.

Q.5(b). Struts framework

Ans. Till oracle 7.0 only 12 triggers could be associated with a given table, but in higher versions of Oracle there is no such limitation. A database trigger fires with the privileges of

www.btechsamplepapers.in

owner not that of user
A database trigger has three parts
1. A triggering event
2. A trigger constraint (Optional)
3. Trigger action

A triggering event can be an insert, update, or delete statement or a instance shutdown or startup etc. The trigger fires automatically when any of these events occur A trigger constraint specifies a Boolean expression that must be true for the trigger to fire. This condition is specified using the WHEN clause. The trigger action is a procedure that contains the code to be executed when the trigger fires.

Types of Triggers

The following are the different types of triggers.

Row triggers and statement triggers

A Row trigger fires once for each row affected. It uses FOR EACH ROW clause. They are useful if trigger action depends on number of rows affected.

Statement Trigger fires once, irrespective of number of rows affected in the table. Statement triggers are useful when triggers action does not depend on Before and afterTriggers

While defining the trigger we can specify whether to perform the trigger action (i.e. execute trigger body) before or after the triggering statement. BEFORE and AFTER triggers fired by DML statements can only be defined on tables.

BEFORE triggers The trigger action here is run before the trigger statement.

AFTER triggers The trigger action here is run after the trigger statement.

INSTEAD of Triggers provide a way of modifying views that can not be modified directly using DML statements.

LOGON triggers fires after successful logon by the user and LOGOFF trigger fires at the start of user logoff.

Points to ponder

- A trigger cannot include COMMIT, SAVEPOINT and ROLLBACK.
- We can use only one trigger of a particular type.
- A table can have any number of triggers.
- We use correlation names :new and :old can be used to refer to data in command line and data in table respectively.

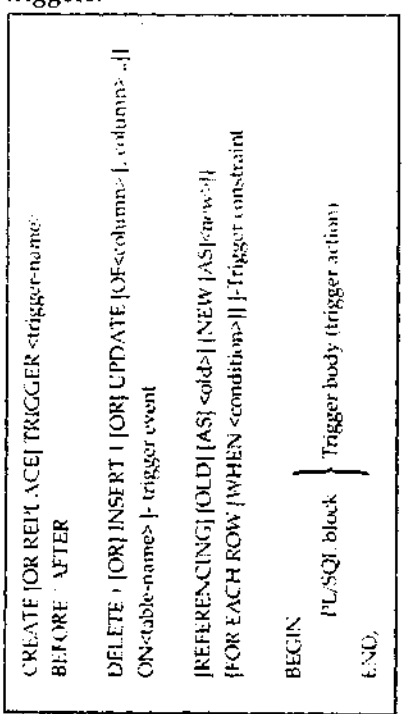
Triggers on DDL statements

DDL trigger are of the following types

BEFORE CREATE OR AFTER CREATE trigger is fired when a schema object is created. BEFORE OR AFTER ALTER trigger is fired when a schema object is altered. BEFORE OR AFTER DROP trigger is fired when a schema object is dropped.

A trigger can be enabled means can be made to run or it can disabled means it cannot run. A trigger is automatically enabled when it is created. We need re-enable trigger for using it if it is disabled. To enable or disable a trigger using ALTER TRIGGER command, you must be owner of the trigger or should have ALTER ANY TRIGGER privilege. To create a trigger you must have CREATE TRIGGER privilege, which is given to as part of RESOURCE privilege at the time of user creation.

Following figures give more understanding about triggers.



Q.5(c). Application specific database action.

Ans. What is JDBC

JDBC stands for Java Database Connectivity. This Java package provides tools for remote

www.btechsamplepapers.in

connection to a relational database, for running SQL queries on the database, and for retrieving and manipulating results of such queries. A program that uses JDBC acts as a client which is connecting to a database server. In our case you will be running your program on the machine puma, remotely connecting to the databsse server birch.

How to compile and run a JDBC program

A JDBC program uses a database driver to "communicate" with the database. The database driver is a program, usually already precompiled, which needs to installed on the client machine. We will use McKoi JDBC driver which you can download in the file mckoidb.jar In the examples below I assume that your program is in the file Myjdbc.java in the same directory as the driver.

Your program can be compiled as usual, for instance:

```
javac Myjdbc.java
```

If your program is an application, to run it, you need to type:

```
java -cp mckoidb.jar:. Myjdbc
```

Here Myjdbc is the class that you are running. The -cp variable specifies the class path: the path to the classes used in this program. The classes include the precompiled classes in mkoidb.jar and those in the current directory (your classes). The current directory is refered to as .(dot). The symbol : separates the two paths.

Example of using JDBC

The example below assumes that the database already exists and has a table Students.

```
import java.sql.*;
public class ConnectToServerDemo {
public static void main(String[] args) {
// Register the Mckoi JDBC Driver
try {
Class.forName("com.mckoi.JDBCDriver").
newInstance();
}
catch (Exception e) {
System.out.println(
"Unable to register the JDBC Driver.\n" +
"Make sure the JDBC driver is in the\n" +
"classpath.\n");
System.exit(1);
}
// This URL specifies we are connecting with
a database server
// on localhost.
String url = "jdbc:mckoi://birch/";
// The username / password to connect under.
String username = "cs349";
tring password = "cs349";
// Make a connection with the database.
Connection con;
try {
con = DriverManager.getConnection(url,
username, password);
}
catch (SQLException e) {
System.out.println(
"Unable to make a connection to the
database.\n" +
"The reason: " + e.getMessage());
System.exit(1);
return;
}
try {
Statement stmt = con.createStatement();
// using executeQuery():
ResultSet rs = stmt.executeQuery(
"SELECT * FROM Students");
// moving forward in the result set:
    while (rs.next()) {
int id = rs.getInt("ID");
String first = rs.getString("First_Name");
    String last = rs.getString("Last_Name");
    String year = rs.getString("Year");
    float gpa = rs.getFloat("GPA");
    Date date = rs.getDate("Date_of_birth");
  system.out.print(id + "  " + first + "  " + last
+ "  ");
  System.out.println(year + "  " + gpa + "  " +
date);
  }
      // Close the connection when finished
    con.close();
    }
    catch (SQLException e) {
      System.out.println(
        "An error occured\n" +
          "The SQLException message is: " +
e.getMessage());
      return;
    }
```

www.pastsamplepapers.in