

Internet Programming

Chapter 1 : Client Side Programming : HTML and CSS

Q. 1 Explain Web System Architectures – 1, 2, 3 and n.

Ans. :

Layers or Services

A software application is created using programming languages(called as frontend) and database(called as backend). In every software we have to implement following three layers:

(I) User Layer (presentation layer)

It is also called as client layer which contain User interface of our application. This layer is used for design purpose. In this, data is presented to the user and also input can be accepted from the user. For example in banking software, the registration form of an account holder can be considered as user layer.

(II) Business Layer

This layer is also known as business service. In this layer we can write all business logic like validation of data, calculations, data insertion etc. This acts as an interface between Client layer and Data Layer. This layer is also called the intermediary layer which helps to make communication faster between client and data layer.

(III) Data Layer

In this layer actual database is comes in the picture. Data Access Layer contains methods to connect with database and to perform insert, update, delete, get data from database based on our input data. Depending upon the implementation of these three layers there are types of database architectures.

A. One Tier Architecture

In One Tier Architecture all the three layers i.e. user layer, business layer and data layer are implemented in a single application. This architecture is usually used for small applications.

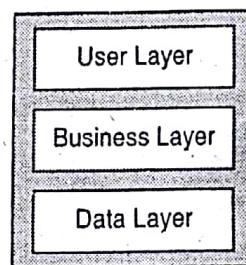


Fig. 1.1 : One Tier Architecture

B. Two Tier Architecture

The two tier architecture is based on the client server architecture. The direct communication takes place between client and server. The two tier architecture is the architecture in which user interface is run on client side and data layer is stored on the server side. In two tier architecture we can integrate business layer with either presentation layer or database layer or can be distributed in both.

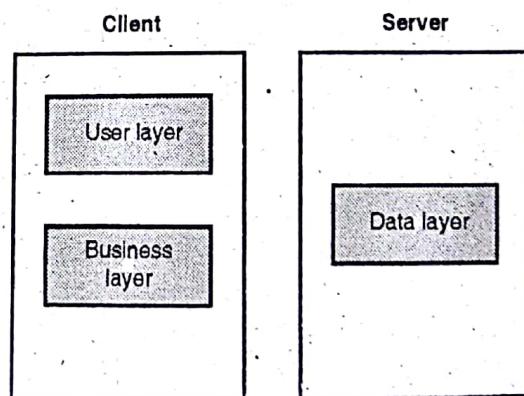


Fig. 1.2 : Fat client

- i) The business layer can be integrated with presentation layer at client side. In this case the size of client application increases, hence it is known as Fat Client.

- ii) The business layer can be integrated with data layer at server side. In this case the size of server application increases, hence it is known as **Fat Server**.
- iii) The business layer can be integrated with both user layer and data layer.

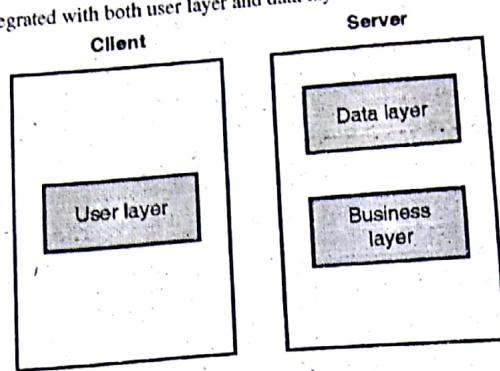


Fig. 1.3 : Fat server

Advantages of two tier architecture

1. In two tier architecture, applications can be easily developed due to simplicity.
2. In this client and server are directly connected, due to which communication becomes faster.
3. Maximum user satisfaction is achieved with accurate and fast prototyping of applications through robust tools.
4. It contains static business rules which are easily applicable for homogeneous environment.
5. We can distribute business layer physically with the user layer as well as data layer.

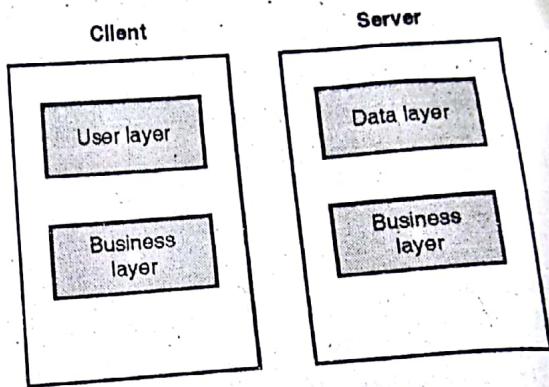


Fig. 1.4 : Two tier architecture

Disadvantages of two tier Architecture

1. It can only support to the limited number of users due to lack of scalability.
2. The performance of two tier architecture degrades when number of user increases.
3. Two tier architecture is cost ineffective.
4. As per security concern it is complicated.

C. Three Tier Architecture

The three tier architecture is most widely used architecture in today's world. In this architecture the user layer, business layer and data layer are implemented independently by three different applications.

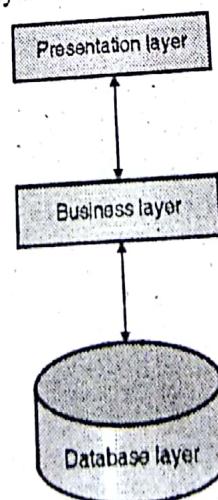


Fig. 1.5 : Tree tier architecture

The data required by the business logic exists in database server. In three tier architecture all layers interact with each other independently.

Advantages of three tier architecture

1. In three tier architecture we can manage the data independently.
2. We can make the changes in presentation layer without affecting other two tiers.
3. As each tier is independent it is possible to use different groups of developers.
4. It is most secure since the client doesn't have direct access to the database layer.
5. When one tier fails there is no data loss, because we are always secure by accessing the other tier.
6. Due to distributed deployment of application server, scalability is increased.
7. A similar logic can be used in various applications. It is reusable.
8. It is robust and secure due to multiple layers.

Disadvantages of three tier architecture

1. It is more complex structure.
2. More difficult to set up and maintain.
3. The physical separation of the tiers may affect the performance.

D. N-Tier Architecture

It is Client-Server architecture in which user layer, business layer and data layer functions are physically separated. This architecture model provides Software Developers to create Reusable application/systems with maximum flexibility. These three layers can be further subdivided into different sub-layers depending on the requirements.

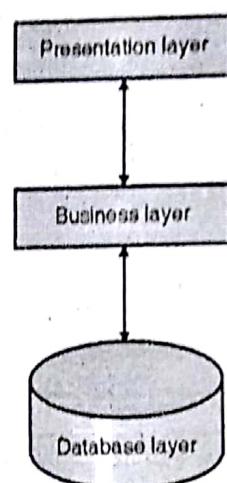


Fig. 1.6 : N-tier architecture

Q. 2 Explain 3-tier web architecture with diagram for online shopping database system?

Ans. :

3-tier web architecture of online shopping system

This is the online shopping diagram for 3 Tier architecture. Here the as a front end Dot Net environment is used while as backend database MS SQL Server 2008 is used.

Online shopping - 3 layer architecture

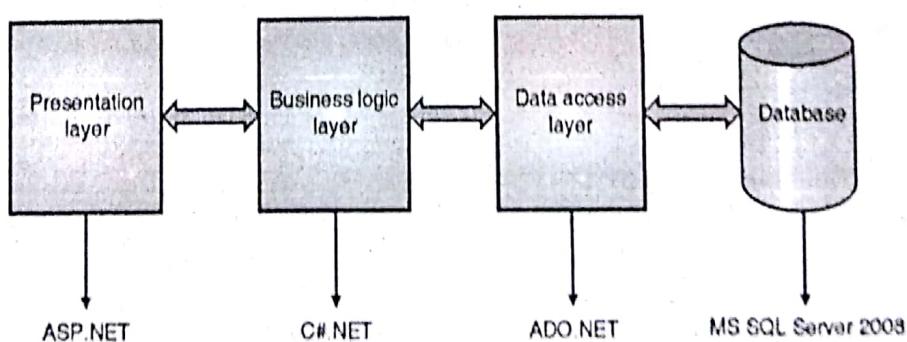


Fig. 1.7 : Online shopping diagram for 3-tier architecture

Fig. 1.7 shows three layers :

1. **Presentation layer :** This layer consist of user interface designed for the interaction with end user. This layer is created in ASP .Net. It includes the screens which will be used by the end user for shopping. Theses screen show the products with details as per their categories. User can select the product to purchase and add them into cart. This design is created with advanced controls available in ASP .Net.

2. **Business Layer :** This layer consists of validation checking code related to product selection of user. Accidentally user may select wrong number of products to purchase. For example, if any user is giving order to purchase 10000 TV sets, then the order should be validate. This logical code is implemented using C# .Net.
3. **Data Layer :** This layer contains the code interacting with database on the server. For example, accessing product details from database, inserting transaction details of user order in database etc. This database handling is implemented using the ADO. Net. Here all the three layers work independently and efficiently.

Q. 3 Explain Domain Name Service, Domain Name Registration and types of hosting.

Ans. :

DNS stands for Domain Name System (DNS)

The DNS is an identification scheme for computers, services, or some other sources that are connected to the Internet or a personal network. It links a variety of information with domain names linked to each and every contributing entity. Most importantly, it translates additional domain names to the number of different IP addresses which are needed for identifying and locating computer devices with network protocols. The DNS is a necessary part for the use of the Internet which has been in use since 1985.

Domain Name Registration

Domain name registration is the method of getting a domain name by a domain name registrar. Domain name registration information is reserved by the domain registries, which deal with domain registrars to supply registration facilities to the people. A client chooses a registrar to offer the registration service, and that particular registrar becomes the elected registrar for the domain selected by the client. Only the selected registrar can delete or modify the information regarding domain names in a central registry record. It is not strange for a client to change registrars. When a registrar registers a com domain for a client, he must pay an annual fee for it. Most of the domain registrars charge their products and services to address both the yearly fees and the management fees which should be paid to ICANN (Internet Corporation for Assigned Names and Numbers).

Domain Hosting Basics

A web hosting service is a type of Internet hosting service that allows individuals and organizations to make their website accessible via the World Wide Web. Web hosts are companies that provide space on a server owned or leased for use by clients, as well as providing Internet connectivity, typically in a data centre.

Types of hosting

1. Smaller hosting services

The most basic is web page and small-scale file hosting, where files can be uploaded via File Transfer Protocol (FTP) or a Web interface. The files are usually delivered to the Web "as is" or with minimal processing. Many Internet Service Providers (ISPs) offer this service free to subscribers. Individuals and organizations may also obtain Web page hosting from alternative service providers. Free web hosting service is offered by different companies with limited services, sometimes supported by advertisements, and often limited when compared to paid hosting. Single page hosting is generally sufficient for personal web pages. Personal web site hosting is typically free, advertisement-sponsored, or inexpensive. Business web site hosting often has a higher expense depending upon the size and type of the site.

2. Larger hosting services

Many large companies that are not Internet service providers need to be permanently connected to the web to send email, files, etc. to other sites. The company may use the computer as a website host to provide details of their goods and services and facilities for online orders. A complex site calls for a more comprehensive package that provides database support and application development platforms (e.g. ASP.NET, ColdFusion, Java EE, Perl/Plack, PHP or Ruby on Rails). These facilities allow customers to write or install scripts for applications like forums and content management. Also, Secure Sockets Layer (SSL) is typically used for websites that wish to keep the data transmitted more secure.

Q. 4 Differentiate between HTTP and FTP.

Ans. :

Parameter	HTTP	FTP
Functionality	Basic functionality is to access websites.	FTP transfers files from one host to another.

Parameter	HTTP	FTP
Connection	Only data connection is established.	FTP establishes two connections: one for data and one for the control connection.
TCP ports	HTTP uses TCP's port number 80.	FTP uses TCP's port number 20 and 21.
Efficiency	HTTP is efficient in the process of transferring smaller files like web pages.	FTP is efficient in the process of transferring larger files.
Authentication	No need of authentication.	FTP requires a password.
Data	The content transferred to a device using HTTP is not saved to the memory of that device.	The file transferred to the host device using FTP is saved in the memory of that host device.

Q. 5 What is cross browser compatibility? What are its issues and how to resolve them?

Ans. :

What is cross browser compatibility?

When a website looks and behaves same on all the browsers, it is known as cross browser compatibility. Ensuring cross browser compatibility is an important aspect for web designers while designing a website.

Here is a simple example

Consider a form on a website which is very important for lead generation. Before launching in the testing phase, the form worked perfectly on the browser. Let us consider on the browsers of 25% visitors, the form is not working properly : it either looks misaligned and behave improperly. This will affect the impression of website badly. And there may loss of 25% leads.

What are cross browser compatibility issues?

There are four main elements which leads to cross browser compatibility issues:

- A) Usually all the W3C standards are implemented by all the browsers. But they may not do it all the same way. For example HTML, CSS and JavaScript may be treated differently by each browser.
- B) In addition to basic implementation, all the browsers handle enhancements or deviations from W3C standards in their own way.
- C) Graphics and visual elements are handled by the each OS in its own way creating differences.
- D) The screen resolutions related to web and mobile are always different which affect the look and feel of the website.

How to ensure cross browser compatibility?

A web designer can ensure cross-browser compatibility with the help of number of things. Sometimes because of complexity, it becomes difficult to achieve 100% cross browser compatibility, but a web designer definitely takes efforts so that the site will look and behaves more or less consistently across browsers. For this purpose, the developer can take following steps :

Steps to resolve cross browser Compatibility issues

1. Understand Browser Differences

The initial aspect which should be concerned by the web designer is the differences between browsers and their way to handle elements. If web designer show this awareness, then it will be an important aspect in ensuring that the website design does not generate any cross browser compatibility issues which are difficult to correct.

2. HTML Editor Selection

The HTML editor used while creating the website often appends lines of code which may create some cross browser compatibility issues. Hence selection of a good editor is very important.

3. Style Sheets

The CSS (Cascading Style Sheets) helps to manage the look and feel of web-elements of the website. However, some CSS may not be compatible with some versions of browsers. Hence a web developer should consider this and choose styles which will have minimum impact on cross browser compatibility.

4. Flash

One more important thing which designers should keep in mind is that the browser designed for the blind and some of the mobile browsers do not have support for flash. While the visual appeal of the site is improved by the sites it can also leads to cross browser compatibility issues.

5. Image links

While using images the use of ALT attribute is very important otherwise they will generate the same issue of cross browser compatibility.

6. Clean Code

Another thing which must be ensured by the designer is that the code should be absolutely clean and do not contain any avoidable elements which may be interpreted differently by different browsers.

7. W3C Compliance

It is responsibility of the web designer to try and achieve W3C compliance. This is surely a difficult task but it will definitely helps in avoiding the cross browser compatibility issues.

8. Cross-Browser Testing

The last but not least is to test the website in maximum possible browsers. Nowadays there are many free and paid tools available for cross-browser testing.

Q. 6 Write note on W3C validators.

Ans. :

W3C validators

W3C stands for the World Wide Web Consortium. It is recognized standards organization for the world-wide web. Berners-Lee founded this organization and continue to develop and maintain web standards with a full time devoted staff. The standards of W3C guide the web designers and developers to design and develop website that lives up to certain standards. W3C has written a book that helps to check whether the code regarding web is well-written or poorly written. The **Markup Validation Service** is a **validator** provided by the World Wide Web Consortium(W3C) that allows Internet users to check HTML and XHTML documents for well-formed markup.

Markup validation is an important step towards ensuring the technical quality of web pages. However, it is not a complete measure of web standards conformance. Though W3C-validation is important for browser compatibility and site usability, it has not been confirmed what effect it has on search engine optimization. **Mark-up validators** are used to check missed closing tags and other technicalities.

DTD-based validators are also limited in their ability to check attribute values according to many specification documents. For example, using an HTML 4.01 DOCTYPE, `bgcolor="#fffff"` is accepted as valid for the "body" element even though the value "#fffff" is missing a preceding '#' character and contains only five (instead of six) hexadecimal digits. Also, for the "img" element, `width="really wide"` is also accepted as valid. DTD-based validators are technically not able to test for these types of attribute value problems. Pages may not display as intended in all browsers, even in the absence of validation errors and successful display in other browsers. The only way to ensure that pages always display as intended is to test them in all browsers expected to render them correctly. While the W3C and other HTML and XHTML validators will assess pages coded in those formats, a separate validator like the W3C CSS validator can check that there are no errors in the associated Cascading Style Sheet. CSS validators apply current CSS standards to referenced CSS documents.

Q. 7 Explain the basic structure of HTML document.

Ans. :

Structure of HTML document

All the elements are included in the main opening and closing `<html>` tags.

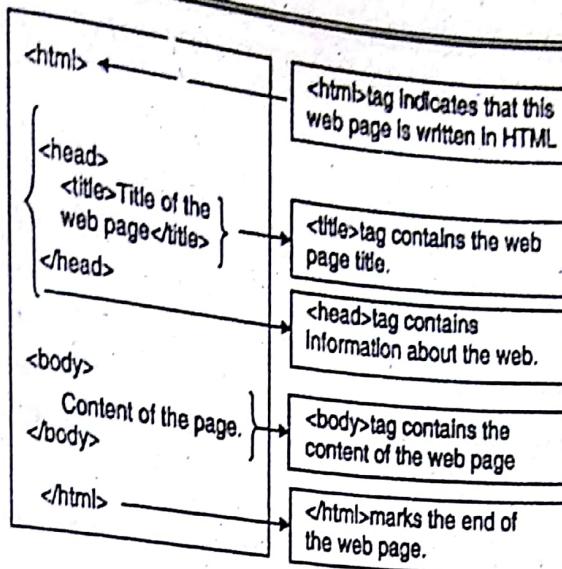


Fig. 1.8 : Structure of HTML Documents

An HTML document contains two main parts:

1. Head

The head element contains title and metadata of a web document. This section is used to declare variables and functions in scripting languages. These variables and functions are then accessible throughout the page.

2. Body

The body element contains the information which we like to display on a web page. All text to be displayed and control creation is done in this section.

Q. 8 Explain Image Map with example (program).

May 15, Dec 15.

Ans. :

Image map with example

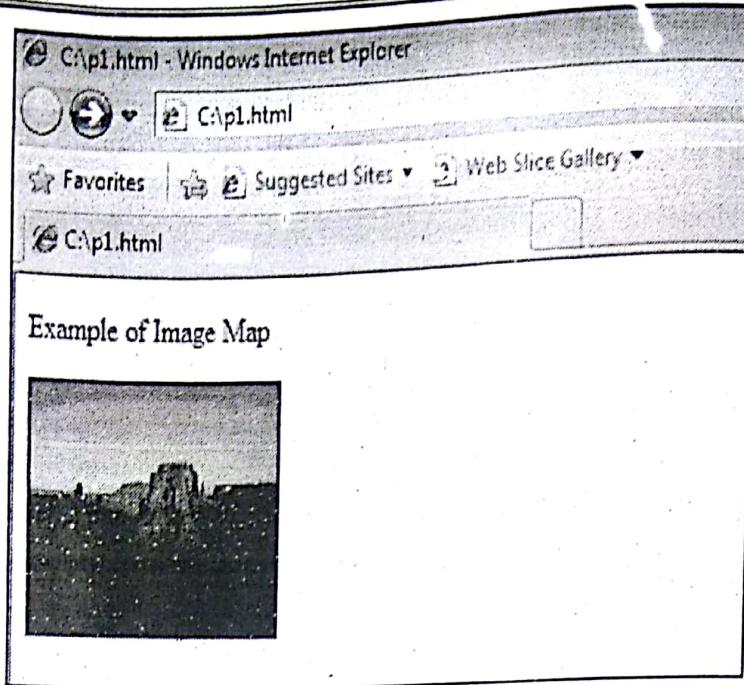
HTML provides the **<map>** tag to define a client-side image-map. It is an image with clickable areas. The **<map>** element has name attribute which is associated with the attribute use map and links the image with the map. The **<map>** element has sub-tag **<area>**, which defines the clickable areas in the image map.

Program to demonstrate use of **<map>** tag

```

<!DOCTYPE html>
<html>
<body>
<p>Example of Image Map</p>

<map name="smap">
<area shape="rect" coords="0,0,82,126" alt="Jellyfish" href="Jellyfish.htm">
<area shape="circle" coords="90,58,3" alt="Koala" href="Koala.htm">
<area shape="circle" coords="124,58,8" alt="Penguins" href="Penguins.htm">
</map>
</body>
</html>
  
```



Q. 9 Explain the Lists and with types in HTML.

Ans. :

Use of list in HTML : As the name suggests, these elements are used to give list of items. Lists are used everywhere on the websites. There are number of things like articles, website navigation menus, and product features on e-commerce websites which makes frequent use of lists.

Types of HTML list

HTML provides three ways to give the lists :

- A. – An unordered list. Plain bullets are used for the list items.
- B. – An ordered list. Different schemes of numbers are used for the list items.
- C. <dl> – A definition list. This is used to give list of definitions just like dictionaries.

A. HTML Unordered Lists

Use : In this list, plain bullets are used for the list items. Generally this option is used when there is no any standard sequence or order of the list items. The tag is used to give this list. Every element in the list is marked by a bullet. Different types of bullets are available. The which is the sub-tag of is used for individual list elements. The tag has attribute *TYPE* which is used to specify the type of bullet for the list items.

There are following options for the attribute *TYPE* of tag.

- (i) Square
- (ii) Circle
- (iii) Disc

B. HTML Ordered Lists

Use : Ordered lists are generally used when we want to specify numbers instead of bullets. The tag is used for such list. The number starts from 1 and also incremented by one for each successive list element in the ordered list. The tag has attribute *TYPE* which is used to specify the type of number for the list items. The default type is 1.

There are following options for the attribute *TYPE* of tag.

- 1 - Number
- I – Upper Roman
- i – Lower Roman
- A – Upper Alpha
- a – Lower Alpha

The start Attribute

This attribute of tag is used to specify the starting point of numbering.

C. HTML Definition Lists

Use : Definition Lists are used to give the list of definitions just like the dictionary or encyclopedia. In general this is considered as the ideal way to represent a glossary, list of terms, or some kind of name/value pair list.

Following tags are used in definition list.

- (i) <dl> - Defines the beginning of the list
- (ii) <dt> - Specify a term
- (iii) <dd> - Specify term definition
- (iv) </dl> - Defines the end of the list

Chapter 2 : JavaScript

Q.1 Explain characteristics of JavaScript.

Ans. :

Characteristics of JavaScript

1. JavaScript is a lightweight, interpreted client side scripting language.
2. Designed for developing network-based applications.
3. JavaScript is complementary to Java.
4. JavaScript is complementary to and integrated with HTML.
5. It is Open source and cross-platform.
6. The user input is validated before sending the page to the server. This minimizes the server traffic, which tends to fewer loads on the server.
7. There is no need for the user to wait to see if something have been forgotten to enter.
8. Interactive interfaces can be created which can give responses to end user actions like mouse or keyboard activities.
9. JavaScript can include elements like drag-drop components and sliders to provide a feel of rich interface to the users.

Q.2 What are the advantages and disadvantages of client side scripting ?

Ans. :

Advantages of client side scripting

- (1) Immediate response to user's actions which enables more interactivity
- (2) No need to go to server hence execution is fast.
- (3) Improve the usability of Web sites for users whose browsers support scripts.
- (4) Developers get more control over the look and behavior of their Web widgets.
- (5) Possible to substitute by HTML if users' browsers do not support scripts.
- (6) Are reusable and obtainable from various types of free resources.

Disadvantages of client side scripting

- (1) Scripts are not supported by all of the browsers, hence there may occur errors if no alternatives have been provided.
- (2) There is need of more quality assurance testing as different browsers and browser versions support scripts differently.
- (3) May need more time and effort to development if the scripts are not already available through other resources.
- (4) Sometimes the web widget looks like a standard control but their behavior may be different or vice-versa which may lead to usability problems.

Q. 3 What are differences between client side and server side scripting language?

Ans. :

Parameter	Client Side Scripting Language	Server Side Scripting Language
Execution	The client side script is executed by Web Browser which is located at the user's computer.	The server side script is executed by the Web Server that outputs the page which is to be sent to the browser.
Database	Client side scripting language cannot connect to the databases which is located on the web server.	Server side scripting language can connect and access to the databases which is located on the web server.
File System	Client side scripting language cannot have access to the file system which is located on the web server.	Server side scripting language has access to the file system which is located on the web server.
Access to Setting	Client side scripting language can access the files and settings that are local at the user's computer.	Server side scripting language cannot access the settings that belong to Web server.
Block	User can block the Client side scripting language	User cannot block the Server side scripting language
Response	Response from a client-side script is quick since the scripts are processed on the local computer.	Response from a server-side script is slow since the scripts are processed on the remote computer.
Examples	JavaScript, VBScript, etc.	PHP, JSP, ASP, ASP.Net, Ruby, Perl, etc.

Q. 4 Explain JavaScript editing tools.

Ans. :

JavaScript

One of main advantage of JavaScript is that there is no need of development tools which are expensive. A simple text editor like Notepad can be used to write the scripts. Inside the context of a web browser, JavaScript is an interpreted language; hence there is no need to buy a compiler.

JavaScript Editing Tools

1. Microsoft FrontPage

This is popular product of Microsoft. It provides various JavaScript tools to web developers for assistance to create interactive websites.

2. Macromedia Dreamweaver MX

It is very popular HTML and JavaScript tool used for web development professionally. It provides various JavaScript components to handle databases, and supports new standards like XHTML and XML.

3. Macromedia HomeSite 5

This is a product of Macromedia which is well-liked HTML and JavaScript editor. It helps to effectively manage personal websites.

Q. 5 Write the difference between Java and JavaScript.

Ans. :

Parameter	Java	JavaScript
Execution	Java creates application which can be executed on virtual machine or browser.	JavaScript creates application which can be executed on browser only.
Features	Java code allows programmer full functionality.	JavaScript code contains limited number of commands and features.
Naming	The first name of Java was OAK and was developed by James Gosling, Sun MicroSystems.	JavaScript was earlier known as LiveScript and was developed by Brendan Eich, Netscape.
Type Safety	Java is high-level, compiled and strongly typed language.	JavaScript is text based and weakly typed language.
Variables	Variables are created using the data type names like int, char double etc.	Variables are created using the var keyword.

Parameter	Java	JavaScript
Extension	Java program has file extension ".Java" and after compilation it creates ".class" file.	JavaScript file has file extension ".js" or ".html".
Objects	Objects of Java are class based.	Objects of JavaScript are prototype based.
Scope	Java has block based scope.	JavaScript has function based scope and object based context.

Q.6 Explain various languages constructs of JavaScript.

Ans. :

Data Types and Variables

Variable is a name given to memory location where we can store some value. The value depends upon the **data type** of variable. In JavaScript there are number of **data types** used to store different types of values. These data types are primarily categorized as :

1. JavaScript primitive data types

In JavaScript, there are five types of primitive data types as follows :

Sr. No.	Data Type	Description
1.	String	Represents sequence of characters e.g. "Ishita"
2.	Number	Represents numeric values e.g. 101
3.	Boolean	Represents Boolean value either true or false
4.	Undefined	Represents undefined value
5.	Null	Represents null means no value at all

2. JavaScript non-primitive data types

Sr. No.	Data Type	Description
1.	Object	Represents instance which helps to access members
2.	Array	Represents set of same values
3.	RegExp	Represents regular expression

JavaScript is considered as a **dynamic type language** that is there is no need to specify type of the variable. This type is dynamically decided by the JavaScript engine. While declaring a variable, "var" keyword is used on place of data type. Var means variant, that is the variable can store any type of value like numbers, strings, dates etc.

Examples

1. var rno = 101; //holding number

2. var sname="Kunal";//holding string

Q.7 What are JavaScript objects? List the important built-in objects. How can you write your own object?

Ans. :

JavaScript object

An object is nothing but an entity having its own state and behavior (properties and methods).

For example : A flower is an object having properties like color, fragrance etc. Other examples of objects are car, pen, bike, chair, glass, keyboard, monitor etc. JavaScript is an object-oriented language. Everything in JavaScript is considered as an object.

Examples of objects

Following are some of the examples of objects in JavaScript.

- (i) Booleans (when defined with the new keyword)
- (ii) Numbers (when defined with the new keyword)
- (iii) Strings (when defined with the new keyword)
- (iv) Dates
- (v) Regular expressions
- (vi) Arrays
- (vii) Functions

We can create our own user defined objects in JavaScript. JavaScript is basically a template based scripting language not class based. Hence, we directly create the object without class.

Creating Objects in JavaScript

1. JavaScript Object by Object Literal

Syntax

Following is the syntax of creating object using object literal :

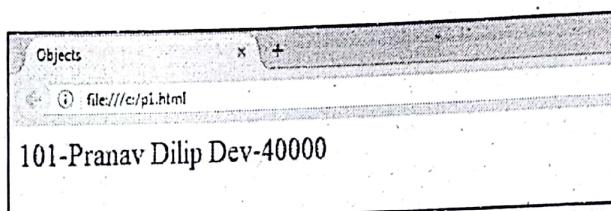
```
object = {property1:value1,property2:value2,...,propertyN:valueN}
```

As we can observe, the property and value is separated by the separator : (colon).

Program of creating object using object literal

```
<html>
<head>
<title>Objects </title>
</head>
<body>
<font size=5>
<script language="JavaScript">
employee = {id:101,name:"Pranav Dilip Dev",salary:40000}
document.write(employee.id + "-" + employee.name + "-" + employee.salary);
</script>
</font>
</body>
</html>
```

Output



2. By creating instance of Object

Syntax

Following is the syntax of creating instance of object :

```
var objectname=new Object();
```

Here, new keyword is used to create object.

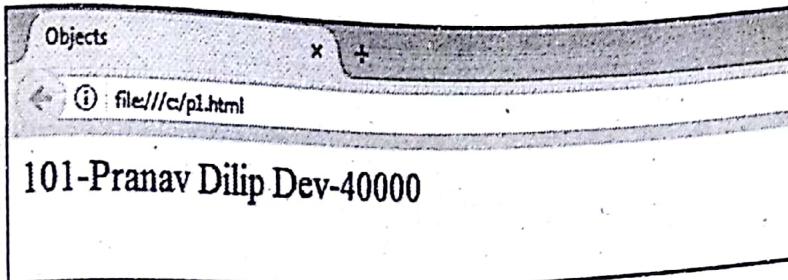
Program of creating the instance of object

```
<html>
<head>
<title>Objects </title>
</head>
<body>
<font size=5>
<script language="JavaScript">
var employee=new Object();
employee.id=101;
employee.name="Pranav Dilip Dev";
employee.salary=40000;
```

(es easy solutions)

```
Internet Programming  
document.write(employee.id + "-" + employee.name + "-" + employee.salary);  
</script>  
</font>  
</body>  
</html>
```

Output



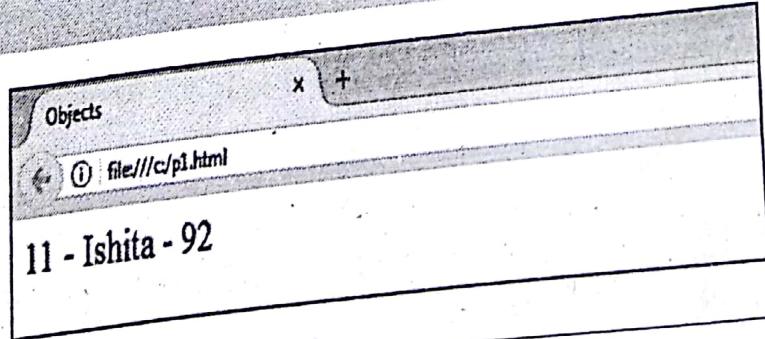
3. By using an Object constructor

Here, we have to create parameterized function. "this" keyword is used to assign each argument value in the current object. The "this" keyword refers to the current object.

Program using "this" keyword

```
<html>  
<head>  
<title> Objects </title>  
</head>  
<body>  
<font size=5>  
<script language="JavaScript">  
function student(id,sname,marks)  
{  
    this.id=id;  
    this.sname=sname;  
    this.marks=marks;  
}  
s = new student(11,"Ishita",92);  
document.write(s.id + "-" + s.sname + "-" + s.marks);  
</script>  
</font>  
</body>  
</html>
```

Output



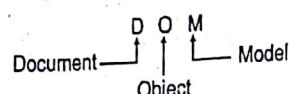
Q. 8 Define DOM. Explain in detail node tree for HTML document. Also explain different levels of DOM.

May 15, May 16, Dec. 16

Ans. :

DOM

DOM stands for Document Object Model. The entire html document is represented by the **document object**. The html document becomes document object when it is loaded in the browser. The root element represents the html document. The document object has properties and methods. The document object helps to add content dynamically in the web page. Any element of HTML page can be accessed by using the document object. According to W3C(World Wide Web Consortium) "The W3C Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document."



Hierarchy of objects in web document

Document Object Model (DOM) is the method by which the content of document is accessed and modified. In a web document, the organization of objects is implemented in a hierarchical structure.

1. Window object

It resides at top of the hierarchy. It is the topmost element of the object hierarchy.

2. Document object

All the HTML documents which get loaded into browser are considered as document objects. The contents of the page are stored in the document object.

3. Form object

Everything which is contained in the opening <form> and closing</form> tag sets the form object.

4. Form control elements

The form object has all the elements which are defined for the objects like text fields, buttons, checkboxes, select box, radio buttons etc.

DOM Levels

The DOM provides all the features to JavaScript to create dynamic HTML :

- (1) Changes can be made in all HTML elements.
- (2) Changes can be made in attributes of HTML elements.
- (3) Changes can be made in all CSS styles in the page.
- (4) Existing HTML elements and attributes can be deleted.
- (5) New HTML elements and attributes can be added.
- (6) Response can be given to HTML events.
- (7) New HTML events can be created in HTML page.

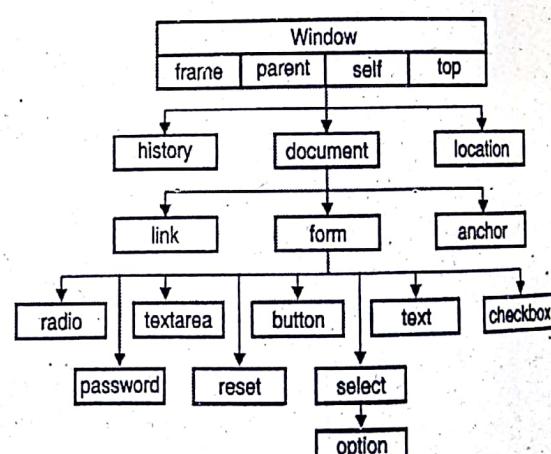


Fig. 2.1 : DOM levels

Q. 9 Explain the event handling in JavaScript with simple example.

Ans. :

Event

Events are the actions performed by the end users while browsing the website. For example mouse move or mouse click on the buttons. When an event is fired, objects are triggered which are associated with that specific event. The event is caught by the Event Handlers and in response the related code is executed.

Events are basically classified in four categories :

1. Window Events

There are various types of events regarding window :

- (i) **onLoad** - triggered when a new page is starting up
- (ii) **onUnload** - triggers when a page is shutting down
- (iii) **onResize** - triggers when a page is resized
- (iv) **onMove** - triggers when a page is moved
- (v) **onAbort** - triggers when a page is cancelled
- (vi) **onError** - triggers when an error occurs
- (vii) **onFocus** - triggers when the window moves to foreground.
- (viii) **onBlur** - triggers when window changes to background

2. Mouse Events

There are various types of events associated with mouse :

- (i) **onmousedown** - triggers when mouse button is pressed on an element
- (ii) **onmouseup** - triggers when mouse button is released
- (iii) **onmousemove** - triggers when mouse pointer is moved and the pointer is already over an element
- (iv) **onmouseout** - triggers when mouse pointer is moved out of an element
- (v) **onmouseover** - triggers when the pointer is over an element
- (vi) **onClick** - triggers when mouse button is clicked once
- (vii) **ondblclick** - triggers when mouse button is clicked twice

3. Keyboard Events

There are various types of events associated with keyboard :

- (i) **onkeydown** - Triggers when a key is pressed down
- (ii) **onkeyup** - Triggers when a key is released
- (iii) **onkeypress** - Triggers when complete key sequence, down press and up release happens.

4. Form events

There are following types of events associated with form :

- (i) **onReset** - triggers when the reset button on the form is clicked
- (ii) **onSubmit** - triggers when the submit button is clicked
- (iii) **onSelect** - triggers when a content is selected on a page

Q. 10 Explain cookie with creating, reading and deleting operations.

Ans. :

Cookie

Cookies are small items of data, each consisting of a name and a value, stored on behalf of a website by visitors' web browsers. In JavaScript, cookies can be accessed through the `document.cookie` object, but the interface provided by this object is very primitive. Cookies.js is a JavaScript object that allows cookies to be created, retrieved, and deleted through a simple and intuitive interface. A cookie can be used for authenticating, session tracking, remember specific information about the user like his name, password, last visited date etc. Cookies are a plain text data record of 5 variable-length fields.

name-value

Each cookie has a *name-value pair* that contains the name and actual information of the cookie. The name of cookie can be used to retrieve its value later on.

Expiry date : Each cookie has an *expiry date* after which it is trashed. If expiry date is not specified, then the cookie is trashed when browser is closed.

Domain and path : Each cookie also has a *domain* and a *path*. The *domain* tells the browser to which domain the cookie should be sent. The *path* helps to specify a directory where the cookie is active. So if you want the cookie to be only sent to pages in the directory `cgi-bin`, set the path to `/cgi-bin`. Usually the path is set to `/`, which means the cookie is valid throughout the entire domain.

1. Creating Cookie

To create a cookie a string value to the document.cookie object can be assigned :

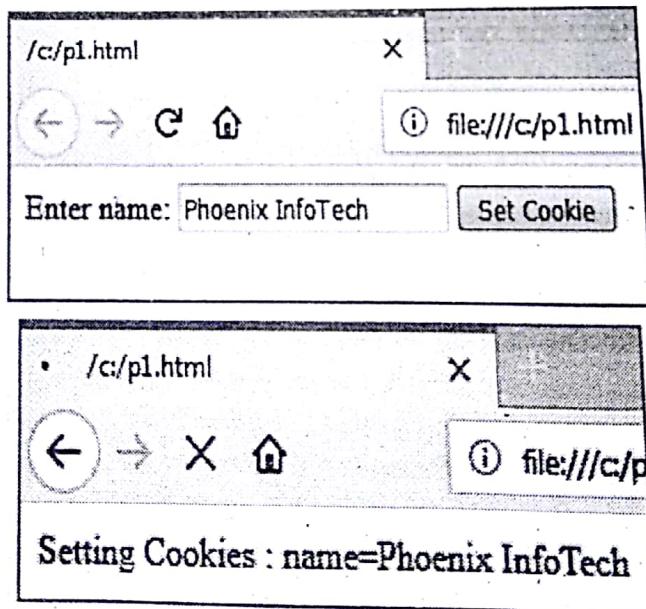
```
document.cookie = "key1=value1;key2=value2;expires=date";
```

Here the **expires** attribute is optional. When it is provided with a valid date or time, then the cookie will expire on a given date and later on the cookie value cannot be accessible.

Example

```
<html>
<head>
<script type = "text/javascript">
    function SetCookie()
    {
        v = document.frm.nm.value ;
        if( v == "" ){
            alert("Please enter name");
            return;
        }
        document.cookie="name=" + v;
        document.write ("Setting Cookies : " + "name=" + v );
    }
</script>
</head>
<body>
<form name="frm" action="">
    Enter name: <input type="text" name="nm"/>
    <input type="button" value="Set Cookie" onclick ="SetCookie();"/>
</form>
</body>
</html>
```

Output



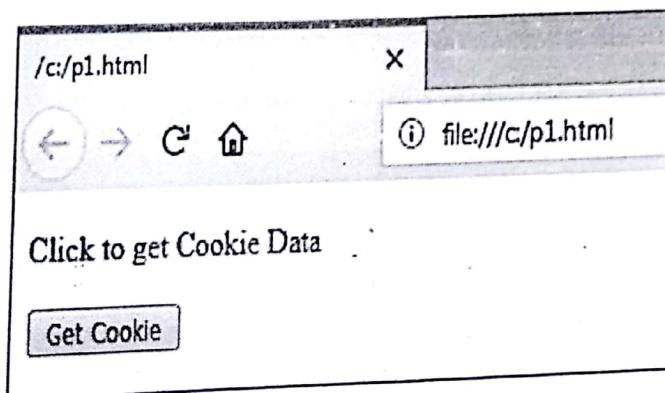
2. Reading Cookies

To read cookie we have to read value of the document.cookie object. The document.cookie string maintains a list of name = value pairs which are basically separated by semicolons, in which name is the name of a cookie while value is its string value. The string can be broken into key and value using split() function.

Example

```
<html>
<head>
<script type="text/javascript">
function ReadData()
{
    var cookies = document.cookie;
    document.write ("Data of all Cookies : " + cookies);
    arr = allcookies.split(' ');
    for(var i=0; i<arr.length; i++)
    {
        name = arr[i].split('=')[0];
        value = arr[i].split('=')[1];
        document.write ("Key is : " + name + " and Value is : " + value);
    }
}
</script>
</head>
<body>
<form name="frm" action="">
<p> Click to get Cookie Data </p>
<input type="button" value="Get Cookie" onclick="ReadData()"/>
</form>

</body>
</html>
```

Output**3. Deleting a Cookie**

Sometimes there is need to delete a cookie explicitly. A cookie can be deleted by setting the previous expiry date.

Example

```
<html>
<head>
<script type="text/javascript">
function Deltecookie()
{
    var dt= new Date();
    dt.setMonth( dt.getMonth() - 1 );
    v = document frm nm.value;
```

Implementing the Geolocation API.

Using the Geolocation API is quite straight forward. We first create a Geolocation object and then use the `getCurrentPosition` method to fetch user coordinates.

Here is the sample Code :

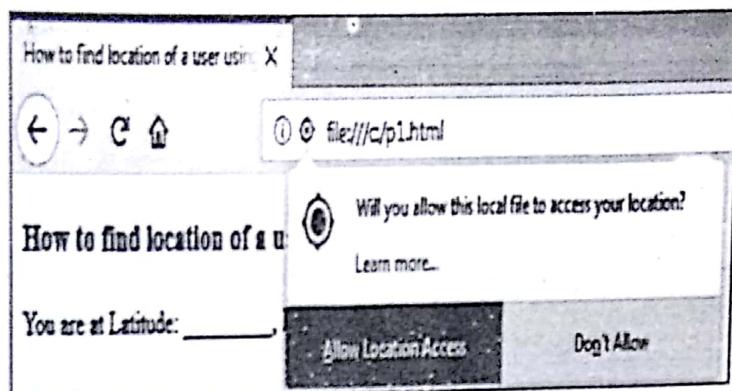
```
<!doctype html>
<html lang="en">
<head>
<title> How to find location of a user using HTML5 Geolocation? </title>
<meta charset="utf-8" > <link href="css/style.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" >
window.onload = getLocation;
var geo = navigator.geolocation;
function getLocation() {
    if( geo ) {
        geo.getCurrentPosition( displayLocation );
    }
    else {alert( "Oops, Geolocation API is not supported" );
    }
}
function displayLocation( position ) {
    var latitude = position.coords.latitude;
    var longitude = position.coords.longitude;
    var div = document.getElementById( 'location' );
    div.innerHTML = "You are at Latitude: " + latitude + ", Longitude: " + longitude;
}
</script>
</head>
<body>
<div class="container" >
<h3> How to find location of a user using HTML5 Geolocation? </h3>
<div id="location" >
You are at Latitude: _____, Longitude: _____
</div>
</div>
</body>
</html>
```

Here we will check if the browser supports the Geolocation API; if exists, then we will display the location

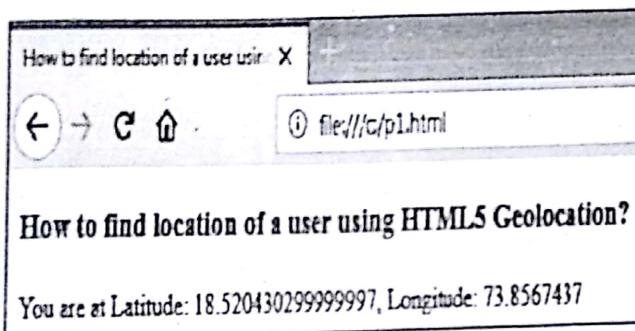
This function displays the latitude and longitude when the browser has a location.

Output

The first screen will be :



If user selects "Allow Location Access" :

**Explanation**

We first create a Geolocation Object geo; We then check if the browser supports Geolocation API . If the API is not supported then we simply return an error. If API is supported then we call the getCurrentPosition method and pass displayLocation() function as a callback. The displayLocation() function takes position object as parameter and print out the user coordinates on the Web Page. Remember that with Geolocation API ; sharing the location is always opt-in. In other words the geolocation API cannot access location data without explicit permission of the user. So, when you run the HTML page for the first time, the browser will ask if you want to share your location with the webpage. Assuming that you click on Yes, then only it can access the information. The webpage simply displays the user coordinates. Now lets display the coordinates on Google Maps.

May 17

Q. 4 Explain detail responsive web design with example.**Ans. :****Responsive web design**

Responsive web design is considered as a practice of creating a website which is suitable to work on each and every device and screen size, regardless of how large or small, mobile or desktop. Responsive web design is basically concentrated on offering spontaneous and satisfying experience for every user. Desktop computers as well as cell phone users get similar benefits from responsive websites. Only HTML and CSS are used in responsive web development. It is possible to view web pages on different devices such as desktops, tablets, and smart-phones. The web page should look good and be user friendly irrespective of the device. Web pages should not skip any information so as to fit smaller devices; instead adapt the content to fit any device.

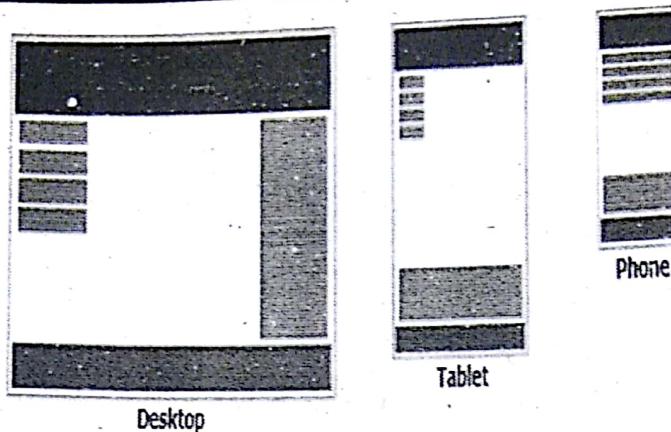


Fig. 3.1 : Responsive Website

Q. 5 Define media query. Explain media query with an example.

Dec. 15, May 16, Dec. 16, May 17, Dec. 17

Ans. :

Viewport

The visible area of a web page for any user is considered as viewport. The viewport is different for different devices. It is smaller on a mobile phone as compared to computer screen. Before the evolution of tablets and mobile phones, the web sites were basically designed only for computer screens, and the viewport is considered as common for all. Then, when mobiles and tablets are being used to surf the web, the web pages are not fit in those devices. To fix this, browsers on those devices scaled down the entire web page to fit the screen. This was not considered as perfect!! But a quick fix.

Setting the Viewport

HTML5 provide a way through `<meta>` tag to allow web designers to take control over the viewport. It is necessary to include following `<meta>` viewport element in all the web pages:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

A `<meta>` viewport element is used to instruct the browser on how to control the page's dimensions and scaling. The `width=device-width` part is used to set the width of the page to follow the screen-width of the device (which is different based on the device). The `initial-scale=1.0` part sets the initial zoom level at the first loading of page by the browser.

Media query

Media query is advanced technique introduced in CSS3. It is used to include block of CSS properties using `@media` rule based on certain condition.

Example

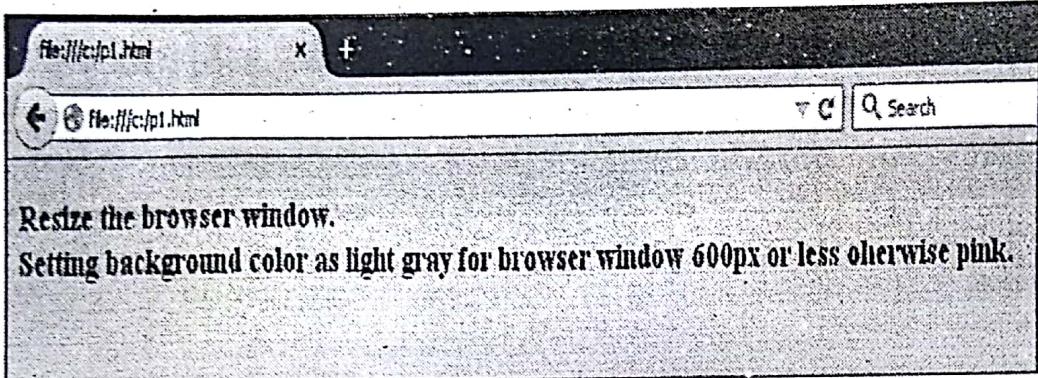
Setting background color as light gray for browser window 600px or less.

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<style>
body {
    background-color: pink;
}
@media only screen and (max-width: 600px) {
    body {
        background-color: lightgray;
    }
}
```

```

</style>
</head>
<body>
<h3>
Resize the browser window.
<br>Setting background color as light gray for browser window 600px or less
otherwise pink.</h3>
</body>
</html>

```

Output

Q. 6 Explain different types of CSS3 selectors with an example.

May 16

Ans. :

CSS3 attribute selectors

We have used CSS attribute selectors to target rules. For example, consider the following rule:

```

img[alt] {
border: 3px dashed #e15f5f;
}

```

This code targets all the image tags in the markup having an alt attribute:

```

```

It can be also made simple by specifying attribute value.

For example, consider the following rule:

```

img[alt="atwi_oscar"] {
border: 3px dashed #e15f5f;
}

```

This would only target images which have an alt attribute of atwi_oscar. It can be done in CSS2. Then what CSS3 provides?

There are three new "substring matching" attribute selectors in CSS3.

CSS3 substring matching attribute selectors

CSS3 offer facility to select elements depending on the substring of their attribute selector.

Here we can select an element, depending upon contents of the attribute.

The three options are whether the attribute is:

1. Beginning with the prefix
2. Contains an instance of
3. Ends with the suffix

1. The "beginning with" substring matching attribute selector

It has the following syntax:

```
Element[attribute^="value"]
```

When we want to select all images on the site having an alt attribute which starts with film, code will be :

```
img[alt^="film"] {  
border: 3px dashed #e1555f;  
}
```

The symbol ^ indicates "begins with".

2. The "contains an instance of" substring matching attribute selector

It has the following syntax:

```
Element[attribute*="value"]
```

When we want to select all images on the site having an alt attribute which contains film, code will be :

```
img[alt*="film"] {  
border: 3px dashed #e1555f;  
}
```

The symbol * indicates "contains".

3. The "ends with" substring matching attribute selector

It has the following syntax:

```
Element[attribute$="value"]
```

When we want to select all images on the site having an alt attribute which ends with film, code will be :

```
img[alt$="film"] {  
border: 3px dashed #e1555f;  
}
```

The symbol \$ indicates "ends with".

4. The :last-child selector

CSS2.1 provided a selector which is applicable for the first item in a list:

```
:first-child
```

Now, CSS3 provides a selector which can also match the last item in the list:

```
:last-child
```

When these selectors are used together, there is no need of any additional classes in the markup.

O.7 Discuss structural Pseudo Classes in CSS3 with example.

May 15, May 1

Ans.:

Pseudo classes In CSS3

CSS pseudo-classes are basically used to apply additional special effects to specific selectors without the help of scripting language.

Syntax of pseudo-classes is as follows :

```
selector:pseudo-class {property: value}
```

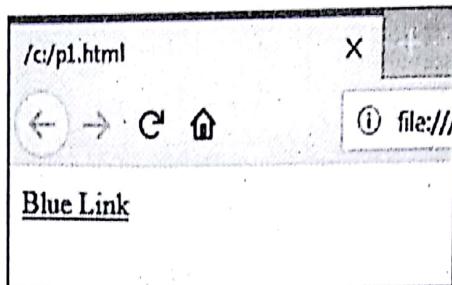
CSS pseudo-classes target elements which are difficult to target with basic selectors like id or class. We can use pseudo-class select elements depending on their attributes, states, and relative position. Here we will first see basic link used pseudo class :

The : link pseudo-class

This class is used to add special style effect to an unvisited link.

In the following example we have used the :link class to set the link color. Any valid color name will be accepted.

```
<html>
<head>
<style type="text/css">
a:link {color:#ff0000}
</style>
</head>
<body>
<a href="#">Blue Link</a>
</body>
</html>
```

Output

Q. 8 Explain in details CSS3 Transitions with example.

Dec. 15

Ans. :

CSS3 Transitions

CSS transitions help to modify property values from one value to another over a given duration of time. To apply a transition effect we have to specify two things : (i) The CSS property to which we want to add effect. (ii) Time duration for the effect. (iii) The default value duration is 0, hence if it is not mentioned then there will no effect of transition.

Example

The following example illustrates 50px * 50px gray <div> element. Transition effect is specified for the width property of The <div> element. The transition duration will be 3 seconds :

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
    width: 50px;
    height: 50px;
    background: gray;
    -webkit-transition: width 3s, height 6s; /* For Safari 3.1 to 6.0 */
    transition: width 3s, height 6s;
}

div:hover {
    width: 200px;
    height: 200px;
}
</style>
</head>
<body>
```

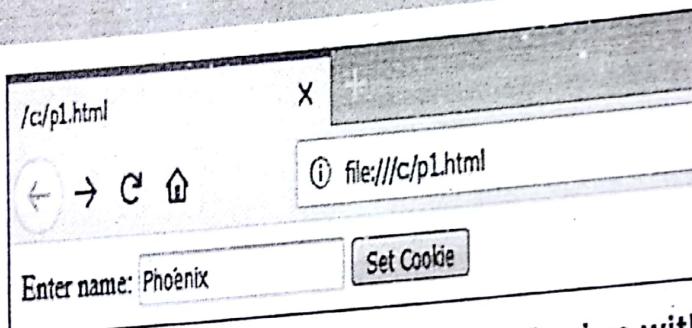
Internet Programming (MU)

```

        document.cookie = "name=" + v;
        document.cookie = "expires=" + dt.toUTCString() + ",";
        document.write("Setting Cookies : " + "name=" + v);
    }
</script>
</head>
<body>
<form name="frm" action="">
    Enter name: <input type="text" name="nm"/>
    <input type="button" value="Set Cookie" onclick="DeleteCookie()"/>
</form>
</body>
</html>

```

Output



Chapter 3 : HTML5 and Responsive Web Design with CSS3

Q. 1 Differentiate between HTML and HTML5.

Ans. :

Parameter	HTML	HTML5
Audio and Video	Audio and Video tags are not provided in HTML4	Audio and Videos are provided in HTML5 e.g, <audio> and <video>.
Vector Graphics	Vector Graphics is only possible by use of technologies like VML, Silverlight, Flash etc.	Vector graphics is by default supported by HTML5 e.g. SVG and canvas
Tracing User Location	It is difficult to trace true GeoLocation of end user browsing the website particularly when it comes to mobile devices.	JS GeoLocation API in HTML5 is used to identify location of user browsing the website (with permission of user)
Cookies	HTML uses cookies.	It supports local storage instead of cookies.
Shapes	The different shapes such as circle, rectangle and triangle are not possible.	It is easy to draw different shapes such as circle, rectangle, and triangle.
Browser Support	Supported by old browsers	Supported by all new browsers.
Syntax	Doctype declaration in HTML is very long <!DOCTYPE HTML PUBLIC "-//PhoenixGlobe//DTD HTML 4.01//EN" "http://www.PhoenixGlobe.com/TR/html4/strict.dtd">	DOCTYPE declaration in HTML5 is very simple <!DOCTYPE html>
Character Encoding	Character encoding in HTML is very long <!DOCTYPE HTML PUBLIC "-// PhoenixGlobe//DTD HTML 4.0 Transitional//EN">	Character encoding is very simple <meta charset="UTF-8">

Q. 2 List metadata elements used in HTML5.

Dec. 15

Ans. :

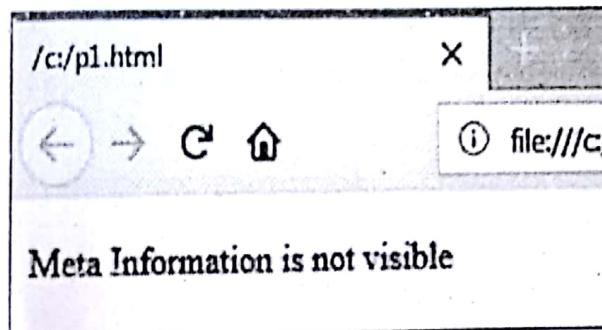
Metadata in HTML5

Metadata is nothing but the data (information) about data. HTML5% provides <meta> tag to manage metadata of the HTML document. The metadata is not visible on the page, but will be machine parseable. The meta elements are usually used to describe page description, keywords, author of the document, last modified, and other metadata. Browsers can use the metadata(how to display content or reload page), also search engines (keywords), and other web services also uses metadata. HTML5 provided a way to allow web designers take control over the viewport (the user's visible area of a web page), with the help of the <meta> tag.

Example

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<meta name="prod_desc" content="Electronics">
<meta name="prod_keywords" content="TV, Mobile, AC">
<meta name="Company" content="D. SVC">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<body>
<p>Meta Information is not visible</p>
</body>
</html>
```

Output



Q. 3 Explain Geo Location with an example.

May 15

Ans. :

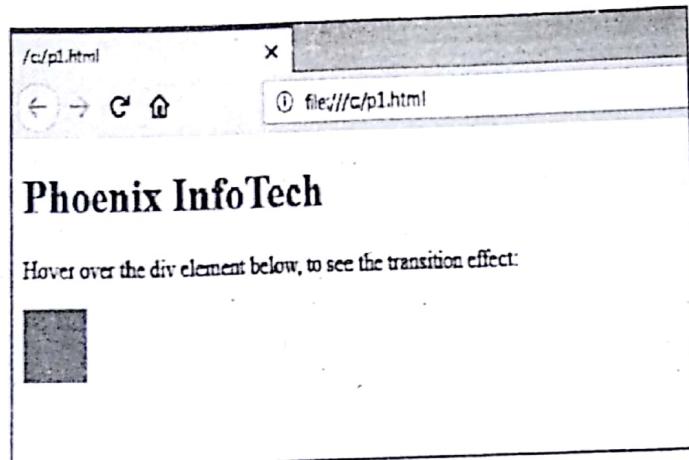
Geo Location

Up till now, the paper maps were ideal solution to search any locations. But now they are replaced by dedicated GPS navigation devices and mobile applications. These devices are found everywhere like in cars, on tablets and smart-phones. The most important feature of a navigation device is to detect the device's current position and update it as per changes. This assists user to reach from one location to other by providing directions. Near about all the current browsers support the geo location. Here we are going to learn the Geolocation API, which allows applications to detect and track the location of devices. There are number of applications of detecting device locations. For example on the web giants such as Google, Microsoft, and Yahoo use the user's location to personalize the SERPs (Search Engine Results Page) depending upon the location of user. Localization is another important application of geo-location.

Geolocation API?

The geolocation API allows us to easily fetch the geographical position of a user. If the browser support Geolocation API, one can easily get the Longitude and Latitude just using Javascript. The Geolocation API uses multiple sources to determine the most accurate position of the user. The location accuracy depends upon the best location source available. The most common sources of location information are IP address, Wi-Fi and Bluetooth MAC address, radio-frequency identification (RFID), Wi-Fi connection location, or device Global Positioning System (GPS) and GSM/CDMA cell IDs.

```
<h1>Phoenix InfoTech</h1>
<p>Hover over the div element below, to see the transition effect:</p>
<div></div>
</body>
</html>
```

Output

Q. 9 Create a webpage to display border effects by using transition effects in CSS3. Assume suitable parameters if required.

May 17

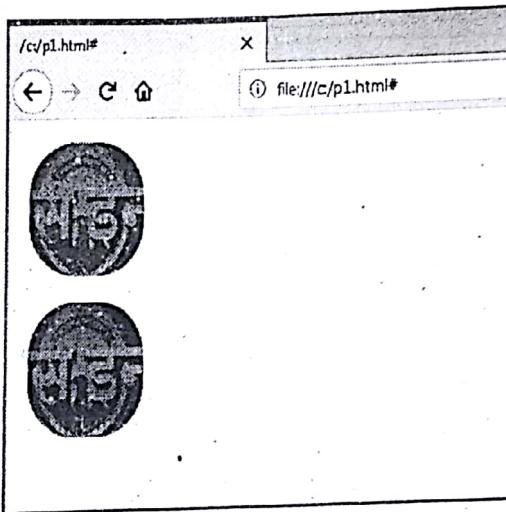
Ans. :**Code to display border effect with transition effect**

```
<html>
<head>
<style type="text/css">
body {
    padding: 10px 0 0 10px;
}
a {
    display:inline-block;
    -webkit-box-sizing:border-box;
    -moz-box-sizing:border-box;
    box-sizing:border-box;
    width:90px;
    height:90px;
    -webkit-border-radius:40px;
    -moz-border-radius:40px;
    border-radius:40px;
    background-image:url(Phoenix.png);
    background-repeat:no-repeat;
    background-origin:border-box;
    background-position:50% 50%;
    border-width:50px;
    border-color:rgba(0,0,0,0);
    -webkit-transition:0.5s ease;
    -moz-transition:0.5s ease;
    -ms-transition:0.5s ease;
    -o-transition:0.5s ease;
```

```

        transition:0.5s ease;
    }
    a:hover {
        border-width:0;
        border-color:rgba(0,0,0,0.5);
    }
}
a.one {border-style:solid;}
a.two {border-style:dashed;}
}
</style>
</head>
<body>
<a class="one" href="#">

```

Output

Q. 10 Explain CSS3 Transformations with example.

Dec. 15

Ans. :

CSS3 Transformation

CSS transforms are used to translate, rotate, scale, and skew elements. A transformation effect helps an element to change its shape, size as well as position. CSS supports both 2D as well as 3D transformations.

(A) CSS 2D Transforms

CSS provides following 2D transformation methods :

1. translate()

The translate() function is used to move an element from its current position (based on the arguments given for the X-axis and the Y-axis).

```

div {
    -ms-transform: translate(70px, 110px); /* IE 9 */
    -webkit-transform: translate(70px, 110px); /* Safari */
    transform: translate(70px, 110px);
}

```

2. rotate()

The `rotate()` function is used to rotate an element clockwise or anti-clockwise based on the given degree.

```
div {
    -ms-transform: rotate(75deg); /* IE 9 */
    -webkit-transform: rotate(75deg); /* Safari */
    transform: rotate(35deg);
}
```

3. Scale()

The `scale()` function is used to increase or decrease the size of an element (based on the arguments specified for the width and height).

```
div {
    -ms-transform: scale(3, 4); /* IE 9 */
    -webkit-transform: scale(3, 4); /* Safari */
    transform: scale(3, 4);
}
```

4. SkewX()

The `skewX()` function is used to skew an element along the X-axis by the specified angle.

```
div {
    -ms-transform: skewX(35deg); /* IE 9 */
    -webkit-transform: skewX(35deg); /* Safari */
    transform: skewX(35deg);
}
```

5. SkewY()

The `skewY()` and `Skew()` methods are same as of `skewX()`. Just `skewY()` method skew an element along the Y-axis while `skew()` method skew an element along both X and Y axis.

(B) CSS 3D Transforms

CSS provides following 3D transformation methods :

The `rotateX()` function is used to rotate an element around its X-axis at specified degree :

```
#Div1 {
    -webkit-transform: rotateX(120deg); /* Safari */
    transform: rotateX(120deg);
}
```

Similarly `rotateY()` and `rotateZ()` methods are used to rotate an element around its Y and Z axis.

Dec. 15

Q. 11 Explain input elements newly introduced In HTML5 with example.

Ans. :

Input element In HTML5

We are already familiar with the `input` element's `type` attribute. This is the attribute which determines what kind of form input will be generated. If it is skipped or if the browser could not understand, a textbox is created. The HTML5 provides nine additional input types for more data-specific UI elements and native data validation :

1. Input Type : Color

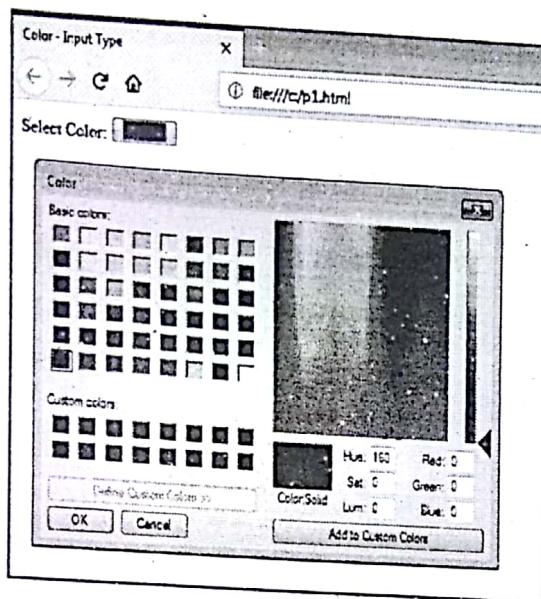
The color input type provides drop-down color picker from which user can select required color.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Color - Input Type</title>
</head>
```

```

<b></b>
<form>
<label>
    Select Color: <input type="color" name="selcolor">
</label>
</form>
</body>
</html>

```

Output**2. Input Type Email**

- The email input type is used to accept e-mail address from user. It is same as of standard text input type, but when used in combination with the required attribute, the browser checks whether the user has entered valid e-mail address or not.

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Email : Input Type</title>

<style type="text/css">
    input[type="email"]:valid{
        outline: 3px solid blue;
    }
    input[type="email"]:invalid{
        outline: 3px solid brown;
    }
</style>
</head>
<body>
<form>
<label>
    Enter your Email ID <input type="email" name="sremail" required>
</label>
</form>
</body>
</html>

```

Output

Email : Input Type X

← → C ⌂ ① file:///c/p1.html

Enter your Email ID

Please enter an email address.

3. Input Type Number

The number input type is used to accept numerical value. We can restrict the user to enter specific values with the help of additional attributes such as min, max, and step.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Number : Input Type</title>
<style type="text/css">
input[type="number"]::valid{
    outline: 3px solid blue;
}
input[type="number"]::invalid{
    outline: 3px solid brown;
}
</style>
</head>
<body>
<form>
<label>
    Select Number: <input type="number" value="100" min="100" max="200" step="5" name="srnumber">
</label>
</form>
<p><strong>Note</strong>: Only numbers between 100-200 are allowed</p>
</body>
</html>
```

Output

Number : Input Type X

← → C ⌂ ① file:///c/p1.html

Select Number:

Note: Only numbers between 100-200 are allowed. Please select a value that is no more than 200.

4. Input Type Range

This same as of number input just it provides easy control for entering a number. It is used to accept values in certain range.

Select Number: <input type="range" value="100" min="100" max="200" step="5" name="srnumber">

Output

5. Input Type Search

This type is used to create search fields. A search field is same as of a regular text field, but in specific browser such as Google Chrome and Apple Safari when user start typing in a search box a small cross appears on the right side of the field that allows user to quickly clear the search field.

Search the Website: <input type="search" name="srsearch">

Output

6. Input Type Tel

As the name suggests, this type is used for accepting a telephone number.

Telephone Number: <input type="tel" name="mytelephone" required>

Output

7. Input Type URL

The url input type is used to accept URL (web address). Browser will carry out validation on URL given by user and displays an error message for wrong URL.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>HTML5 URL Input Type</title>
<style type="text/css">
input[type="url"]:valid{
    outline: 3px solid blue;
}

```

Internet Programming (MU)

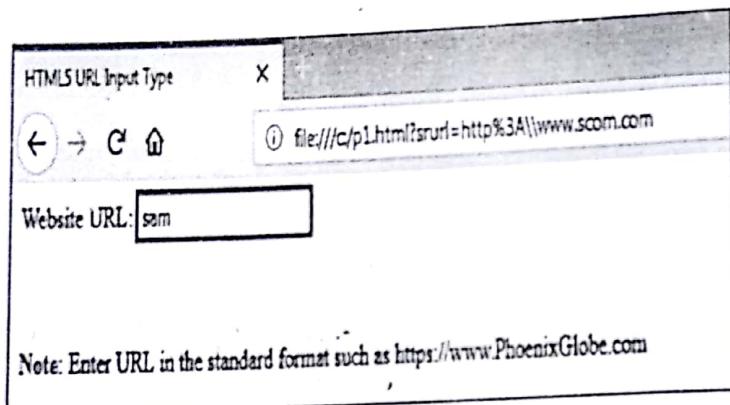
```

input[type="url"]:invalid{
    outline: 3px solid brown;
}

</style>
</head>
<body>
<form>
<label>
    Website URL: <input type="url" name="srurl" required>
</label>
</form>
<p> <strong>Note</strong>: Enter URL in the standard format such as https://www.PhoenixGlobe.com </p>
</body>
</html>

```

Output



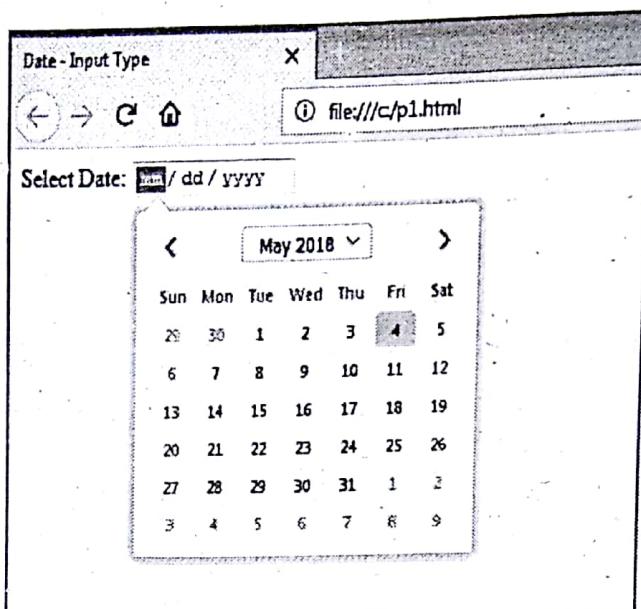
8. Input Type Date

The date input type provides a drop-down calendar from which user can select a date.

```

<!DOCTYPE html>
<html lang="en">
<head>
<title>Date - Input Type</title>
</head>
<body>
<form>
<label>
    Select Date: <input type="date" name="srdate">
</label>
</form>
</body>
</html>

```

Output**9. Input Type Datetime**

The datetime provides date time picker from which user can select a date and time along with time zone.

Select Date and Time: <input type="datetime" name="srdatetime">

10. Input Type Datetime-local

The datetime-local input type provides facility to select a local date and time. It does not include timezone information.

Select Local Date and Time: <input type="datetime-local" name="srlocaldatetime">

Output

Local Date & Time:	dd/mm/yyyy -- : -- PM
--------------------	-----------------------

11. Input Type Month

The month input type provides drop-down calendar from which user can select month and year.

Select Month: <input type="month" name="mymonth">

Output

Select Month:	February, 2019
---------------	----------------

12. Input Type Time

The time input type is used to accept time from user.

Select Time: <input type="time" name="srtime">

Output

HTML5 Time Input Type	X
← → C ⌂	file:///c/p1.html?srurl=http%3A\www
Select Time:	08 : 46 AM

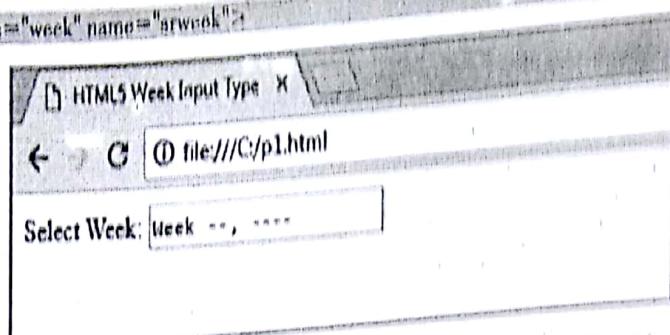
Internet Programming (MU)

13. Input Type Week

The week input type allows the user to select a week and year from a drop-down calendar.

Select Week: <input type="week" name="arweek">

Output



Chapter 4 : Rich Internet Application (RIA)

Ques. 16

Q. 1 Write short note on Rich Internet Application (RIA).

Ans. :

Rich Internet Application

A rich Internet application (RIA) is a Web application designed to deliver the same features and functions normally associated with desktop applications. RIA is built with very powerful development tools which enable it to run faster. RIA provides users a better visual experience and more interactivity as compared to traditional browser applications which just use HTML and HTTP. Nowadays, it is possible for programmers to embed nearly any functionality which they like inside a Web-based graphical interface, which makes it to view and behave as if it is traditional desktop application. Modern tools helps programmers to create complex application screens with the help of variety of mixed media like several fonts, bit-map and vector graphic files, animations, online conferencing, audio and video.

Such applications provide functionality which goes far beyond just reading and browsing, and they can be served up over the Web. These are called as **Rich Internet applications**. RIAs generally split the processing across the Internet/network divided by locating the user interface and related activity and capability on the client side, and the data manipulation and operation on the application server side. An RIA normally runs inside a Web browser and usually does not require software installation on the client side to work.

However, some RIAs may only work properly with one or more specific browsers. For security purposes, most RIAs run their client operating system on the client to the application server on the other side of the connection. This approach allows the client system to handle local activities, calculations, reformatting and so forth, thereby lowering the amount and frequency of client-server traffic, especially as compared to the client-server implementations built around so-called thin clients. One distinguishing feature of an RIA (in contrast to other Web-based applications) is the client engine that intermediates between the user and the application server. The client engine downloads when the RIA launches. The engine can be augmented during subsequent operation with additional downloads in which the engine acts as a browser extension to handle the user interface and server communications. The term "rich Internet application" was introduced in a white paper of March 2002 by Macromedia (now merged into Adobe).

Q. 2 Explain about the object that helps AJAX reload parts of a webpage without reloading the whole page. May 16

Ans. :

AJAX

AJAX uses the XMLHttpRequest object to communicate with the server. The Fig. 4.1 shows the flow of AJAX. Here you will get exact idea about the working of AJAX. An important role is played by the XMLHttpRequest object in AJAX processing. Request is sent by the user through UI (User interface) and a JavaScript call goes to XMLHttpRequest object. Using the XMLHttpRequest object, the HTTP Request sent to the server. Now the server interacts with the database using any of the server side scripting language like JSP, PHP, Servlet, or ASP.net.

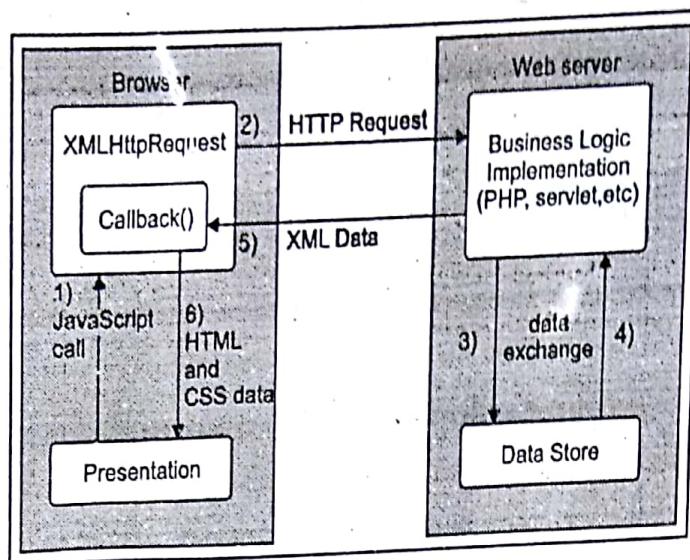


Fig. 4.1

Data is retrieved from the database as per the requirement. Server sends the data in the form of XML or JSON through the XMLHttpRequest callback function. Then the data is displayed using HTML and CSS on the browser.

Dec. 16

Q. 3 What are two methods in ajax to update part of webpage? Describe them in detail.

Ans. :

Methods in ajax

In Ajax, we frequently want to update just part of a Web page, not the whole page. Two methods enable us to do this : the `insertAdjacentHTML` method, which lets us to insert HTML next to an element that already exists, and the `insertAdjacentText` method, which lets us to insert text in the same way. We can determine where the new text or HTML will go with respect to the already existing element by passing the constants `BeforeBegin`, `AfterBegin`, `BeforeEnd`, or `AfterEnd` to `insertAdjacentHTML` and `insertAdjacentText`. The example `insertAdjacent.html` uses the `insertAdjacentHTML` method. This page starts with just a button, as shown in Fig. 4.2. When you click the button, the code inserts a new text field into the Web page, as shown in Fig. 4.3.

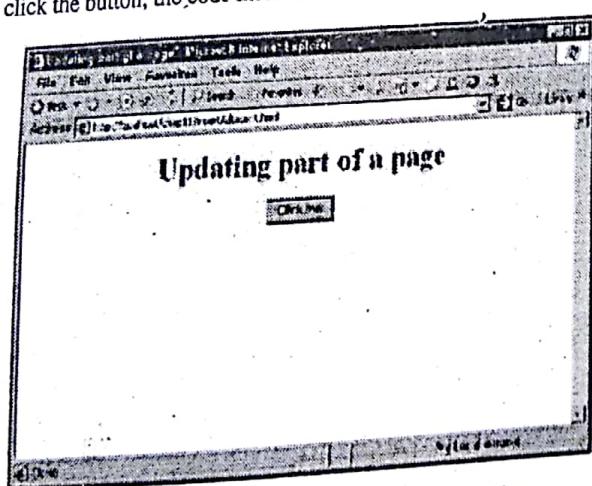
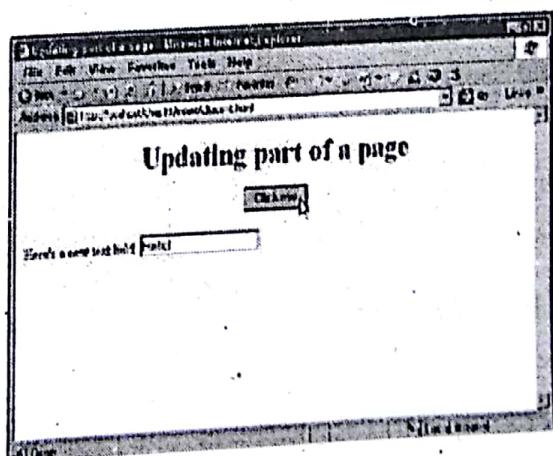
Fig. 4.2 : The `insertAdjacent.html` application

Fig. 4.3 : Inserting a new text field into a page

Here's how it works: the application contains a `<div>` element with the ID `div1`:

```
<body>
<center>
<h1> Updating part of a page </h1> </center>
<div id="div1"></div>
</body>
```

Internet Programming (MU)

And that `<div>` element contains the button, which is connected to a JavaScript function named `update`:

```
<body>
<center>
<h1> Updating part of a page </h1> </center>
<div>
  <input type=button value="Click Me!" onclick="update()">
</div>
</body>
```

In the `update` function, we can call the `<div>` element's `insertAdjacentHTML` method to insert the text field next to the `<div>` element:

```
<head>
<title> Updating part of a page </title>
<script language="JavaScript">
function update()
{
  div1.insertAdjacentHTML(...);
}
</script>
</head>
```

Here's how we can insert the new text field after the end of the `<div>` element:

```
<head>
<title> Updating part of a page </title>
<script language="JavaScript">
function update()
{
  div1.insertAdjacentHTML("AfterEnd", "<p>Here's A new text field: <input type=text value='Hello!'></p>");
}
</script> </head>
```

Using `insertAdjacentHTML` and `insertAdjacentText`, we can insert HTML and text into a Web page, adjacent to another Web page item.

Dec. 16

Q. 4 Explain the following AJAX patterns :

- 1) Submission Throttling 2) Predictive Fetch 3) Fallback patterns

Ans. :

Design pattern of AJAX are following

The three important design patterns of AJAX are following :

1. Submission Throttling

If retrieving data from the server is one part of the problem sending data to server is another. Science in AJAX page refreshes needs to be avoided, it is important to know when to send user data to the server. One approach that could be taken is to send data to server on every user interaction. But this results in a lot of requests submitted to the server in a short period. In case of submission throttling, the data to be send to the server is buffered on the client. This data is then sent to the server at predefined times. The delay from typing to sending data is fine-tuned such that it doesn't seem like a delay to the user. Then a client side function is invoked that begins buffering the data. It can be sent at a predefined time interval. This determination depends on the use case being used. After the data is sent, the application continues to gather data.

2. Predictive Fetch

In case of traditional web applications, the application reacts only when there is an interaction. This is called "fetch on demand". The user action tells the server what data should be retrieved. In the predictive fetch algorithm, the application guesses what the user is going to do next and retrieves the appropriate data. Determining the future action of the user is just a guess based on the user's intentions. For eg : say a user is reading an online article of 3 pages. It can be assumed here that if the user is reading the 1st page for few seconds, the person will also be interested in reading the 2nd page. Hence the 2nd page can be downloaded at the background before the user explicitly clicks on the 'Next'. Therefore when the user clicks on next, the 2nd page instantaneously appears reducing the response time. Similarly the 3rd page can be downloaded when the user reads 2nd page for a few seconds. This extra data being downloaded is cached on the client. Some approach can be applied in emails. If a person starts composing a mail, it is logical to anticipate that the mail would be sent to someone in the address book so this can be preloaded and kept. By using AJAX to fetch information related to any possible next step, can overload the server. Therefore this algorithm has to be implemented only when it is logical to assume that information will be requisite to completing the user's next request.

3. Fallback patterns

All the methods work fine when there is no problem at the server side.

The following problems can occur:

- (1) The request might never make it to the server.
- (2) An error might occur at the server.

Q. 5 Draw the diagram of AJAX application model and traditional applications web model and compare them.

Dec. 16

Ans. :

(A) Traditional Web Applications

Fig. 4.4 represents the typical model of client and server interactions in a traditional web application, such as one which includes user registration form.

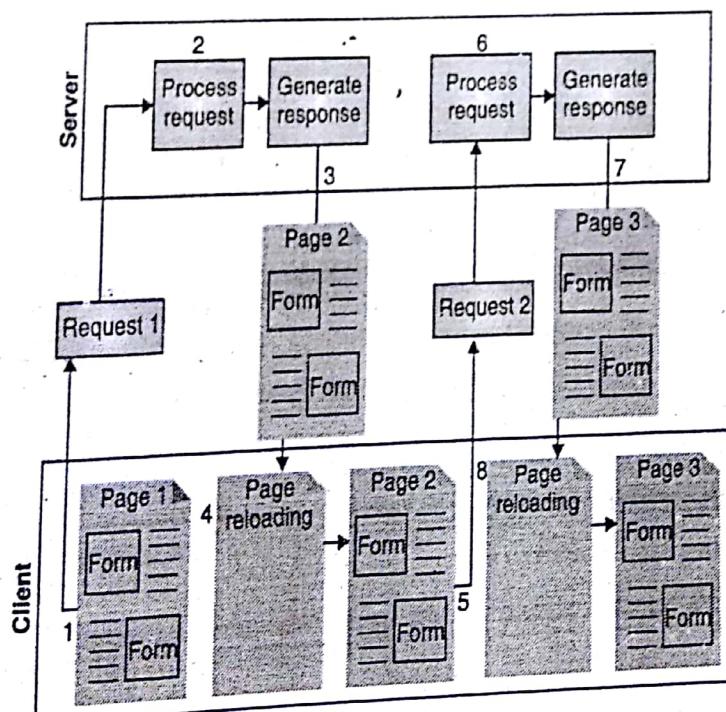


Fig. 4.4 : Traditional Model

In the first step, the form is filled by the user and then submitted. (Fig. 4.4, Step 1). A request is generated by browser and sent to server. The server receives the request and processes it (Step 2). After processing the request, a response is generated by the server and the response containing the exact page that the browser will render (Step 3) is sent to the client.

forces the browser to load the new page (Step 4) and for the time being makes the browser window blank. Meanwhile the client has to wait for the server to give response and reloads the whole page containing the data from the server. In time being when the asynchronous request is processing on the server, the user is not able to interact with the client. Frequent long periods of waiting impacts the site performance and let some users to refer to the World Wide Web as the "Wide Wait". If the further user interacts with another form and submits it, the process is repeated (Steps 5-8). Basically this model is designed for a web of hypertext documents. As the web evolved into a full-scale applications platform, the model represented in Fig. 4.4 leads to irregular application performance. The full-page refresh needs the users to reset their understanding of the full-page contents. Users started to insist a model which would generate the responsive feel of desktop applications.

May 17, Dec. 17

Q. 6 Explain in detail AJAX Web application model with neat diagram.

Ans. :

AJAX web application model

To manage communication between client and server, Ajax applications add a layer between them (Fig. 4.5). When user is interacting with page, an XMLHttpRequest object is generated by the client to manage a request (Step 1). The XMLHttpRequest object further sends the request to the server (Step 2) and waits for the response. The requests are asynchronous, that means it is possible for user to carry on interacting with the application on the client-side while the server processes the previous request simultaneously. Other user interactions generate some more requests to the server (Steps 3 and 4). As soon as the server responds to the first request (Step 5), the XMLHttpRequest object which has issued the request gives call to a client-side method to process the data which is returned by the server.

This method which is called as callback method uses partial page updates (Step 6) for the purpose of displaying the data in the existing web page without the process of re-loading the entire page. At the same time, the server can respond to the second request (Step 7) whereas the client-side may be ready to implement another partial page update (Step 8). Just the designated part of the page is updated by the callback method. These kinds of partial page up-dates are useful in making the web applications more responsive, making them feel more like desktop applications. Entire page is not loaded by the web application when the user interacts with it.

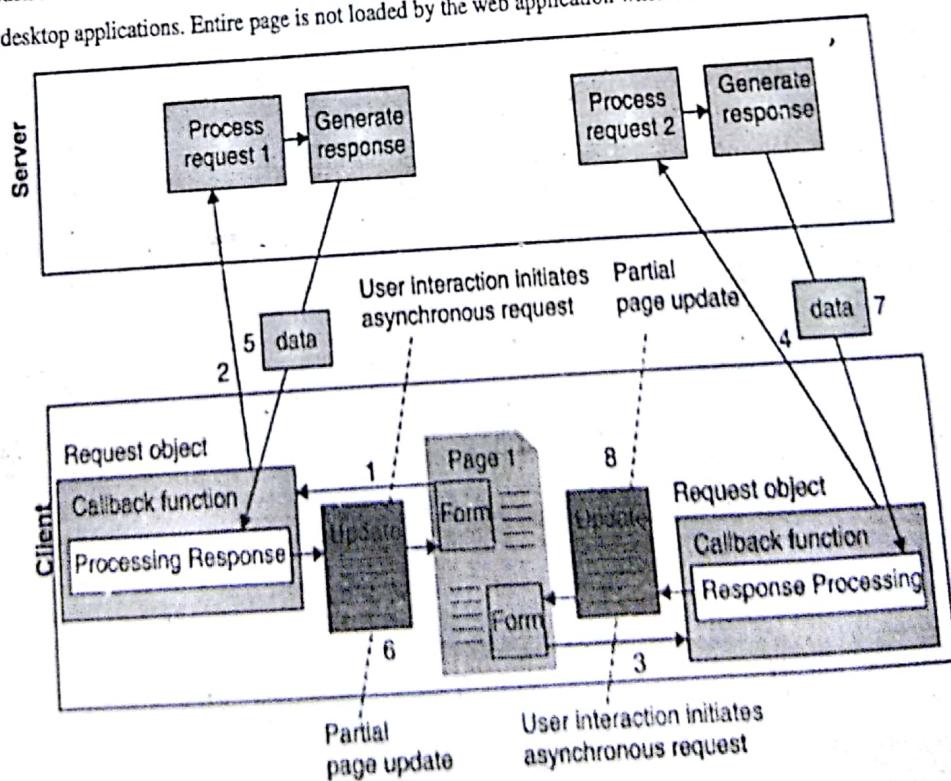


Fig. 4.5 : AJAX Model

Q. 7 Explain detail RUI implementation using AJAX.

Ans. :

May 16

RUI implementation using AJAX

Website design has recently gone through a good change when they start to become more user friendly. The use of a Rich User Interface is slowly becoming more evident, especially in E-Commerce Web design. When Rich User Interface is used on a Website, it is refreshingly different and visitors are apt to notice the difference in Web design right away. Although it has been promised for a long time, it seems that the day of its arrival has come. The applications that provide the new enhanced user interactivity are much better developed and can really make a difference between an ordinary Web design and a quality Web design. Instead of the old-fashioned static approach, there are now many options that allow developers to put their Webpage years ahead of their competitors.

So much more can be done with tools like AJAX to enhance the beauty, usefulness, and interactivity of a Webpage that a designer needs to consider between many possibilities. Not only is it more eye appealing, but the changing colors of buttons when pressed, the boxes that enlarge on mouse overs, identifying tags, etc., all work together to make a visitor feel better. Another good thing is that more Websites are taking advantage of less crowded Web pages. There's more white space which gives it a more relaxed and inviting look – causing people to want to stay longer. It is possible to achieve rich user interfaces with the help of combination of dynamic HTML elements like HTML and JavaScript. But, such an interface has limited scope to client-side behavior and has minimal functional implications because it lacks the server-side interactions.

The power of AJAX is basically included in its capacity to offer richer interface by providing its dynamic user interface supported by powerful functionality by the use of faultless server-side invocation power. AJAX offer features to individual user interface components so that they are able to communicate with the server and exchange data without the necessity of refreshing the entire screen. To achieve this a process is used known as Web Remoting. There are multiple ways to perform this process. More frequently used approaches by the latest browsers are IFrames and XMLHttpRequest. It is possible to complement Dynamic HTML with either of these methods to create AJAX functionality.

Q. 8 What do you mean by JSON?

Dec. 16

Ans. :

JSON

In computing, JavaScript Object Notation or JSON is an open-standard file format that uses human-readable text to transmit data objects consisting of attribute-value pairs and array data types (or any other serializable value). It is a very common data format used for asynchronous browser/server communication, including as a replacement for XML in some AJAX-style systems. JSON is a language-independent data format. It was derived from JavaScript, but as of 2017 many programming languages include code to generate and parse JSON-format data. The official Internet media type for JSON is application/json. JSON filenames use the extension .json.

Douglas Crockford originally specified the JSON format in the early 2000s; two competing standards, RFC 7159 and ECMA-404, defined it in 2013. The ECMA standard describes only the allowed syntax, whereas the RFC covers some security and interoperability considerations.

Uses of JSON

1. JSON is used in JavaScript based applications created for web related features.
2. The data transfer between a server and web applications is done by JSON.
3. Over the network the structured data can be serialized and transferred through JSON.
4. JSON format is used by Web services and APIs to provide public data.
5. Most modern programming languages use JSON

Characteristics of JSON

- (1) It is a lightweight text-based interchange format.
- (2) JSON is easy to read and write.
- (3) JSON is language independent.

Q. 9 Why to use JSON over XML?

Dec. 16

Ans. :

Parameter	JSON	XML
Long form	JSON stands for JavaScript Object Notation.	XML stands for eXtensible Markup Language.
Operations	The read and write operations are simple in JSON	XML is less simple than JSON.
Simplicity	JSON is easy to learn.	XML is less easy than JSON.
Orientation	JSON is data-oriented.	XML is document-oriented.
Array	JSON supports array.	XML doesn't support array.
Readability	JSON files are more human readable than XML.	XML files are less human readable.

Dec. 16, Dec. 17

Q. 10 Define and describe mashups. What are primary reasons for success of mashups?**Ans. :****Mashups**

A **mashup** in web development, is a web page, or web application, that uses content from more than one source to create a single new service displayed in a single graphical interface. For example, a user could combine the addresses and photographs of their library branches with a Google map to create a map mashup. The term implies easy, fast integration, frequently using open application programming interfaces (open API) and data sources to produce enriched results that were not necessarily the original reason for producing the raw source data. The main characteristics of a mashup are combination, visualization, and aggregation. It is important to make existing data more useful, for personal and professional use. To be able to permanently access the data of other services, mashups are generally client applications or hosted online.

In the past years, more and more Web applications have published APIs that enable software developers to easily integrate data and functions the SOA (Service Oriented Architecture) way, instead of building them by themselves. Mashups can be considered to have an active role in the evolution of social software and Web 2.0. Mashup composition tools are usually simple enough to be used by end-users. They generally do not require programming skills and rather support visual wiring of GUI widgets, services and components together. Therefore, these tools contribute to a new vision of the Web, where users are able to contribute. The generation of mashups is enabled by a multipart ecosystem of interconnected data providers, mashup platforms as well as the end users. Mashups have recently exploded on the web, for two main reasons.

First, many of the major internet companies, such as Yahoo! (www.yahoo.com), Google (www.google.com), and Amazon (www.amazon.com), have opened up their data to be used with other data sources without a lengthy licensing negotiation. In just a minute or two, user can set up and use the data resources they make available. The other reason for this rapid growth is the advent of new tools that make creating mashups easy for anyone, regardless of their technical know-how.

Popular Mashups

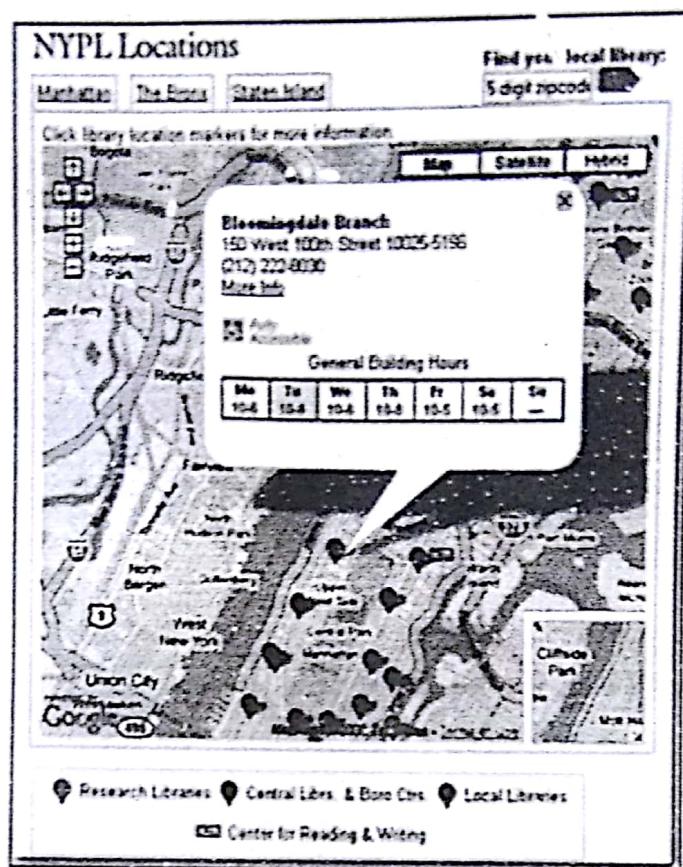


Fig. 4.6 : Google Maps mashup of New York Public Library branches

The most popular type of mashup is a map mashup. Map mashups make up 36 percent of the mashups tracked by ProgrammableWeb (www.programmableweb.com), the most comprehensive listing of mashups.

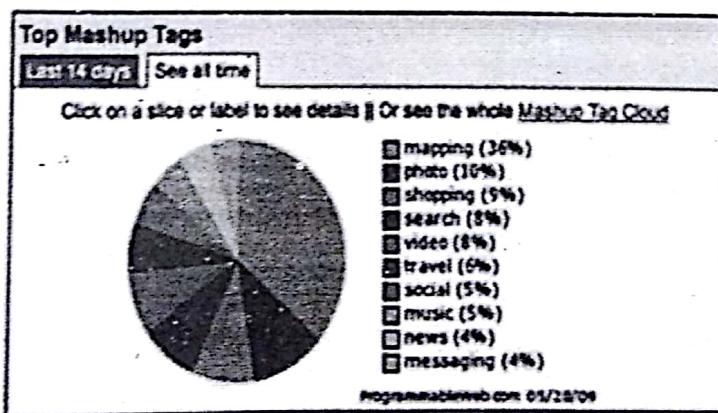


Fig. 4.7 : Pie chart showing popularity of different types of mashups

Fig. 4.7 shows the distribution of mashups by type. Other types of popular mashups mix video and photos. For example, the Viral Video Chart (viralvideochart.unrulymedia.com) site tracks YouTube (www.youtube.com), MySpace (www.myspace.com), and Google Video (video.google.com) in order to identify the most talked about new clips, overall and by category Fig. 4.8.



Fig. 4.8

There are many wonderful mashup services built using photos. Here are three examples :

- 1) **Colr Pickr** (krazydad.com/colrpickr), developed by Jim Bumgardner, lets us search Flickr (www.flickr.com) photos by color. Flickr is a widely used photo-sharing site (offering both free and professional accounts) that attracts many libraries and librarians. Use Colr Pickr to click on a color in a photo or color wheel, and it will retrieve photos that have a large concentration of that color.
- 2) There are also mashup tools that use photographs to create books, posters, magazine covers, and so on. Most of us have received photograph booklets comprised of stunning images with short captions. One way to create these photograph books is to use a mashup tool called Bookr (www.pimpampump.net/bookr). Start by searching Flickr to locate images. Then, add these photos to a booklet page and type in the caption. Save work, and email the book to friends or publish it on a blog or library website.
- 3) A mashup tool called Ad Generator (theadgenerator.org), created by Alexis Lloyd for his MFA thesis project is intended to inform, enlighten, and entertain site visitors. The mashup randomly combines slogans and images to explore the relationship among language, manipulation, and images. Mashups come in all shapes and sizes, from the very simple to the complex. Some mash up search results, others introduce interesting visualizations, and still others aggregate and combine newsfeeds. Libraries have lots of opportunities to use mashups to help liven up their websites, deliver new and interesting services, or entertain website visitors.

Q. 11 Explain in detail architecture of simple mashup on web server. List out advantages and disadvantages. May 16

Ans. :

Simple mashup on web server

Web server is an integral part of any web system. Web server has the main responsibility to up the web pages. There are number of web servers such as IIS (Internet Information Server), PWS (Personal Web Server) both work in Microsoft environment. Other Web Servers such as Apache Tomcat serves java environment. Most of the time the entire process of mashing is handled by the web server while the browser just sits and waits for response.

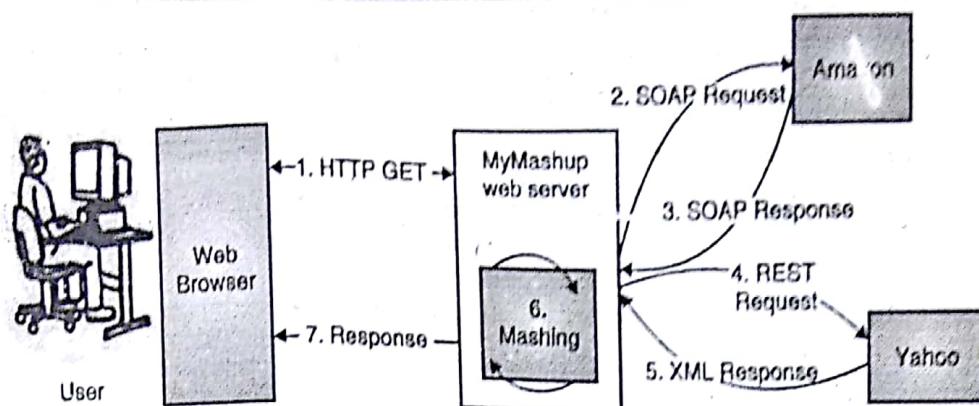


Fig. 4.9 : Browser, Server and partner site interactions with typical mashup

The Architecture of Web server Mashups working is as per the following steps :

1. The web browser uses HTTP to send request for a page to the server.
2. The web server constructs the page by connecting to the source or partner sites, which include Google, amazon, Yahoo, etc. In our example Amazon gets the first request from the browser using SOAP over HTTP.
3. The web server gets the SOAP response back from Amazon.
4. Our example has the second request being sent to Yahoo through REST.
5. The web server receives plain old XML over HTTP from Yahoo.
6. The web server now combines and rationalizes the data in the most suitable manner and sends the response back.
7. The data resulting from the web server is combined in an HTML format and the response is sent back to the browser.

Advantages of Mashing on the Web Server

- (1) Browser remains entirely separated from the partner sites, which are actually responding to the request by providing data.
- (2) The web server itself works as an aggregator and proxy for the responses.

Disadvantage of Mashing on the Web Server

It involves a full page request being sent by the web server, thereby reducing scalability.

Q. 12 Explain in detail architecture of mashups in a JSON implementation with neat diagram.

May 15, Dec. 15, Dec. 17

Ans. :

Dataflow in JSON Implementation

Nowadays, the JSON becomes very popular in mashup community because of its features such as readability and simplicity. JSON, being a notation, provides a way in which objects are written so that human beings can read it easily. It has built-in JavaScript feature that made JSON a viable mashup technology. Here we are going to discuss the communication mechanism of mashup with the help of JSON.

The architecture of mashups in JSON implementation is shown in Fig. 4.10. The flow of JSON mashups that use dynamic Script method goes in the following steps :

1. The flow of the process starts with the browser sending request to the server by using HTTP GET.
2. The Web Server responds with a page that includes the following couple of important JavaScript functions :
 - A parsing function that expects JavaScript objects to be parameters.
 - The Dynamic Script method is the core of the initiation script through which a new script tag is added to the page, specifying the source for that script tag to be the Uniform Resource Locator (URL) at some partner site.
3. The source code for the new script tag gets loaded by the browser.
4. Amazon receives an HTTP Get request sent from the browser using the loaded script.

Gardill

being serialized into a JSON Object, is served by the partner site.

The render function wraps the JSON script, and the JavaScript entirely becomes the content for the script tag.

Execution of JavaScript is tried for execution by the browser, which calls the render method from step 2.

The browser invokes the render method and evaluates the JSON script, which is converted into a JavaScript object. The data defined in the render method is pushed into the page after the render method, which uses the new JavaScript object in its execution.

Architecture of Mashling with JSON

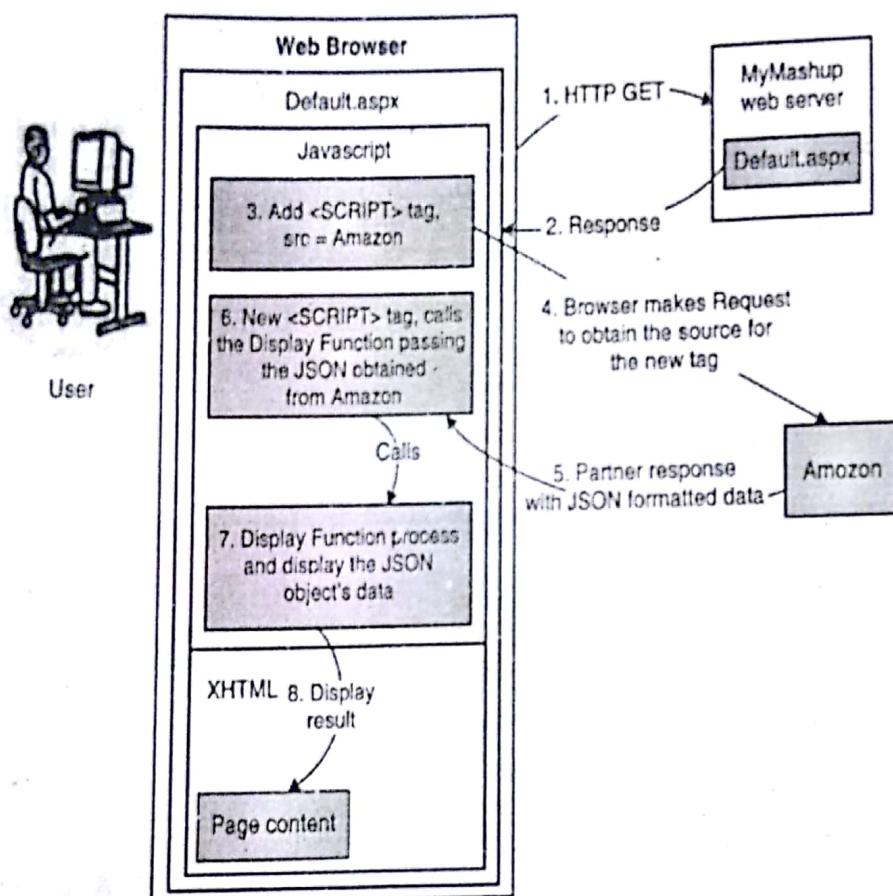


Fig. 4.10 : Dataflow in JSON environment

Advantages of Mashling with JSON

1. The communication path is considered as the main benefit of JSON approach. The browser can directly communicate with the partner site without any help of server. This results in reduction of loads on the server since browser handles the entire process.
2. The built in objects provided by JSON are always easier to handle for the developers as compared to processing XML using XSL or XPath in JavaScript.
3. JSON is readable.

Disadvantages of Mashling with JSON

1. Since the browser directly communicates with the partner site, there is no possibility of consolidation of data on the server.
2. Number of APIs does not support JSON.

Chapter 5 : Server Side Programming : PHP

Q. 1 What are the advantages of PHP over scripting languages?

Ans. :

Advantages of PHP over scripting language

1. Simple

PHP is very simple and easy to use scripting language. PHP mostly use syntax of C programming. The rules and regulations in PHP are simple to understand and follow.

2. Interpreted

Rather than compiled, the PHP code is interpreted. There is no need of any compiler.

3. Faster

The execution speed of PHP is very fast as compare to other scripting languages like ASP and JSP

4. Open Source

PHP is open source. Means the software do not have any license charges and it is totally free of cost. It is freely available on internet.

5. Platform Independent

Once created, the PHP script can be executed on any machine irrespective of its platform. PHP script can be executed on number of platforms like Linux, Unix, Mac OS and various Windows operating systems.

6. Efficiency

The efficiency of PHP is very good. PHP provides various advanced features through OOPS concepts and also a rich set of function library which makes it efficient.

7. Flexibility

PHP is very flexible language. It can be easily embedded with HTML. All the HTML tags can be used in PHP with all their attributes. PHP can also integrate with other scripting languages like JavaScript or VBScript.

Q. 2 Discuss about various control structures used in PHP.

Ans. :

Control structure in PHP

(i) Conditional Statements

Use

Conditional statements are used to check conditions depending upon which the flow of program can be decided. PHP supports following three conditional statements:

- (i) **if...else statement** - It used to check single condition .If the condition is true then one block of code is executed and if condition is false another block of code is executed.
- (ii) **Else if statement** - This is also known as else if ladder. It is used to check multiple conditions.
- (iii) **switch statement** - It is a multi way decision making statement. It is used to check multiple conditions.

(i) The if...else Statement

Use

It used to check single condition .If the condition is true then one block of code is executed and if condition is false another block of code is executed.

Internet Programming (MU)

Syntax

```
if (condition)
    statements;
else
    statements;
```

(II) The else if ladder Statement

Use

This is also known as else if ladder. It is used to check multiple conditions

Syntax

```
if (condition)
    statements;
elseif (condition)
    statements;
else
    statements;
```

(III) The Switch Case Statement

Use

It is a multi way decision making statement. It is used to check multiple conditions. Irrespective of number of conditions, unlike else if we have to use single opening and closing curly brackets in switch case.

Syntax

```
switch (expression)
{
    case constant_expression:
        statements;
        break;
    case constant_expression:
        statements;
        break;
    default:
        statements;
}
```

Expression

It is usually name of a variable, value of which we want to check.

Constant_expression

These are the constant values with which the value of expression is compared. The statements in the case are executed, whose value matches with the expression. If none of the constant expression is matched with the expression, then the statements written in default are executed.

(II) Loops (Loop statement)

Use

Loops are used to execute specific task repeatedly in our program. Rather than writing the code again and again we can use the concept of loop. There are various situations when we may want to execute specific task multiple times. For example student mark sheet. Here we want to accept details from student and want to generate the marks sheet. This task is obviously repeated for number of students. In such situations we use the loops.

(i) While loop

While loop is considered as an entry controlled loop. That means the condition is given at the beginning of loop. If the given condition does not satisfy, the loop statements never get executed. If the condition is satisfied, then loop statements are repeatedly executed until the condition is satisfying. Once the condition becomes false, the control exits the loop.

Syntax

```
while(condition)
{
    Statements;
}
```

(ii) do while loop

The do while loop is exit controlled loop. That is the condition is checked at the end of loop. Hence even if the condition does not satisfy, the loop statements will be executed at least once.

Syntax

```
do
{
    Statements;
}while(condition);
```

(iii) For loop

In for loop the initialization, condition and increment or decrement of loop variable is implemented in a single statement. This loop is basically used to minimize the code.

Syntax

```
for(initialization; condition; increment/decrement)
{
    Statements;
}
```

Q.3 Explain various built in String functions in PHP.

Ans. :

PHP string built in function

- PHP provides various types of functions to access and manipulate the strings. We will see some important functions.

(1) PHP strtolower() function

The strtolower() function converts the string into lowercase format and return it.

Syntax

```
string strtolower ( string $string )
```

Example

```
<?php
$string1 = "ISHITA";
$string1 = strtolower($str);
echo $string1;
?>
```

Output

```
ishita
```

Output

```
ISHITA
```

(2) PHP ucfirst() function

This function convert the first character of string in uppercase and then return the string.

Syntax

string ucfirst (string \$str)

Example

```
<?php  
$str1="kunal";  
$str1=ucfirst($str1);  
echo $str1;  
?>
```

Output

Kunal

(3) PHP lcfirst() function

This function converts the first character of string in lowercase and then return the string.

Syntax

string lcfirst (string \$str)

Example

```
<?php  
$str1="Phoenix";  
$str1=lcfirst($str1);  
echo $str1;  
?>
```

Output

phoenix

(4) PHP ucwords() function

This function converts the first character of each word in uppercase in the sentence and return the string.

Syntax

string ucwords (string \$str)

Example

```
<?php  
$str1="my name is kunal";  
$str1=ucwords($str1);  
echo $str1;  
?>
```

Output

My Name Is Kunal

(5) PHP strrev() function

The strrev() function reverses the string and return it.

Syntax

string strrev (string \$string)

Example

```
<?php  
$str1="Rahul";  
$str1=strrev($str1);  
echo $str1;  
?>
```

Output

luluR

(6) PHP strpos() function

This function searches substring in the main string. If a substring is found, the function returns position of character of the first match. If not found, it will return FALSE.

```
<?php
echo strpos("Hello friends!", "friends");
?>
```

Output

6

(7) PHP str_replace() function

This function replaces the existing characters with new given characters in a string.

```
<?php
echo str_replace("friends", "world", "Hello friends");
?>
```

Output

Hello world

(8) PHP chr() function

The chr() function returns a character of given ASCII value.

```
<?php
echo chr(52)           // Decimal value
echo "<br>".chr(052); // Octal value
echo "<br>".chr(0x52); // Hex value
?>
```

Output

4

*

R

(9) PHP strcmp()

This function compares two strings. It returns 0 if the two strings are equal. It is case sensitive. The chr() function returns a character of given ASCII value.

```
<?php
echo strcmp("Hello", "Hello");
?>
```

Output

0

Q. 4 Write note on PHP Framework.**Ans. :****PHP framework**

A framework is a structure that developers choose to build their application. It determines the structure of the application and facilitates it to connect with many different API's. A proficient PHP framework enables developers to develop PHP application faster, efficiently and assist in building stable applications thereby reducing the amount of repetitive coding for PHP programmers. Frameworks provide scaffolding features that facilitates the development team to build faster and cleaner application. They often provide toolsets for both the UI components and the database access. A PHP Framework is a basic platform that allows us to develop web applications. In other words, it provides structure.

By using a PHP Framework, programmer will end up saving loads of time, stopping the need to produce repetitive code, and able to build applications rapidly. PHP operates on the Model View Controller (MVC) fundamentals. MVC is an architectural pattern featured in various popular programming languages which breaks apart the domain logic from user interface. The domain logic is the function that handles information exchange between database and user interface.

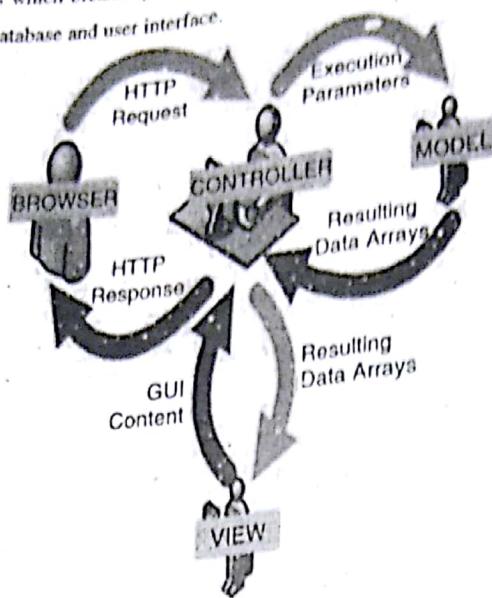


Fig. 5.1 : MVC Architecture

Therefore programmers are able to modify the domain logic and most importantly for designers, the user interface separately. This removes a lot of confusion and simplifies the entire developmental process. When we refer to MVC we generally perceive it as this: The M (Model) stands for the raw data, the V (view/user interface) represents what's actually being viewed, and C (controller) is in fact the domain logic. Once one is able to make sense of how MVC works, then PHP Frameworks become much more clearer and easier to use.

Q. 5 What are the aspects of proficient PHP Framework?

Ans. :

Aspects of proficient PHP framework

1. Database Support

It is one of the most crucial aspects of every PHP development framework. We need to decide the framework depending on the database we are going to use for web application.

For Example : 'CodeIgniter' supports MySQL, Oracle, and SQLite, whereas the 'Kohana' framework doesn't.

2. Community and Documentation

The framework should be supported by a strong community, not just in terms of size but also in terms of activity and usefulness. Even if it's a small community, user should be able to get ample support from the community. A PHP framework should also have good documentation. It should be comprehensive and up-to-date. The user guide should be relatively easy to follow.

3. Model View Controller Architecture

A framework should also use the Model View Controller architecture. Most of the good frameworks also offer libraries, plug-ins, helpers, and extensions. It's good to find a framework that has at least two of these options.

Chapter 6 : Web Extensions and Web Services

Q. 1 What do you mean by XML?

Ans. :

Dec. 16

XML

- (1) XML stands for eXtensible Markup Language.
- (2) XML is basically designed to store and transport data.
- (3) XML supports both human- and machine-readable data format.
- (4) XML was designed to be self-descriptive.
- (5) XML is a W3C Recommendation.
- (6) XML is a simple text-based format which represents information in structured formats like documents, transactions, data, invoices, books etc. This language is derived from comparatively older standard format called SGML (Standard Generalized Markup Language), to make it more suitable for Web-use.

Q. 2 What are the advantages of XML?

Ans. :

Advantages of XML

There are several reasons for the need of XML as follows :

1. Simplicity

XML can be easily understood. We can create our own tags and build the application. We are free to develop the system as per our requirements and with our own conventions. This makes the thing very simple for us.

2. Organization

The design process can be segmented to build the platform. Data can be stored on one page while the formatting rules can be stored on another page. It is possible to create the data page to store the content first and later on we can work on design. XML allows us to create the website in stages and stay organized in the entire process.

3. Accessibility

Data can be divided in XML. This makes the access of data easy and fast whenever there is need of making change in the data.

4. Standardization

XML is an international standard. This means XML document can be viewed anywhere in the world.

5. Multiple Applications

"Write once, use anywhere, any number of times" rule is applied to XML. For XML data we can create any number of display pages as we want. XML allows us to create various styles and formats for a single page as per requirement.

Q. 3 Explain the concepts of Entity and Attributes in XML.

Ans. :

Dec. 16

Different concept of XML

1. Character

An XML document is a string of characters. Almost every legal Unicode character may appear in an XML document.

2. Processor and Application

The processor analyzes the markup and passes structured information to an application. The specification places requirements what an XML processor must do and not do, but the application is outside its scope. The processor (as the specification calls) often referred to formally as an XML parser.

3. Markup and Content

The characters making up an XML document are divided into markup and content, which may be distinguished by the application of simple syntactic rules. Generally, strings that constitute markup either begin with the character < and end with a >, or they begin with the character & and end with a ;. Strings of characters that are not markup are content.

4. Tag

A tag is a markup construct that begins with < and ends with >. Tags come in three flavors:

- (1) Start-tag, such as <section>
- (2) End-tag, such as </section>
- (3) Empty-element tag, such as <line-break />

5. Element

An element is a logical document component that either begins with a start-tag and ends with a matching end-tag or consists only of an empty-element tag. The characters between the start-tag and end-tag, if any, are the element's content, and may contain markup including other elements which are called as child elements.

An example is

```
<greeting>Hello, world!</greeting>
```

Another is

```
<line-break />
```

6. Attribute

An attribute is a markup construct consisting of a name-value pair that exists within a start-tag or empty-element tag.

Example

```

```

where the names of the attributes are "src" and "alt", and their values are "Rose.jpg" and "Rose" respectively.

Another example is <step number="3">Connect A to B.</step>, where the name of the attribute is "number" and its value is "3". An XML attribute can only have a single value and each attribute can appear at most once in each element. In the common situation where a list of multiple values is desired, this must be done by encoding the list into a well-formed XML attribute with some format beyond what XML defines itself. Usually this is either a comma or semi-colon delimited list or, if the individual values are known not to contain spaces, a space-delimited list can be used. <div class="inner greeting-box">Welcome!</div>, where the attribute "class" has both the value "inner greeting-box" and also indicates the two CSS class names "inner" and "greeting-box".

7. XML Declaration

XML documents may begin with an XML declaration that describes some information about themselves. An example is <?xml version = "1.0" encoding = "UTF-8"?>.

Q. 4 Differentiate between HTML and XML.

Dec. 16

Ans. :

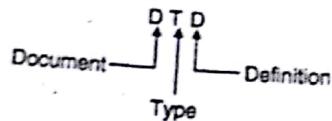
Sr. No	Parameter	HTML	XML
1.	Long Form	HTML stands for HyperText Markup Language.	XML stands for eXtensible Markup Language.
2.	Purpose	HTML was designed to display data which concentrates on look of data.	XML was designed to store and transport the stored data.
3.	Markup Language	HTML is a markup language itself.	XML provides a framework for defining markup languages.
4.	Use	HTML is used for presentation purpose.	XML is not used for presentation purpose.

Sr. No	Parameter	HTML	XML
5.	Case Sensitivity	HTML is case insensitive.	XML is case sensitive.
6.	Tags	HTML uses only predefined tags.	XML allows user defined tags
7.	Restrictions	HTML is flexible related to syntax. E.g. no need to close the tags.	XML makes it compulsory for the user to close all the tags.
8.	White Space	HTML does not preserve white space.	XML preserves white space.
9.	Static or Dynamic	HTML is static.	XML is dynamic.

Q. 5 Explain the term Document Type Definition (DTD).

Ans. :

DTD



XML Document Type Declaration

which is also known as DTD is a way which helps to describe specifically the XML language. DTDs examine the validity of structure and vocabulary of an XML document against the grammatical rules and regulations of the suitable XML language. An application can take help of a DTD to check that XML data is valid. Generally an XML document is defined as : (1) Well-formed : An XML document is considered as well-formed if it follows all the XML rules like tags must be correctly nested, open-close tags are balanced, and empty tags ends with '/>' OR (2) Valid : Valid means, any XML document is not just only well-formed but also ensures availability of DTD that indicates which tags it uses, attributes of those tags, and which tags can be nested inside other tags, among other properties.

In the Fig. 6.1 we can observe that a DTD is used to structure the XML document :

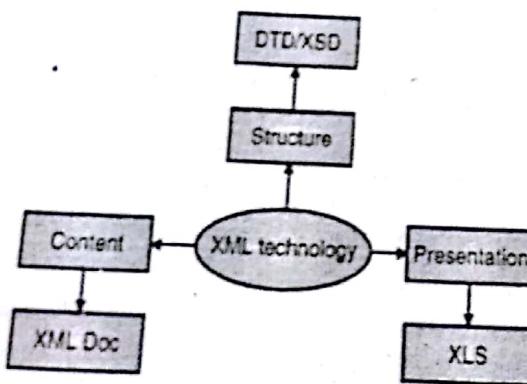


Fig. 6.1 : DTD structure

Q. 6 Explain the attributes in XML.

Ans. :

Dec. 16

Attributes in XML

Attribute provides some more data about an element or more precisely it defines a property of an element. In XML the attribute is in the form of a name-value pair. There may be multiple unique attributes to an element.

Syntax

Basic syntax of DTD attributes declaration is as follows :

<!ATTLIST element-name attribute-name attribute-type attribute-value>

In syntax, the DTD attributes start with `<!ATTLIST` keyword if the element contains the attribute.

1. **element-name** specifies the name of the element to which the attribute applies.
2. **attribute-name** specifies the name of the attribute which is included with the element-name.
3. **attribute-type** defines the type of attributes.
4. **attribute-value** takes a fixed value that the attributes must define.

Example

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

<!DOCTYPE Syntax

The syntax of DTD attributes declaration:

```
<!ATTLIST element-name attribute-name attribute-type attribute-value>
```

In this syntax, If there is attribute in element, then the DTD attributes start with `<!ATTLIST` keyword.

1. **element-name** - name of the element to which the attribute applies.
2. **attribute-name** - name of the attribute which is included in the element-name.
3. **attribute-type** - type of attributes.
4. **attribute-value** - fixed value

Example

```
<?xml version = "1.0"?>
<!DOCTYPE Details [
  <!ELEMENT Details (emp_name)>
  <!ELEMENT emp_name (#PCDATA)>
  <!ATTLIST emp_name id CDATA #REQUIRED>
]>
<Details>
  <emp_name id="123">Kunal</emp_name>
</Details>
```

Rules of Attribute Declaration

It is necessary to declare all the attributes in Document Type Definition (DTD) using an Attribute-List Declaration which we want to use in XML document. Attributes may appear in either start or empty tags. ATTLIST keyword should be in upper case. Duplication of attribute names is not allowed in the attribute list for a given element.

Attribute Types

When attributes are declared, we can specify how the processor will handle the data of attribute value. The attribute types are categorized in three different categories :

1. String type
2. Tokenized types
3. Enumerated types

Q. 7 Explain the entities in XML.

Dec. 16

Ans. :

Entities In XML

Entity references are placeholders for other values that are otherwise reserved in the language or that maybe misinterpreted. For example the less than (`<`) and the greater than (`>`) symbols are reserved for demarcating the tags. If the entity description itself contains one of these symbols the data would be misinterpreted. To avoid such a scenario Entities are used. The ampersand (`&`) symbol is reserved to indicate start of an entity. The various predefined entities are as follows :

<	LESS THAN
>	GREATER THAN
&	AMPERSAND
"	QUOTATIONS
&apos	APOSTROPHEE

Q. 8 What are XML Sitemaps? Name some different types of sitemaps. Explain main benefits of using XML Sitemaps.

May 15, May 17

Ans. :

XML sitemap

Sitemap is basically the map that contains information of all located resource on a website. It helps users, as well as search engines, to easily explore a website for their required object. On the other hand it also helps them to improve their search (for user) and indexing experience (for search engines). XML Sitemaps can have two types. Index Sitemap (how many URL sitemaps a website have), URL Sitemap (contain final information of URLs on webpage).

XML sitemaps are further divided into 3 different categories.

- (1) Sitemaps for webpages (commonly known as xml sitemap in community)
- (2) Image sitemaps (details of images and their URLs on website)
- (3) Video sitemaps (what webpages have videos embedded in them and their details)

So, final tree of sitemaps categorization we have is;

XML Sitemap

- (1) Index Sitemap
- (2) URL Sitemap
 - (i) Sitemaps for Webpages
 - (ii) Sitemaps for Images
 - (iii) Sitemaps for videos

XML is basically a language that store information about an object in organized or pre-defined format. This format is not understandable by humans but search engines do understand what is written in it and the purpose of writing. So, we can say that; XML sitemaps are primarily for search engines to have a map of internal/external resources of website with their information. Fast and secure indexing of website by search engines rely on these sitemaps. In XML sitemaps, we write information about objects like its location of availability, importance, type of object, how frequently it gets changed etc.

Q. 9 What you understand by web services? Make a general diagram for any application involving web services. It should show various technologies/tools used for creating such application.

Dec. 16, May 17, Dec. 17

Ans. :

Web services

Now days there are various programming platforms available to develop web-based applications. Some applications are developed in Java, some in .Net while some are in Node.js, AngularJS etc. Sometimes these heterogeneous applications required to communicate with each other. But as these applications are developed in different platforms, it becomes difficult to set communication between them. In such situation, the **Web Services** comes in picture. Web Services offers a common platform which helps the various applications built on different platforms to communicate with each other.

A Web Service is a software component which is developed to perform specific tasks regarding communication between two platforms. It is possible to search the Web Services over the network or internet and can be called accordingly. In the Fig. 6.2 we can observe that the Java Application can communicate with .Net Application through Web Service.

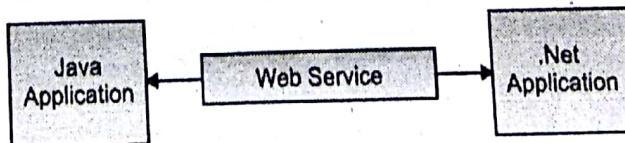


Fig. 6.2 : Web Service

A Web Service is any software component which is available on internet. The standardized XML messaging system is used by the Web Service for communication purpose. All communications are encoded through XML.

Example

An XML message is sent by the client to invoke the Web Service and also waits for XML response. The Web Service is not dependent on any operating system since the communication is done in XML. Application in any platform can interact with any other application of any platform. Web Services are modular, dynamic, distributed and self contained software components which can be easily published, located and also called over the network or internet. Web Services are built on the standard languages like HTML, Java, XML and with standard protocols like TCP/IP and HTTP.

Distributed systems

Distributed systems have been a part of computer science for decades. They are systems where different components in a network, communicate with each other and coordinate their actions only by passing messages. A component may be a program execution on a computer or a device such as a computer or a printer. It is a rather simple definition, but it covers the entire range of systems that can be called distributed systems. Both the general characteristics and some of the specific characteristics of distributed systems are explored.

Web services

Web services technology is a relatively new development. It is based upon the principles of distributed systems. A web service is a set of functions that are published to a network for use by other programs. Many people regard web services as a technology only for publishing software services on the Internet via browsers, while others regard them as the "new big thing" in distributed computing that is working as general purpose architectures. Both the general characteristics and some of the specific characteristics of web services are explored.

Evaluation

Web services contain many features which satisfy the goals of distributed systems. And in many cases they will probably be the appropriate way of designing a distributed system.

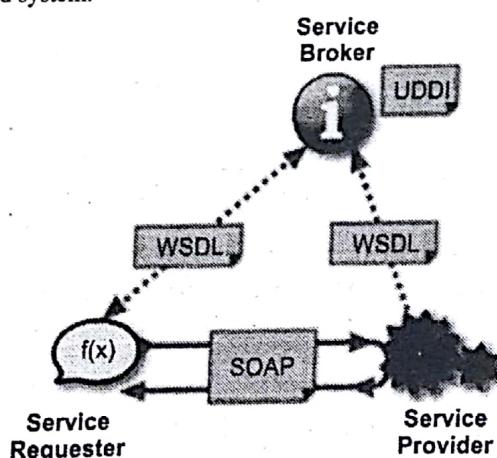


Fig. 6.3 : Web Service

However there are limits to when web services can be used as foundation for distributed systems. It is probably better to use a distributed solution internally in an organisation or system and to use a web service solution in smaller and less trivial solutions, especially if the web service is made by an external party.

Q. 10 Describe the significance and working of WSDL with an example.

May 15, Dec. 16

Ans. :

WSDL

WSDL was developed jointly by Microsoft and IBM. WSDL stands for Web Services Description Language. WSDL is the standard format for describing a Web Service. WSDL is an XML based document which contains the information regarding Web Services like names of methods, method parameters and how to access the methods. WSDL also contains information regarding which operation the Web Service will perform. WSDL is used for information exchange in decentralized and distributed environments. WSDL is a part of UDDI, an XML-based worldwide business registry. It works as an interface between various Web Service applications. WSDL is pronounced as wiz-dull. The WSDL describes services as collections of network endpoints, or ports.

The WSDL specification provides an XML format for documents for this purpose. The abstract definitions of ports and messages are separated from their concrete use or instance, allowing the reuse of these definitions. A port is defined by associating a network address with a reusable binding, and a collection of ports defines a service. Messages are abstract descriptions of the data being exchanged, and port types are abstract collections of supported operations. The concrete protocol and data format specifications for a particular port type constitutes a reusable binding, where the operations and messages are then bound to a concrete network protocol and message format. In this way, WSDL describes the public interface to the Web service. WSDL is often used in combination with SOAP and an XML-Schema to provide Web services over the Internet. A client program connecting to a Web service can read the WSDL file to determine what operations are available on the server.

Any special data types used are embedded in the WSDL file in the form of XML Schema. The client can then use SOAP to actually call one of the operations listed in the WSDL file using for example XML over HTTP. The current version of the specification is 2.0; version 1.1 has not been approved by the W3C but version 2.0 is a W3C recommendation.

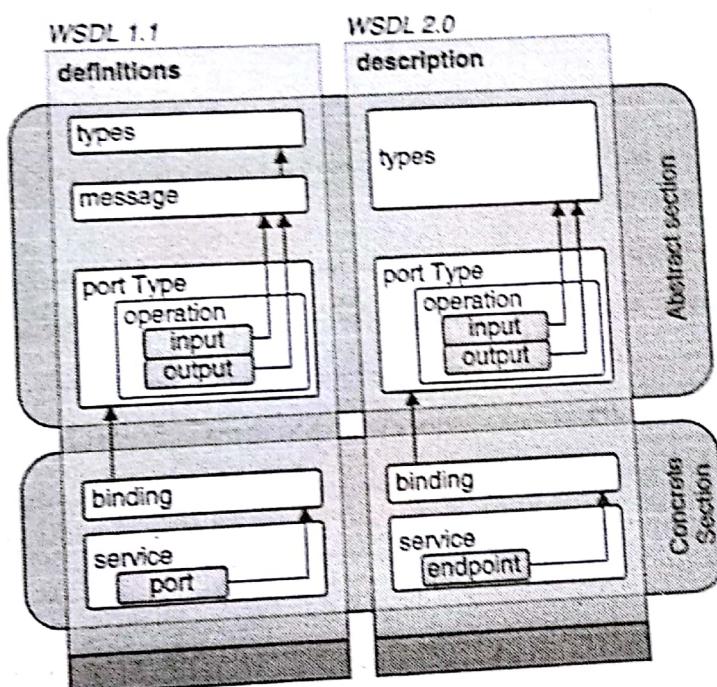


Fig. 6.4 : Representation of concepts defined by WSDL 1.1 and WSDL 2.0 documents

Q.11 Write short note on : REST.

Ans. :

REST

The acronym of REST is Representational State Transfer. REST is a style of architecture. It is not any set of standard rules. The REST architecture is followed by number of applications which are known as RESTful. While SOAP targets the actions, the REST concentrates on resources. The resources are located by the REST with the help of URL and it totally depends on the type transport protocol for the actions to be performed on the resources.

Example

In a RESTful architecture, this URL `http://[serverAddress]/employee/id/101` can be used to retrieve employee information, we send REST call of GET type, and the service will sends back the information of employee with id 101. The same service can also be used to make changes in the employee data, by sending the new updated values as form data in a PUT request.

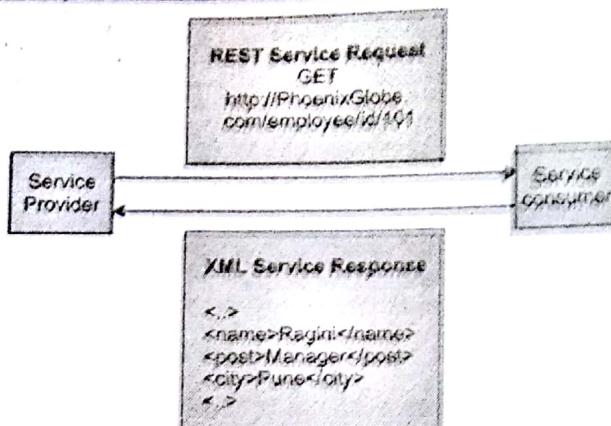


Fig. 6.5

Q. 12 What are characteristics of REST WSs?

Dec 16 Dec 17

Ans. :**Characteristics of REST Ws**

- Fast :** As RESTful Web Services do not have any strict specification like SOAP, they are comparatively fast. Also the bandwidth and resources consumed by the RESTful Web Services are very less.
- Language and Platform independent :** It is possible to write the RESTful Web Services in any programming language and also they can be executed on any platform.
- Can use SOAP :** SOAP can be used by the RESTful Web Services as the implementation.
- Permits different data format :** In RESTful Web Service, data of different formats like HTML, XML, JSON and even plain text can be used.

Chapter 7 : Python Web Framework : Django**Q. 1 Explain the types of framework architectures.****Ans. :**

Types of framework architecture

1. Model-view-controller (MVC)

Many frameworks follow the MVC architectural pattern to separate the data model with business rules from the user interface. This is generally considered a good practice as it modularizes code, promotes code reuse, and allows multiple interfaces to be applied. In web applications, this permits different views to be presented, such as web pages for humans, and web service interfaces for remote applications.

2. Push-based vs. pull-based

Most MVC frameworks follow a push-based architecture also called "action-based". These frameworks use actions that do the required processing, and then "push" the data to the view layer to render the results. Django, Ruby on Rails, Symfony, Spring MVC, Stripes, Diamond, CodeIgniter are good examples of this architecture. An alternative to this is pull-based architecture, sometimes also called "component-based". These frameworks start with the view layer, which can then "pull" results from multiple controllers as needed. In this architecture, multiple controllers can be involved with a single view. Lift, Tapestry, Moos Seam, JavaServer Faces, (μ)Micro, and Wicket are examples of pull-based architectures.

3. Three-tier organization

In three-tier organization, applications are structured around three physical tiers: client, application, and database. The database is normally an RDBMS. The application contains the business logic, running on a server and communicates with the client using HTTP. The client on web applications is a web browser that runs HTML generated by the application layer.

Q. 2 Discuss the features of Web Frameworks.

Ans. :

Features of web framework

1. Frameworks typically set the control flow of a program and allow the user of the framework to "hook into" that flow by exposing various events.
2. **Web template system** : Frameworks provide web template system which is used in web publishing to allow web designers and developers to work with web templates for the automatic generation of custom web pages, such as the results from a search.
3. **Caching** : Frameworks offer Web caching which is the caching of web documents in order to reduce bandwidth usage, server load, and perceived "lag".
4. **Security** : Some web frameworks come with authentication and authorization frameworks, that enable the web server to identify the users of the application, and restrict access to functions based on some defined criteria. Drupal is one example that provides role-based access to pages, and provides a web-based interface for creating users and assigning them roles.
5. **Database access, mapping and configuration** : Many web frameworks create a unified API to a database backend, enabling web applications to work with a variety of databases with no code changes, and allowing programmers to work with higher-level concepts. Additionally, some object-oriented frameworks contain mapping tools to provide object-relational mapping, which maps objects to tuples.
6. **Scaffolding** : Scaffolding is a technique supported by some model-view-controller frameworks, in which the programmer can specify how the application database may be used. The compiler or framework uses this specification, together with pre-defined code templates, to generate the final code that the application can use to create, read, update and delete database entries, effectively treating the templates as a "scaffold" on which to build a more powerful application.
7. **URL mapping** : A framework's URL mapping or routing facility is the mechanism by which the framework interprets URLs. Some frameworks, such as Drupal and Django, match the provided URL against pre-determined patterns using regular expressions, while some others use rewriting techniques to translate the provided URL into one that the underlying engine will recognize.
8. **Web services** : Some frameworks provide tools for creating and providing web services. These utilities may offer similar tools as the rest of the web application.
9. **Web resources** : A number of newer Web 2.0 RESTful frameworks are now providing resource-oriented architecture (ROA) infrastructure for building collections of resources.

Q. 3 What are the features of Django?

Ans. :

Features of Django.

i. Loosely Coupled

The basic aim of Django is to make all the elements in the stack independent to each other.

ii. Less Coding

Facilitates less code which leads to quick development,

iii. Don't Repeat Yourself (DRY)

All the elements are developed only in exactly single place rather than repeating again and again.

iv. Fast Development

The development in Django is effectively fast.

v. Clean Design

A strictly clean design is maintained by Django all over the code and makes it simple to follow best web-development practices.

Q. 4 Explain advantages of Django.

Ans. :

Advantages of Django

1. Object-Relational Mapping (ORM) Support

A bridge is provided by Django between the data model and the database engine, and also gives a strong support to a large group of DBMSs such as MySQL, Oracle, Postgres, etc.

2. Multilingual Support

Multilingual websites are supported by Django with the help of its inbuilt internationalization system. This helps user to develop website which can support multiple languages.

3. Framework Support

There is built-in support in Django for Ajax, RSS, Caching and several other frameworks.

4. Administration GUI

An efficient ready-to-use user interface is provided by Django for administrative activities.

5. Development Environment

A lightweight web server is provided by Django to support end-to-end application development as well as testing.

