AS-4227

B.Tech(7th Sem), Examination, 2013

Dept. of Computer Science & Engg

Sub: Web. Technology Maximum Marks: 60

Section-A

1. A search engine is a program to search
   a. For information
   b. Web pages
   c. Web pages for specified index terms
   d. Web pages for information using specified search terms
2. The tag used in HTML to link it with other URL's is:
   a. <A>
   b. <H>
   c. <U>
   d. <L>
3. What type of technology sends information to you without you requesting that information?
   a. F2b2C
   b. Infoware
   c. Push
   d. Wiki
4. What type of web technology provides news that can automatically download right to your desktop?
   a. Social Network
   b. RSS feed
   c. Podcast
   d. Wiki

5. Why a scripting language is called a scripting language?
    a. Scripting languages, like JavaScript and VBScript, are designed as an extension to XML
    b. Scripting languages, like JavaScript and VBScript, are designed as an extension to XHTML
    c. Scripting languages, like JavaScript and VBScript, are designed as an extension to HTML
    d. Scripting languages, like JavaScript and VBScript, are designed as an extension to DHTML
6. How to take request using post method?
    a. Using Request.form
    b. Using Response.Write
    c. Using Request.Querystring
    d. Using Request.Servervariables
7. Which of the following way can be used to indicate the LANGUAGE attribute?
    a. <LANGUAGE=”JavaScriptVersion”>
    b. <SCRIPT LANGUAGE =”JavaScriptVersion”>
    c. <SCRIPT LANGUAGE= “JavaSciptVersion”>
    JavaScript statements… </SCRIPT>
    d. <SCRIPT LANGUAGE=”JavaScriptVersion”!>
    JavaScript statements… </SCRIPT>
8. If you wanted to create a broadcast that could be downloaded from the Internet, what would you create?
    a. IBroadcast
    b. EShow
    c. Podcast
    d. ICast
9. Which of the following terms applies to all the web pages for Amazon.com?
    a. Top-level Domain
    b. Web site
    c. Web site address

d. Web Domain

10. Which of the following ISP is free?
    a. CompuServe
    b. AOL
    c. NetZero
    d. MSN

Section B

UNIT-1

Q2. What are search engines? Explain the working of search engines.

Ans. Search engines are programs that search documents for specified keywords and returns a list of the documents where the keywords were found.

Search engines are the key to finding specific information on the vast expanse of the World Wide Web. Without sophisticated search engines, it would be virtually impossible to locate anything on the Web without knowing a specific URL. But do you know how search engines work? And do you know what makes some search engines more effective than others?

When people use the term search engine in relation to the Web, they are usually referring to the actual search forms that searches through databases of HTML documents, initially gathered by a robot.

There are basically three types of search engines: Those that are powered by robots (called crawlers; ants or spiders) and those that are powered by human submissions; and those that are a hybrid of the two.

Crawler-based search engines are those that use automated software agents (called crawlers) that visit a Web site, read the information on the actual site, read the site's meta tags and also follow the links that the site connects to performing indexing on all linked Web sites as well. The crawler returns all that information back to a central depository, where the data is indexed. The crawler will periodically return to the sites to check for any information that has changed. The frequency with which this happens is determined by the administrators of the search engine.

Human-powered search engines rely on humans to submit information that is subsequently

indexed and catalogued. Only information that is submitted is put into the index.

In both cases, when you query a search engine to locate information, you're actually searching through the index that the search engine has created —you are not actually searching the Web. These indices are giant databases of information that is collected and stored and subsequently searched. This explains why sometimes a search on a commercial search engine, such as Yahoo! or Google, will return results that are, in fact, dead links. Since the search results are based on the index, if the index hasn't been updated since a Web page became invalid the search engine treats the page as still an active link even though it no longer is. It will remain that way until the index is updated.

So why will the same search on different search engines produce different results? Part of the answer to that question is because not all indices are going to be exactly the same. It depends on what the spiders find or what the humans submitted. But more important, not every search engine uses the same algorithm to search through the indices. The algorithm is what the search engines use to determine the relevance of the information in the index to what the user is searching for.

One of the elements that a search engine algorithm scans for is the frequency and location of keywords on a Web page. Those with higher frequency are typically considered more relevant. But search engine technology is becoming sophisticated in its attempt to discourage what is known as keyword stuffing, or spamdexing.

Another common element that algorithms analyze is the way that pages link to other pages in the Web. By analyzing how pages link to each other, an engine can both determine what a page is about (if the keywords of the linked pages are similar to the keywords on the original page) and whether that page is considered "important" and deserving of a boost in ranking. Just as the technology is becoming increasingly sophisticated to ignore keyword stuffing, it is also becoming more savvy to Web masters who build artificial links into their sites in order to build an artificial ranking.

•

A search engine is a tool that allows a user to enter keywords and retrieve information on websites contained in its catalog or database

•Search engine tools like Google are run by search engine software that allows the database to be

searched.

Features of Search Engine

•A true search engine is an automated software program that moves around the Web collecting WebPages to include in its catalog or database.

•It searches when an user requests information from a search engine; not the entire Web.

•Each search engine has its own catalog or database of collected WebPages, so you will get different results/hits by using different search engines.

Types of Search Engines

•

There are three main types of search engine tools:

–

Search directories or indexes

–

Hybrid search engines

–

Meta search engines

•All of these tools are often referred to as search engines.

A meta search engine is a tool that helps to locate information available via the WWW.

•

It provide a single interface that enables users to search many different search engines, indexes and databases.

•

Thus Meta search engines are capable of searching several search engine databases at once.

The Three Parts of a Search Engine

•

Spider, crawler or robot

•

Index, catalog or database

•

Search engine software

•

The second part of a search engine is called the index, catalog, or database.

•

This index contains a copy of each page that was collected by the spider.

• A spidered page must be indexed to become a search result.

•

When a user requests keywords from a search engine, the search engine software sifts through all the indexed pages to find matching keywords, then returns the results/hits to the user.

Part Three: Search Engine Software

Q3. Explain the terms quality assurance and testing with examples in connection with web development.

Ans. The term **quality assurance**, when applied to web sites, describes the process of enforcing **quality *control*** standards and working to improve the processes that are used in producing the web site and its components, infrastructure and content. When quality assurance is well implemented, a web site should see progressive improvement in terms of both lessening rate of defects *and* general increase in site usability and performance.

Quality assurance should function as a "voice" for the user, a reminder to the designers and developers that the site is designed for users outside the office. Quality assurance as ombudsman would be a positive force for a quality user experience.

If you are limited in what you can accept responsibility for, document those limits. For example, if you can't test data or middleware, announce that fact whenever you provide test results for the quality of the site. Even the best designed and developed sites will experience problems and failures, so a good quality assurance team should set expectations — for the entire web site team *and* with management — for what QA can effectively accomplish.

Focus on Improving Processes

The key to understanding quality assurance is understanding the emphasis on **process**: quality control focuses on what comes out of the web site creation process (creation, development, publication — whatever term you prefer that describes the process that results in the web site). Quality assurance focuses on what goes *in*to the creation process as well as on the *process itself* with the goal of improving the quality of output by improving everything "downstream".

Quality assurance looks beyond the structured testcases used by quality control because these testcases are necessarily limited. Quality assurance focuses on more than a site's ability to meet a specific benchmark; quality assurance aims to make the site better so tests are passed more consistently, so that the benchmark can in fact be refined, and so that problem areas can be eliminated.

Quality assurance should be involved in the development process. QA should review new designs *before* they are finalized with an eye towards usability and user experience factors; heading of problems before they become real improves quality immediately and reduces problems "downstream".

Quality assurance should be involved in customer service and user-support communications, especially with a commerce site, so that usability defects can be reviewed. With user input, QA can refine user scenarios to better match "real" behavior. There is no substitute for user comments.

Are the tools used to create and maintain your web site appropriate for their tasks? Can the tools be tweaked to shorten the processes or eliminate some steps? Can some tests be incorporated earlier in the creation process, such as spell checking? If your site has changing content, is the content checked after it is published, or before it is entered into the database? Quality assurance should pay attention to all of these issues.

Following any major phase of your web site, perform a postmortem analysis: review the success of the changes, redesigns, scheduling, file transfers, etc. What could be made more efficient? Which processes could be smoothed out?

Focus on Tracking Problems

Quality assurance also involves a closer involvement with defects and their resolution. During the quality control process, problems are discovered and typically reported and handed off to the people who "own" the work with the defects. Quality control can be a binary process:

something passes, or it fails and is “bounced” back to the team responsible for fixing it enough to be tested again.

Quality assurance catches the problems discovered through use of quality control testcases, but also finds problems uncovered through more general site reviews and ad hoc usability and consistency testing. In addition, quality assurance testing finds areas for improvements that may not be defects, but rather opportunities; user input is a great source for such “opportunities”. The set of problems returned by thorough quality assurance testing is therefore larger than the set found through quality control testing. The handling of this larger set of problems is a major function of QA.

Quality assurance should log reported problems in a database of some kind, assigning properties to the problem such as the priority and scope, and recording such attributes as description, error message, affected functionality, etc. In addition, QA should assign and track **ownership** of the problem, and should track the progress made towards**resolution** of the problem. Quality assurance must take an active role in getting problems fixed; demanding schedules for the fixes, explanations for the problems, working to eliminate the type of problem in the future — these are all common actions.

The success of software engineering depends upon the delivery of high quality software. Quality is one of the key factors in the market growth and success of a product. In recent years, quality of software product and quality in service has become principles for many corporations and organizations to distinguish themselves from competitors and to cover larger market place. Quality is an ambiguous word; means there are lot of definitions available for quality. the customer satisfaction in mathematical equation as –

“Customer Satisfaction = Compliant Product + Good Quality + Delivery within Schedule an Budget”.

Software quality assurance include the following important elements.

- Quality management tools.
- Effective software engineering methods and tools.
- Formal technical reviews applicable to whole software process.
- Effective testing strategies and techniques.
- Procedures to control documentations and changes in it.
- Procedures to assure compliance to standards.
- Mechanisms for measurement and reporting.

QUALITY ASSURANCE ENABLING TECHNOLOGIES

In order to build reliable and high quality Web applications the Web engineer should be familiar with the quality assurance enabling technologies.

The brief description of these enabling technologies is as under:

• Component based Development: The explosive growth of Web-based applications has evolved the component technologies. The available famous infrastructure standards for web development are CORBA (Common Object Request Broker Architecture), COM/DCOM
(Component Object Model/ Distributed Component Object Model) and JavaBeans.

Internet Standards: Internet standards are specifications which are stable and wellunderstood, has multiple, independent, and interoperable implementations with operational experience and recognizably useful in some or all elements of the Internet. In early 1990's HTML (Hyper Text Markup Language) was the dominant standard to develop Web applications but now the Web applications have become more complex, their size is growing, therefore new standards have emerged.

Security: When our application deployed and launched on the network or Internet, there are great risks of unauthorized used. There are great threats of vulnerabilities. The hackers try to unauthorized access in the intent of some profit or for some other aims. Sometimes internal personnel can be involved in unauthorized access of particular application for their specific benefits and aims or malicious intents.

PURPOSE OF INTEGRATION TESTING

The purpose of integration testing is to make sure that modules and their interfaces in an application interact with each other in a correct and secure way. Basically integration testing is based on functional requirements specification and design which are used as an input in integration testing process.

Integration testing covers following types of concerning areas during integrating different modules:

• Calls of different software components, while interacting to each other

• Data and information sharing between the modules in proper manners

• Compatibility, which ensures one module that does not effect on the performance and functionality of the other modules.

• Non-functional issues

Q4. Write the difference between web browser and search engine.

Ans. A web browser is usually a software program that runs on your computer while a search engine is a software program that runs on a server somewhere else. A browser normally provides access to a specific search while a search engine stores information and indexes for searchers to find. There are less browsers than there are search engines.

a browser is a program installed on your PC locally (IE for most people, firefox , safari, konqueror, opera, there are loads of them) that is used to access the Internet, to view what there is, as it were.

A search engine is a program, that in varying ways aggregates reference data so that when you type in a phrase it can point you in the direction of a website that relates to the words you type in.

A browser is a program installed on your PC locally (IE for most people, firefox , safari, konqueror, opera, there are loads of them) that is used to access the Internet, to view what there is, as it were.

A search engine is a program, that in varying ways aggregates reference data so that when you type in a phrase it can point you in the direction of a website that relates to the words you type in.

You use a browser to get to a search engine..such as typing www.google.com - that takes you to the google web page which allows you to enter information that it then utilizes its search engine to cross reference with what it has found out on the web to point you to links to those sites which seem pertinent according to its formulaes (wrong word but general idea) (though these days many browsers have search engine access built into the browser window, blurring the lines a little and understandably causing a source of confusion to the uninitiated).

web browser such as Microsoft Internet Explorer, Mozilla Firefox, Apple Safari, Netscape, and Opera is a software application that enables a user to display and interact with text, images, and other information typically located on a web page at a website on the World Wide Web or a local area network. Web browsers allow a user to quickly and easily access information provided on web pages at websites by traversing these links.

A search engine is a program designed to help find information stored on a computer system

such as the World Wide Web, inside a corporate network or a personal computer. The search engine allows one to ask for content meeting specific criteria (typically those containing a given word or phrase) and retrieves a list of references that match those criteria. Search engines use regularly updated indexes to operate quickly and efficiently. Search engine usually refers to a Web search engine such as Google, Yahoo! Search, MSN Search, and Ask.com, which searches for information on the public Web. Other kinds of search engines are enterprise search engines, which search on intranets, personal search engines, which search individual personal computers, and mobile search engines.

UNIT-2

Q5. What is the classification of objects?

Ans.

Classification of web page content is essential to many tasks in web information retrieval such as maintaining web directories and focused crawling. The uncontrolled nature of web content presents additional challenges to web page classification as compared to traditional text classification, but the interconnected nature of hypertext also provides features that can assist the process. Classification plays a vital role in many information management and retrieval tasks. On the Web, classification of page content is essential to focused crawling, to the assisted development of web directories, to topic-specific web link analysis, and to analysis of the topical structure of the Web. Web page classification can also help improve the quality of web search.

Everything is an Object

In JavaScript almost everything is an object. Even primitive datatypes (except null and undefined) can be treated as objects.

- Booleans can be objects or primitive data treated as objects
- Numbers can be objects or primitive data treated as objects
- Strings are also objects or primitive data treated as objects
- Dates are always objects
- Maths and Regular Expressions are always objects
- Arrays are always objects
- Even functions are always objects

Q6. What is WWW? Discuss in brief.

Ans. The **World Wide Web** (abbreviated as **WWW** or **W3**,[3] commonly known as **the web**) is a system of interlinked hypertext documents accessed via the Internet.

With a web browser, one can view web pages that may contain text, images, videos, and other multimedia and navigate between them via hyperlinks.

Tim Berners-Lee, a British computer scientist and at that time employee of CERN, a European research organisation near Geneva,[4] wrote a proposal in March 1989 for what would eventually become the World Wide Web.[1] The 1989 proposal was meant for a more effective CERN communication system but Berners-Lee eventually realised the concept could be implemented throughout the world.[5] Berners-Lee and Flemish computer scientist Robert Cailliau proposed in 1990 to use hypertext "to link and access information of various kinds as a web of nodes in which the user can browse at will",[6] and Berners-Lee finished the first website in December that year.[7] Berners-Lee posted the project on the alt.hypertext newsgroup on 7 August 1991.[8]

In the May 1970 issue of *Popular Science* magazine, Arthur C. Clarke predicted that satellites would someday "bring the accumulated knowledge of the world to your fingertips" using a console that would combine the functionality of the photocopier, telephone, television and a small computer, allowing data transfer and video conferencing around the globe.[9]

In March 1989, Tim Berners-Lee wrote a proposal that referenced ENQUIRE, a database and software project he had built in 1980, and described a more elaborate information management system.[10]

With help from Robert Cailliau, he published a more formal proposal (on 12 November 1990) to build a "Hypertext project" called "WorldWideWeb" (one word, also "W3") as a "web" of "hypertext documents" to be viewed by "browsers" using a client–server architecture.[6] This proposal estimated that a read-only web would be developed within three months and that it would take six months to achieve "the creation of new links and new material by readers, [so that] authorship becomes universal" as well as "the automatic notification of a reader when new material of interest to him/her has become available." While the read-only goal was met,

accessible authorship of web content took longer to mature, with the wiki concept, blogs, Web 2.0 and RSS/Atom.[11]

The proposal was modeled after the SGML reader Dynatext by Electronic Book Technology, a spin-off from the Institute for Research in Information and Scholarship at Brown University. The Dynatext system, licensed by CERN, was a key player in the extension of SGML ISO 8879:1986 to Hypermedia within HyTime, but it was considered too expensive and had an inappropriate licensing policy for use in the general high energy physics community, namely a fee for each document and each document alteration.

The terms Internet and World Wide Web are often used in everyday speech without much distinction. However, the Internet and the World Wide Web are not the same. The Internet is a global system of interconnectedcomputer networks. In contrast, the web is one of the services that runs on the Internet. It is a collection of text documents and other resources, linked by hyperlinks and URLs, usually accessed by web browsers from web servers. In short, the web can be thought of as an application "running" on the Internet.[29]

Viewing a web page on the World Wide Web normally begins either by typing the URL of the page into a web browser or by following a hyperlink to that page or resource. The web browser then initiates a series of communication messages, behind the scenes, in order to fetch and display it. In the 1990s, using a browser to view web pages—and to move from one web page to another through hyperlinks—came to be known as 'browsing,' 'web surfing,' or 'navigating the web'. Early studies of this new behavior investigated user patterns in using web browsers. One study, for example, found five user patterns: exploratory surfing, window surfing, evolved surfing, bounded navigation and targeted navigation.[30]

The following example demonstrates how a web browser works. Consider accessing a page with the URL `http://example.org/wiki/World_Wide_Web`.

First, the browser resolves the server-name portion of the URL (*example.org*) into an Internet Protocol address using the globally distributed database known as

the Domain Name System (DNS); this lookup returns an IP address such as *208.80.152.2*. The browser then requests the resource by sending an HTTP request across the Internet to the computer at that particular address. It makes the request to a particular application port in the underlying Internet Protocol Suite so that the computer receiving the request can distinguish an HTTP request from other network protocols it may be servicing such as e-mail delivery; the HTTP protocol normally uses port 80. The content of the HTTP request can be as simple as the two lines of text GET */wiki/World_Wide_Web* HTTP/1.1 Host: *example.org*

The computer receiving the HTTP request delivers it to web server software listening for requests on port 80. If the web server can fulfill the request it sends an HTTP response back to the browser indicating success, which can be as simple as HTTP/1.0 200 OK Content-Type: text/html; charset=UTF-8 followed by the content of the requested page. The Hypertext Markup Language for a basic web page looks like <html> <head> <title>Example.org – The World Wide Web</title> </head> <body> The World Wide Web, abbreviated as WWW and commonly known ... </body> </html>

The web browser parses the HTML, interpreting the markup (`<title>`, `<p>` for paragraph, and such) that surrounds the words in order to draw the text on the screen.

Many web pages use HTML to reference the URLs of other resources such as images, other embedded media, scripts that affect page behavior, and Cascading Style Sheets that affect page layout. The browser will make additional HTTP requests to the web server for these other Internet media types. As it receives their content from the web server, the browser progressively renders the page onto the screen as specified by its HTML and these additional resources.

Q7. What is object manipulation? Give answer with suitable example.

Ans.

Creating JavaScript Objects

Almost "everything" in JavaScript can be objects. Strings, Dates, Arrays, Functions....

You can also create your own objects.

This example creates an object called "person", and adds four properties to it:

Example

```
person=new                                    Object();
person.firstname="John";
person.lastname="Doe";
person.age=50;
person.eyecolor="blue";
```

There are many different ways to create new JavaScript objects, and you can also add new properties and methods to already existing objects.
You will learn much more about this in a later chapter of this tutorial.

Accessing Object Properties

The syntax for accessing the property of an object is:

```
objectName.propertyName
```

This example uses the length property of the String object to find the length of a string:

```
var                          message="Hello                          World!";
var x=message.length;
```

The value of x, after execution of the code above will be:

```
12
```

Accessing Object Methods

You can call a method with the following syntax:

```
objectName.methodName()
```

This example uses the toUpperCase() method of the String object, to convert a text to uppercase:

```
var                          message="Hello                          world!";
var x=message.toUpperCase();
```

The value of x, after execution of the code above will be:

```
HELLO WORLD!
```

UNIT-3

Q8. How can we create a 4X10 table and set the different color in each block of table?

Ans.

color names can also be used instead of the color code

```
<html>
<head>
</head>
<body>
<table>
<tr>
<td bgcolor="#0000"></td>
<td bgcolor="#0001"></td>
<td bgcolor="#0002"></td>
<td bgcolor="#0003"></td>
<td bgcolor="#0004"></td>
<td bgcolor="#0005"></td>
<td bgcolor="#0006"></td>
<td bgcolor="#0007"></td>
<td bgcolor="#0008"></td>
<td bgcolor="#0009"></td>
</tr>
<tr>
<td bgcolor="#000A"></td>
<td bgcolor="#000B"></td>
<td bgcolor="#000C"></td>
<td bgcolor="#000D"></td>
<td bgcolor="#000E"></td>
<td bgcolor="#000F"></td>
<td bgcolor="#0010"></td>
<td bgcolor="#0011"></td>
<td bgcolor="#0012"></td>
<td bgcolor="#0013"></td>
</tr>
<tr>
<td bgcolor="#0014"></td>
```

```
<td bgcolor="#0015"></td>
<td bgcolor="#0016"></td>
<td bgcolor="#0017"></td>
<td bgcolor="#0018"></td>
<td bgcolor="#0019"></td>
<td bgcolor="#001A"></td>
<td bgcolor="#001B"></td>
<td bgcolor="#001C"></td>
<td bgcolor="#001D"></td>
</tr>
<tr>
<td bgcolor="#001E"></td>
<td bgcolor="#001F"></td>
<td bgcolor="#0020"></td>
<td bgcolor="#0021"></td>
<td bgcolor="#0022"></td>
<td bgcolor="#0023"></td>
<td bgcolor="#0024"></td>
<td bgcolor="#0025"></td>
<td bgcolor="#0026"></td>
<td bgcolor="#0027"></td>
</tr>
</table>
</body>
</html>
```

Q9. Write the importance of CSS? Give an example.

Ans. Cascading Style Sheets, better known as CSS, have become a staple tool in web design. In short, CSS is the way in which web designers separate the presentation or design of a web site from the structure or the markup. By using CSS a web designer can built his web site so that any style or presentation changes he wants to make in the future can all be handled in one place. This consolidation of resources is not only helpful to the web design professional but also to the site visitor as well.

When you use CSS the idea is to generate very clean HTML markup that has absolutely no built-in styling. That means when you write a heading you use the < h1 > tag alone with no extra tags to define the heading's font, color, size, etc. Then you built a separate file with a

.css extension where you define tag styles. In your CSS file, you would then tell the browser how the < h1 > tag should look. Back on your HTML file you create a link to the CSS file and then any changes you make on that CSS file in the future will instantly be reflected on your site without having to change any of the site's HTML.

The added bonus of using a separate CSS file is in bandwidth and load times. Most modern browsers will cache any files with a .css extension. This is a nice bonus because it means the visitors to your site will have less to download and make your site appear faster. If you make a change to the CSS file, the browser will be smart enough to recognize the difference and reload the cached copy it has on file. This way you never have to worry about people seeing an older version of your site's design.

CSS can even be used to style other sites you don't control as well. Some browserds have extensions that have been developed to allow you to overwrite a websites style with a CSS file on your local machine. This is especially common with some of the more popular web applications like free email services from the major providers.

Cascading Style Sheets are practically a must for any serious web design professional. There is no excuse to not use CSS in your site designs especially when you consider all of the obvious benefits involved. CSS makes your life easier by increasing the speed and efficiency in which you can update your site's appearance.

Cascading style sheets (CSS) is a standard defined by the World Wide Web Consortium that offers designers more flexibility and accuracy when defining the appearance of text and formats than standard HTML. Essentially, CSS allows designers to manipulate the appearance of the webpage without affecting its HTML structure .For example, if you wanted to change all the text in your document to blue, and all the headlines to green, with standard HTML one would have to manually change the elements on the page one by one. Significance of Cascading Style Sheets

If used appropriately, cascading style sheets can be quite beneficial to the web designer. In addition to text, cascading style sheets can define spacing between lines, the size of the type in pixels instead of points, and define specific fonts within pages.

Formatting each individual page becomes very cumbersome for the designer, especially if their site contains hundreds of pages. With style sheets, one only needs to specify such preferences once, and the style can be applied to an entire site. And if the designer decides to change the width of the page, then he or she only needs to change this preference in one place, rather than having to search through all of the pages to change the HTML (Web Design Group, 1997).

Style sheets offer flexibility in terms of the presentation effects that they provide. Properties such as color, background, margin, border, and many more can be applied to Significance of Cascading Style Sheets

If used appropriately, cascading style sheets can be quite beneficial to the web designer. In addition to text, cascading style sheets can define spacing between lines, the size of the type in pixels instead of points, and define specific fonts within pages. Formatting each individual page becomes very cumbersome for the designer, especially if their site contains hundreds of pages. With style sheets, one only needs to specify such preferences once, and the style can be applied to an entire site. And if the designer decides to change the width of the page, then he or she only needs to change this preference in one place, rather than having to search through all of the pages to change the HTML (Web Design Group, 1997).

Style sheets offer flexibility in terms of the presentation effects that they provide.

Properties such as color, background, margin, border, and many more can be applied to Significance of Cascading Style Sheets

If used appropriately, cascading style sheets can be quite beneficial to the web designer. In addition to text, cascading style sheets can define spacing between lines, the size of the type in pixels instead of points, and define specific fonts within pages.

Formatting each individual page becomes very cumbersome for the designer, especially if their site contains hundreds of pages. With style sheets, one only needs to specify such preferences once, and the style can be applied to an entire site. And if the designer decides to change the width of the page, then he or she only needs to change this preference in one place, rather than having to search through all of the pages to change the HTML (Web Design Group, 1997).

Style sheets offer flexibility in terms of the presentation effects that they provide.

Properties such as color, background, margin, border, and many more can be applied to all elements. With just HTML and its proprietary extensions, one must rely on attributes

like **BGCOLOR**, which are only available for a few elements. Style sheets give the flexibility of applying a style to all paragraphs, or all level-two headings, or all

emphasized text (Web Design Group, 1997).

However, there is a down side to using CSS. Only Netscape 4.0+, 6.0+ and

Explorer 4 and 5 support CSS. Some of these browsers don't offer full support for the entire CSS1 specification (Green, 2002). For instance, if someone were to look at stylebased pages in an older browser, they would not see any formatting beyond default colors, sizes, fonts and positions.

Q10. What is frameset? How would you give space between framesets?
Ans.

Following are important attributes of <frameset> and should be known to you to use frameset.

- **cols:** specifies how many columns are contained in the frameset and the size of each column. You can specify the width of each column in one of four ways:
    - Absolute values in pixels. For example to create three vertical frames, use*cols="100, 500,100"*.
    - A percentage of the browser window. For example to create three vertical frames, use *cols="10%, 80%,10%"*.
    - Using a wildcard symbol. For example to create three vertical frames, use*cols="10%, *,10%"*. In this case wildcard takes remainder of the window.
    - As relative widths of the browser window. For example to create three vertical frames, use *cols="3*,2*,1*"*. This is an alternative to percentages. You can use relative widths of the browser window. Here the window is divided into sixths: the first column takes up half of the window, the second takes one third, and the third takes one sixth.
- **rows:** attribute works just like the cols attribute and can take the same values, but it is used to specify the rows in the frameset. For example to create two horizontal frames, use *rows="10%, 90%"*. You can specify the height of each row in the same way as explained above for columns.
- **border:** attribute specifies the width of the border of each frame in pixels. For example border="5". A value of zero specifies that no border should be there.

- **frameborder:** specifies whether a three-dimensional border should be displayed between frames. This attrubute takes value either 1 (yes) or 0 (no). For example frameborder="0" specifies no border.
- **framespacing:** specifies the amount of space between frames in a frameset. This can take any integer value. For example framespacing="10" means there should be 10 pixels spacing between each frames.
- The <frameset> tag defines a frameset.
- The <frameset> element holds one or more <frame> elements. Each <frame> element can hold a separate document.
- The <frameset> element specifies HOW MANY columns or rows there will be in the frameset, and HOW MUCH percentage/pixels of space will occupy each of them.
- Example
- A simple three-framed page:
- <frameset cols="25%,*,25%">

```
<frame src="frame_a.htm">
<frame src="frame_b.htm">
<frame src="frame_c.htm">
</frameset>
```

UNIT-4

Q11. What are the steps to run the ASP program with local host?

Ans.

ASP can run by following 3 methods :

1) INSTALL IIS
2) Open the Control Panel from the Start menu
3) Double-click Programs and Features
4) Click "Turn Windows features on or off" (a link to the left)
5) Select the check box for Internet Information Services (IIS), and click OK

After you have installed IIS, make sure you install all patches for bugs and security problems. (Run Windows Update).

After you have installed IIS or PWS follow these steps:

1. Look for a new folder called **Inetpub** on your hard drive
2. Open the Inetpub folder, and find a folder named **wwwroot**
3. Create a new folder, like "MyWeb", under wwwroot
4. Write some ASP code and save the file as "test1.asp" in the new folder
5. Make sure your Web server is running (see below)
6. Open your browser and type "http://localhost/MyWeb/test1.asp", to view your first web page

Your Windows PC as a Web Server

- Your own PC can act as a web server if you install IIS or PWS
- IIS or PWS turns your computer into a web server
- Microsoft IIS and PWS are free web server components

IIS - Internet Information Server

IIS is a set of Internet-based services for servers created by Microsoft for use with Microsoft Windows.

IIS comes with Windows 2000, XP, Vista, and Windows 7. It is also available for Windows NT.

IIS is easy to install and ideal for developing and testing web applications.

PWS - Personal Web Server

PWS is for older Windows system like Windows 95, 98, and NT.

PWS is easy to install and can be used for developing and testing web applications including ASP.

We don't recommend running PWS for anything else than training. It is outdated and has security issues.

How to Install IIS on Windows 7 and Windows Vista

Follow these steps to install IIS:

1. Open the Control Panel from the Start menu
2. Double-click Programs and Features
3. Click "Turn Windows features on or off" (a link to the left)

4. Select the check box for Internet Information Services (IIS), and click OK

After you have installed IIS, make sure you install all patches for bugs and security problems. (Run Windows Update).

How to Install IIS on Windows XP and Windows 2000

Follow these steps to install IIS:

1. On the Start menu, click Settings and select Control Panel
2. Double-click Add or Remove Programs
3. Click Add/Remove Windows Components
4. Click Internet Information Services (IIS)
5. Click Details
6. Select the check box for World Wide Web Service, and click OK
7. In Windows Component selection, click Next to install IIS

After you have installed IIS, make sure you install all patches for bugs and security problems. (Run Windows Update).

Test Your Web

After you have installed IIS or PWS follow these steps:

1. Look for a new folder called **Inetpub** on your hard drive
2. Open the Inetpub folder, and find a folder named **wwwroot**
3. Create a new folder, like "MyWeb", under wwwroot
4. Write some ASP code and save the file as "test1.asp" in the new folder
5. Make sure your Web server is running (see below)
6. Open your browser and type "http://localhost/MyWeb/test1.asp", to view your first web page

Note: Look for the IIS (or PWS) symbol in your start menu or task bar. The program has functions for starting and stopping the web server, disable and enable ASP, and much more.

Q12. What are ASP applications ? Explain in brief.

Ans.

Dynamically edit, change, or add any content of a Web page

- Respond to user queries or data submitted from HTML forms
- Access any data or databases and return the results to a browser
- Customize a Web page to make it more useful for individual users

- The advantages of using ASP instead of CGI and Perl, are those of simplicity and speed
- Provide security - since ASP code cannot be viewed from the browser
- Clever ASP programming can minimize the network traffic

Q13. What is subroutine and arguments? Explain in brief.

Ans.

In computer programming, a **subroutine** is a sequence of program instructions that perform a specific task, packaged as a unit. This unit can then be used in programs wherever that particular task should be performed. Subprograms may be defined within programs, or separately in libraries that can be used by multiple programs.

In different programming languages a subroutine may be called a **procedure**, a **function**, a **routine**, a **method**, or a **subprogram**. The generic term**callable unit** is sometimes used.[1]

As the name *subprogram* suggests, a subroutine behaves in much the same way as a computer program that is used as one step in a larger program or another subprogram. A subroutine is often coded so that it can be started (called) several times and/or from several places during one execution of the program, including from other subroutines, and then branch back (*return*) to the next instruction after the *call* once the subroutine's task is done.

Subroutines are a powerful programming tool,[5] and the syntax of many programming languages includes support for writing and using them. Judicious use of subroutines (for example, through the structured programming approach) will often substantially reduce the cost of developing and maintaining a large program, while increasing its quality and reliability.[6] Subroutines, often collected into libraries, are an important mechanism for sharing and trading software. The discipline of object-oriented programming is based on objects and methods (which are subroutines attached to these objects or objectclasses).

ARGUMENT :

An argument is something that is passed in to a function. It's the same thing as a parameter. For example: int main(int argc, char **argv);
"argc" and "argv" are "arguments" being passed in to the main function.

Arguments are variables that can be used by the function receiving them.

Q14. Write the importance of XML.

Ans. **XML** is a standard for *describing* how information is structured. This makes it much easier to move structured information from place to place, or from one program to another. However, what it doesn't do is *specify* how information should be structured, beyond a few basic precepts.

So it doesn't (as some would have you believe) solve all the world's problems. If my invoicing system has a very different idea from yours, for example, about the information that should be held on a given customer, then putting it into XML is not going to magically fill in the missing bits. Similarly, if you store a customer's full name, and I want to store first name, middle initial, and surname, then XML will not tell you how to split up your customer's names. Data conversion between formats can still be fiddly and tedious.

However, if you dismiss XML because of this, then you miss an important part of the picture. XML is a standard language for talking about data. It performs a very similar function to ASCII (a standard way of encoding characters as bytes) or SQL (a standard way of asking questions of a database). Having it is an important step on the ladder. Many years ago there was a time, before ASCII, when any text transmitted as a sequence of bytes from one computer to another might not be understood because the two computers didn't agree on which number should represent the letter 'A'. Thankfully those days are now largely behind us. Programmers no longer have to think about the conversions before they even start looking at the contents.

XML performs a similar function when exchanging data which has a structure more complex than simply a string of text. There are standard instructions that XML programmers can use to find the third sub-item of a group, in much the same way that there are standard calls to find the length of a piece of ASCII text. Programmers no longer have to think about how to decode the structure before they even look at the contents. XML certainly isn't perfect, and there will always be situations where system designers will want to create their own custom-built alternative. But in the past they had little choice. Now, you can just use XML unless you have a good reason not to. And if you do, a whole world of tools, utilities, conventions, and libraries of code are available to help you along.

XML has another important function, though. While it doesn't specify how information on a particular topic should be structured, it does provide a syntax for writing such specifications, called XML Schemas or DTDs (document type definitions). If I have a music-composing program which can output XML conforming to a particular schema, and you have a music-printing program which can read XML conforming to the same schema, *then* we have

understanding about the way the information is structured. And because XML is an open standard which isn't owned by anybody, there's less incentive for people to feel proprietary about the way that their data is stored in it. So, slowly but surely, people are starting to agree on schemas for storing and transmitting information on particular topics. This should mean that in future, your customers' names are more likely to be stored in a similar way to mine, and transferring information between us should be easier.

In summary, here's an analogy. Imagine that you have to organise a large meeting for lots of people from different companies and different cultures. They might all disagree on which items should be on the agenda. They might also disagree on how such meetings should be organised. But you are never going to make any progress on either of these if the participants all speak different languages. Once you agree on a common language, you can concentrate on higher-level things.

That's the problem XML is addressing, and that's why it's so important.

The computing press has found a new savior for the ills that afflict computing and the web: XML. XML is new, it's exciting, and it's got to be good, because the specification for it looks indecipherable. XML's hype level has already drawn fire from some quarters, from those accusing it of 'balkanizing the web' or of increasing the load on an already strained Internet. Most important, many developers are wondering why exactly they need to learn yet another language.

XML's set of tools allows developers to create web pages - and much more. XML allows developers to set standards defining the information that should appear in a document, and in what sequence. XML, in combination with other standards, makes it possible to define the content of a document separately from its formatting, making it easy to reuse that content in other applications or for other presentation environments. Most important, XML provides a basic syntax that can be used to share information between different kinds of computers, different applications, and different organizations without needing to pass through many layers of conversion.

Web developers are the initial target audience, but database developers, document managers, desktop publishers, programmers, scientists, and other academics are all getting involved. XML provides a simple format that is flexible enough to

accommodate wildly diverse needs. Even developers performing tasks on different types of applications with different interfaces and different data structures can share XML formats and tools for parsing those formats into data structures that applications can use. XML offers its users many advantages, including:

- Simplicity
- Extensibility
- Interoperability
- Openness
- A core of experienced professionals

Simplicity

XML provides both programmers and document authors with a friendly environment, at least by computing standards. XML's rigid set of rules helps make documents more readable to both humans and machines. XML document syntax contains a fairly small set of rules, making it possible for developers to get started right away. DTDs can be developed through a standards process, set by experts, or through experimentation, based on the structures of documents that seem to work well. XML parsers are also reasonably simple to build, especially parsers that only check well-formedness.

XML documents are built upon a core set of basic nested structures. While the structures themselves can grow complex as layers and layers of detail are added, the mechanisms underlying those structures require very little implementation effort, from either authors or developers. These basic structures can be used to represent complex sets of information, from the full contents of a document to persistent object state information to a set of commands for a program, without needing to change the structures themselves.

Extensibility

XML is extensible in two senses. First, it allows developers to create their own DTDs, effectively creating 'extensible' tag sets that can be used for multiple applications. Second, XML itself is being extended with several additional standards that add

styles, linking, and referencing ability to the core XML set of capabilities. As a core standard, XML provides a solid foundation around which other standards may grow.

Creating DTDs is most likely what the creators of XML had in mind when they called it Extensible Markup Language. XML is, after, a meta-language, a set of rules that can be used to create sets of rules for documents. In a certain sense, there's no such thing as an 'XML document' - all the documents that use XML-compliant syntax are really using applications of XML, with tag sets chosen by their creators for that particular document. XML's facilities for creating DTDs give standard-builders a set of tools for specifying what document structures may or must appear in a document, making it easy to define sets of structures. These structures can then be used with XML tools for authoring, parsing, and processing, and used by applications as a guide to the data they should accept.

At the same time that XML is being used to create other standards, other supporting standards for XML are being defined. XML can already use many of the standards applied to HTML, like Cascading Style Sheets (CSS) and HyperText Transfer Protocol (HTTP). W3C working groups are developing additional supporting standards for XML. XML-Linking (XLink) provides linking facilities that are far more sophisticated than those in HTML. XPointers, derived from the Text Encoding Initiative's (TEI) extended pointers, provide a way to consistently reference portions of documents. Extensible Style Language (XSL) provides a more complete set of formatting tools than CSS, and is notable for using XML syntax to define its style sheets. Other standards, including support for data-typing, are under discussion.

Interoperability

XML can be used on a wide variety of platforms and interpreted with a wide variety of tools. Because the document structures behave consistently, parsers that interpret them can be built at relatively low cost in any of a number of languages. XML supports a number of key standards for character encoding, allowing it to be used all over the world in a number of different computing environments. XML complements Java, another force for interoperability, very well, and a considerable amount of early

XML development has been in Java. A generic application programming interface (API) for parsers, the Simple API in XML (SAX), is freely available. Parsers are also available in C++, C, JavaScript, Tcl, and Python, with more on the way. XML parser development so far has focused on freeware plug-ins that provide parsing capabilities to XML applications, greatly lowering the cost of building XML-enabled applications.

Openness

Although there have been some questions about the process used to create XML, the standard itself is completely open, freely available on the web. The W3C members have early access to standards (and, apart from invited experts, are the only ones who can participate directly in their creation), but once the standard is complete the results are public. The XML Working Group and the Working Groups for the supporting standards also release drafts of their work on a regular basis, making it possible to follow work in progress. Several non-W3C XML developments have also been extremely open, including SAX.

XML documents themselves are also considerably more open than their binary counterparts. Anyone can parse a well-formed XML document, and validate it if a DTD is provided. While companies may still create XML that behaves in a specific way bound to their application, the data in the XML document is available to any application. While developers could create obfuscated DTDs or encrypt their data in a proprietary manner, they would lose most of the benefits of using XML. XML doesn't bar the creation of proprietary formats, but its openness is one of its greatest advantages. Application developers can partition tasks among multiple tools, possibly even from different vendors, allowing them all to operate on the same structured data set.

Experience

Successful application of XML will require data modeling expertise and (eventually) the building of a new set of tools. Fortunately, the skills of the SGML community are mostly transferable, providing XML with a large group of 'experts' early in its

existence. The XML specification was created for the most part by a group of experienced SGML developers, and has received vocal support from many sectors of the SGML community. Vendors are repurposing their tools, simplifying them for XML. Authors who had been writing SGML texts are focusing on XML as well, bringing markup structure to a wider audience. Companies that need to bring in outside vendors to help with large projects have a pool of firms with established track records to choose from.

14XML is a short form of eXtensible Markup Language". IT is very helpful file format. Here are 11 Advantages you can get by having xml file format:

> It is the most effective and economical way to publish your data on the web.

> It is very easy to handle and control.

> You can create large flat files in this format very easily.

> Xml is very useful web publishing format because it is cross platform supported.

> It is also support non printable characters.

> It helps you to reduce the storage space as this format required very less space to store information.

> This format is very flexible. This format has adaptability ability.

> There are some standards in the representation of information in this format. So, Xml helps you to improve the efficiency of data access.

> There is no programming required because it is schema driven language.

> You can validate the format quick because it has the validation facility.

> Xml is very easy to archive and retrieve.

XML file format is very much beneficial for the business. So, you as business well-wisher can convert html, sgml, pdf, word or any other file format to xml format and get the above mentioned benefits.

Q15. Explain the features of PHP. How we can run PHP program?

Ans.

HTTP authentication with PHP

- **Cookies**
- **Sessions**
- **Dealing with XForms**
- **Handling file uploads**
  - **POST method uploads**
  - **Error Messages Explained**
  - **Common Pitfalls**
  - **Uploading multiple files**
  - **PUT method support**
- **Using remote files**
- **Connection handling**
- **Persistent Database Connections**
- **Safe Mode**
  - **Security and Safe Mode**
  - **Functions restricted/disabled by safe mode**
- **Command line usage — Using PHP from the command line**
  - **Introduction**
  - **Differences to other SAPIs**
  - **Options — Command line options**
  - **Usage — Executing PHP files**
  - **I/O streams — Input/output streams**
  - **Interactive shell**
  - **Built-in web server**
  - **INI settings**
- **Garbage Collection**
  - **Reference Counting Basics**
  - **Collecting Cycles**
  - **Performance Considerations**
- **DTrace Dynamic Tracing**
  - **Introduction to PHP and DTrace**
  - **Using PHP and DTrace**
  - **Using SystemTap with PHP DTrace Static Probes**

To create a PHP program, simply create a new file, such as hello.php. In the file, place the following:

```
<?php
 echo "Hello, world!";
?>
```

The above program can be executed using the EditRocket Tools -> PHP -> Execute Program option, or you can execute it from a command prompt. To execute the script in the command prompt, use the cd command to cd to the directory where the hello.php file was saved, such as

```
cd C:\Scripts
```

Then type the following:

```
C:\php\php.exe hello.php
```

Hello World! should then be printed to the screen.

For more information about runing PHP programs via a local Apache Web Server on Windows, see the following:

Installin apache WAMP Server to run PHP:

1. Goto **www.wampserver.com** and download **WAMP Server**.
2. Just install it like other softwares by just clicking next next...
3. Now go to START menu of windows and start wampserver. Generally, the path is **Start -> WapmServer -> start WampServer**
4. Open your web browser and type **http://localhost** (or **http://127.0.0.1**) If you see a default WampServer home page, you installation is success.

5. Now put your php code in **www** folder in your wamp installation directory. Usually this is **C:\wamp\www**

6. Now type **http://localhost/filename.php** in your browser. (where *filename* is your php file name)

   It will execute your php code.. Thats it...

- PHP can generate dynamic page content
- PHP can create, open, read, write, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can restrict users to access some pages on your website
- PHP can encrypt data

With PHP you are not limited to output HTML. You can output images, PDF files, and even Flash movies. You can also output any text, such as XHTML and XML.

To start using PHP, you can:

- Find a web host with PHP and MySQL support
- Install a web server on your own PC, and then install PHP and MySQL
- f your server has activated support for PHP you do not need to do anything.
- Just create some .php files, place them in your web directory, and the server will automatically parse them for you.
- You do not need to compile anything or install any extra tools.
- Because PHP is free, most web hosts offer PHP support.

However, if your server does not support PHP, you must:

- install a web server
- install PHP
- install a database, such as MySQL

The official PHP website (PHP.net) has installation instructions for PHP: http://php.net/manual/en/install.php

ANS 16.

ML Document Example

```
<?xml version="1.0"?>
<note>
   <to>Tove</to>
   <from>Jani</from>
   <heading>Reminder</heading>
   <body>Don't forget me this weekend!</body>
</note>
```