Web Technology Assignment

Name: Ishita Chalise
NCE077BCT012

2080 Baishakh (Back)

Q.2 Explain CSS (Cascading Style Sheet). How external style sheets can be inserted in a HTML document. Illustrate with example. (2+3)

Ans: CSS (Cascading Style Sheets) is a stylesheet language used to describe the visual presentation of a document written in HTML or XML. It controls the layout, colours, fonts, and overall appearance of web pages, allowing for separation of content and design.

Key Features:

1. Separation of Content and Presentation: Keeps HTML content separate from design.
2. Cascading: Applies styles from multiple sources with a specific precedence.
3. Selectors and Properties: Targets HTML elements using selectors and styles them using properties.

External style sheets allow you to apply the same styles to multiple web pages. This is done by linking an external CSS file to the HTML document using the **<link>** element within the **<head>** section.

Steps to Link an External CSS File:

1. Create a CSS File: Write your CSS styles in a file with a **.css** extension (e.g., styles.css).
2. Link the CSS File to HTML: Use the **<link>** element to connect the CSS file to your HTML document.

For example,

HTML:

```
<link rel="stylesheet" type="text/css" href="style.css">
```

CSS (style.css):

```
p {color: blue; }
```

Q. 3. Write is Document Object Model (DOM)? Explain rowspan and colspan attributes in html tables with example. (2+3)

Ans:

DOM is a representation of the document in an object form, accessible from JavaScript programs.

Document Object Model (DOM) is a programming interface for web documents. It represents the structure of a web page as a tree of objects, where each object corresponds to a part of the page (like elements, attributes, and text). This allows scripts and programs to manipulate the content and structure of web pages dynamically.

In HTML tables, the rowspan and colspan attributes are used to merge cells across rows and columns, respectively.

- **rowspan**: This attribute specifies the number of rows a cell should span vertically.
- **colspan**: This attribute specifies the number of columns a cell should span horizontally.

For example,

```
<tr>
    <td rowspan="2">Row 1, Cell 1</td>
    <td colspan="2">Row 1, Cell 2</td>
</tr>
```

rowspan="2": The first cell will span two rows vertically.
colspan="2": The second cell will span two columns horizontally.

Q.6 What are cookies? Discuss the importance of cookies in JavaScript using suitable example. (2+3)

Ans:

Cookies are small pieces of data stored by a web browser that retain user-specific information. Websites use cookies to remember stateful information, such as login status, preferences, and other data, to enhance user experience.

Importance of Cookies in JavaScript

Cookies play a vital role in maintaining the state and providing a seamless user experience in web applications. Here are some key points:

1. **State Management**: Cookies help in maintaining user sessions across different pages.
2. **Personalization**: They store user preferences to personalize content and interface settings.
3. **Tracking**: Cookies are used for tracking user behavior, helping in analytics and targeted advertising.
4. **Authentication**: They maintain login sessions, allowing users to stay logged in across multiple sessions.

Q.8 What is Web2.0? Highlight the differences between Web1.0 and Web2.0? (2+3)

Ans:

Web 2.0 represents the evolution of the web from static web pages to a more interactive and user-driven experience. It focuses on user participation, dynamic content, and collaboration. Web 2.0 enhances the web experience by making it more interactive and collaborative, with a focus on user-generated content and real-time updates.

Feature	Web 1.0	Web 2.0
Content	Static, read-only	Dynamic, user-generated
User Interaction	Limited interaction (view-only)	Interactive (contribute and engage)
Data Handling	Simple data presentation	Asynchronous updates (AJAX)
Technology	Basic HTML and static pages	Richer interfaces (AJAX, CSS, JavaScript)
Content Creation	Created by webmasters and developers	Created by users (blogs, wikis)
Communication	Unidirectional (website to user)	Bidirectional (user to user)
Collaboration	Minimal collaboration	Collaborative tools (wikis, social networks)
Updates	Full page reloads	Partial page updates (AJAX)
Design	Basic, fixed layout	Flexible, responsive design
Examples	Simple informational websites	Social media, interactive platforms

Q.10 What is service oriented architecture? Explain SOAP. (2+3)

Ans:

Service-Oriented Architecture (SOA) is an architectural approach in which applications make use of services available in the network. In this architecture, services are provided to form applications, through a communication call over the internet.

- SOA allows users to combine many facilities from existing services to form applications.

- SOA encompasses a set of design principles that structure system development and provide means for integrating components into a coherent and decentralized system.
- SOA based computing packages functionalities into a set of interoperable services, which can be integrated into different software systems belonging to separate business domains

SOAP

SOAP, or Simple Object Access Protocol, is a communication protocol used to exchange information between applications. It uses XML to format messages, making it platform-independent and able to work across different systems and technologies. As a W3C recommendation, SOAP ensures standardized communication. It was designed to allow web applications to communicate over the Internet using HTTP, which is supported by all web browsers and servers. This makes SOAP useful for enabling communication between applications running on different operating systems and using various technologies and programming languages.

Below is a sample skeleton of the SOAP XML file

```
<?xml version="1.0"?>
<soap:Envelope
<soap:Header>
</soap:Header>
<soap:Body>
</soap:Body>
</soap:Envelope>
```

Q.11 What is REST? Mention the HTTP methods supported by REST. (2+3)

Ans: Representational State Transfer (REST), or RESTful web services, enable different computer systems on the Internet to work together. RESTful services allow systems to access and modify web resources using a standard set of operations that don't retain session information between calls. Unlike web services that use WSDL and SOAP, which define their own sets of operations, REST uses a simpler, standardized approach.

Key Points:

- **Web Resources**: Originally defined as documents or files identified by URLs on the web.
- Requests and Responses: In RESTful services, when you request a resource via its URI, the server responds with data that could be in XML, HTML, JSON, or other formats. This response might confirm changes to the resource and include links to related resources.
- **Common HTTP Methods:** REST commonly uses HTTP methods like GET (retrieve data), POST (create data), PUT (update data), and DELETE (remove data).
- **Stateless and Standard Operations:** REST's stateless nature and use of standard operations lead to high performance, reliability, and scalability. Components can be reused and updated independently, without disrupting the entire system.

In simpler terms, REST makes it easy for different systems to talk to each other over the Internet, using a common language and standard procedures to manage web resources efficiently.

HTTP Methods Supported by REST

1. **GET**:
    - Used to retrieve information from the server.
    - Example: Fetching a list of users.
2. **POST**:
    - Used to create a new resource on the server.
    - Example: Adding a new user.
3. **PUT**:
    - Used to update an existing resource on the server.

    - Example: Updating user information.
4. **DELETE**:
    - Used to delete a resource from the server.
    - Example: Deleting a user.
5. **PATCH**:
    - Used to apply partial modifications to a resource.
    - Example: Updating a single field of user information.
6. **HEAD**:
    - Similar to GET but used to retrieve only the headers and not the body of the response.
    - Example: Checking if a resource is available.
7. **OPTIONS**:
    - Used to describe the communication options for the target resource.
    - Example: Determining the allowed methods on a resource.
8. **TRACE**:
    - Used to perform a message loop-back test along the path to the target resource.
    - Example: Debugging and diagnostics.

Q.13 Write short notes on: a) AJAX (5)

Ans:

AJAX (Asynchronous JavaScript and XML) is a technique for creating interactive web applications. It allows web pages to fetch and send data to the server asynchronously, meaning updates can happen in the background without reloading the entire page.

Key Points

- Asynchronous Requests: Updates web content without refreshing the whole page.
- XMLHttpRequest: The core object used to make HTTP requests and handle responses.
- Data Formats: Commonly uses JSON for data exchange, but XML and other formats are also supported.
- User Experience: Enhances speed and responsiveness, enabling features like live search and dynamic content updates.

AJAX improves web application performance and user experience by enabling smooth, asynchronous updates.

For example,

2079 Bhadra (Regular)

Q.6 Explain the working principle of AJAX request and response. Explain the term wiki and blogs. (4+4)

Ans: AJAX (Asynchronous JavaScript and XML) allows for asynchronous communication between a client and server, meaning web pages can update dynamically without a full page reload. Here's how AJAX request and response work:

1. **Trigger Action**: An action on the webpage (like a button click) starts the AJAX request.
2. **Create Request**: JavaScript creates a request to send to the server using XMLHttpRequest or the Fetch API.
3. **Send Request**: The request is sent to the server without refreshing the webpage.
4. **Server Response**: The server processes the request and sends back data.
5. **Receive Data**: JavaScript receives the server's response and processes it.
6. **Update Page**: The webpage is updated with the new data without reloading.

Wiki

- **Definition**: A wiki is a collaborative platform where users can create, edit, and organize content together. It's used for shared documentation and knowledge.
- **Key Features**:
  - **Hyperlinked Text**: Pages link to each other for easy navigation.
  - **Version Control**: Tracks changes and edits.
  - **User Collaboration**: Multiple users can edit content simultaneously.
- **Example**: Wikipedia, where people globally contribute to an online encyclopedia.

Blog

- **Definition**: A blog is a regularly updated web page or site, often by an individual or small group, sharing personal opinions, experiences, or news.
- **Key Features**:
  - **Posts**: Articles displayed in reverse chronological order.
  - **Comments**: Readers can interact and discuss posts.
  - **Categories and Tags**: Helps organize and find posts.
- **Example**: A personal blog with posts about travel, book reviews, or recipes.

Q.1. What is Internet and Web? Differentiate Web1.0, Web2.0 and Web3.0. (3+5)

Ans:

The Internet is a global network of interconnected computers and other devices that communicate using standard protocols like TCP/IP. It's the physical infrastructure that allows various types of data and communication to take place.

The Web is a service that operates over the Internet, consisting of a system of interlinked hypertext documents and multimedia content. It's often referred to as "WWW" or simply "the Web."

Differentiate between Web1.0, Web 2.0 and Web3.0

Aspect	Web 1.0	Web 2.0	Web 3.0
Content	Static, read-only pages	Dynamic, user-generated content	Semantic, AI-enhanced data integration
User Interaction	Minimal, primarily browsing	Interactive, users can comment and create content	High, with AI-driven interactions and automation
Technology	Basic HTML and CSS	AJAX, JavaScript, APIs, social media platforms	Blockchain, Semantic Web, AI, decentralized technologies
Data Ownership	Centralized, controlled by website owners	Centralized, controlled by large platforms	Decentralized, controlled by users
Personalization	Limited, uniform content	High, personalized recommendations	Advanced, deeply personalized experiences
Examples	Simple static websites	Social media sites like Facebook, Twitter	Decentralized apps (dApps), blockchain platforms

Q. 2 What is CSS? Explain its types with example. (2+6)

Ans: CSS (Cascading Style Sheets) is a stylesheet language used to describe the presentation of a document written in HTML or XML. It defines how elements should be displayed, controlling layout, colors, fonts, and overall visual appearance.

Types of CSS

1. Inline CSS

Inline CSS is used to apply styles directly to an HTML element using the style attribute. This method is useful for quick, single-use styles but can be cumbersome for larger projects due to poor maintainability.

For example

```
<p style="color: red;">
    This is a red paragraph.
</p>
```

2. Internal (Embedded) CSS:

Internal CSS is defined within the <style> tag inside the <head> section of an HTML document. It is useful for applying styles to a single document.

For example

```
<head>
  <style>
      p{ color: blue ;}
  </style>
</head>
```

3. **External CSS:**

External CSS is written in a separate .css file and linked to the HTML document using the <link> element. This method is ideal for maintaining styles across multiple web pages.

For example

HTML:

```
<link rel="stylesheet" type="text/css" href="style.css">
```

CSS (style.css):

```
p {
    color: blue;
}
```

Q.3. Design a webpage which contains a form where user can input name, gender, address, email and submit button to submit the form" Also validate the form using JavaScript. (8)

Ans:

```
<!DOCTYPE html>
<html>
<head>
  <title>Simple Form</title>
  <script>
    function validateForm()
   {
      var name = document.forms["userForm"]["name"].value;
      var gender = document.forms["userForm"]["gender"].value;
      var address = document.forms["userForm"]["address"].value;
      var email = document.forms["userForm"]["email"].value;

      if (name === "" || gender === "" || address === "" || email === "")
     {
        alert("All fields must be filled out.");
        return false;
      }
```

```
            alert("Form submitted successfully!");
            return true; // Submit the form
        }
    </script>
</head>
<body>
    <h2>User Information Form</h2>
    <form name="userForm" onsubmit="return validateForm()">
        <label for="name">Name:</label><br>
        <input type="text" id="name" name="name"><br><br>

        <label for="gender">Gender:</label><br>
        <select id="gender" name="gender">
            <option value="">Select...</option>
            <option value="male">Male</option>
            <option value="female">Female</option>
        </select><br><br>
        <label for="address">Address:</label><br>
        <textarea id="address" name="address"></textarea><br><br>
        <label for="email">Email:</label><br>
        <input type="email" id="email" name="email"><br><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

Q.8 What is service oriented architecture? Compare SOAP and REST. (4+4)

Ans:

Service-Oriented Architecture (SOA) is an architectural design pattern where software components, called services, are provided over a network and can interact with each other. These services are independent, reusable, and loosely coupled, allowing different systems to communicate and share data efficiently. SOA promotes flexibility, scalability, and easier integration of diverse applications.

Aspect	SOAP	REST
Protocol	Uses HTTP, SMTP, and other protocols.	Primarily uses HTTP.
Message Format	Typically XML.	Can use XML, JSON, or other formats.
Complexity	Generally more complex with strict standards.	Simpler and more flexible.
Statefulness	Can be stateful or stateless.	Usually stateless.
Security	Built-in security (WS-Security).	Relies on underlying protocol security (e.g., HTTPS).
Performance	Can be slower due to XML processing.	Typically faster due to lightweight formats (e.g., JSON).

Q. 10 Write short notes on: **(4)**

a) Crawling

Ans: Crawling refers to the process by which automated programs, known as web crawlers or spiders, systematically browse and index web pages. Crawlers follow links from known web pages to discover and collect information about new pages, which is then used by search engines to organize and rank web content, ultimately improving the relevance and accuracy of search results.

Crawling involves automated programs called "web crawlers" or "spiders" that browse the web systematically.

How It Works:

The crawler starts with a list of known web pages (called "seeds").

It visits these pages and follows links to other pages.

It repeats this process to explore as many pages as possible.

Purpose: Crawling helps search engines build a database of web pages, which they use to deliver relevant search results to users.

Example: When we search for something on Google, crawlers have already indexed many pages to provide quick and accurate search results.

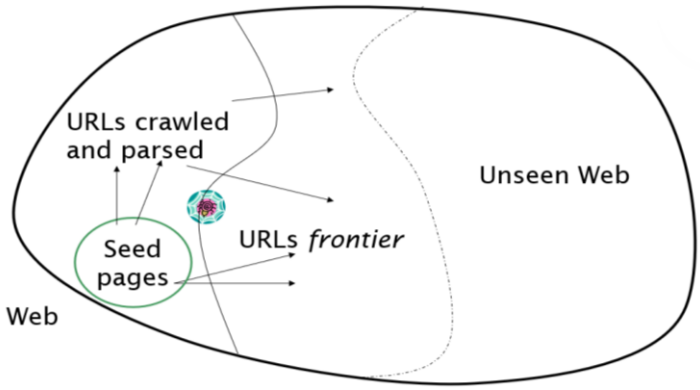


Fig: Crawling

b. Tagging and tolksonomies **(4)**

Ans:

Tagging:

- **Definition**: In web technology, tagging involves assigning descriptive keywords to online content (like blog posts, videos, or images) to categorize and organize it.
- **Purpose**: Tags help users and search engines understand and find content more easily.
- **Usage**: Commonly used in blogs, social media, and websites for better search and navigation.
- **Example**: On a blogging platform, an article about healthy recipes might be tagged with "healthy eating," "recipes," and "nutrition."

Folksonomies:

- **Definition**: Folksonomies are collaborative classification systems where users create and manage tags to organize content.
- **Function**: They emerge from collective tagging by a community of users, rather than being imposed by a single authority.
- **Purpose**: Enables users to find and organize information based on collective tagging.
- **Benefits**:
  - **Flexibility**: Users can create and use tags that make sense to them.
  - **Collaboration**: Multiple users contribute to a more comprehensive tagging system.
- **Example**: Websites like YouTube and Flickr use folksonomies as users add tags to videos and photos, respectively, making it easier for others to discover related content.

Tagging and folksonomies play a crucial role in organizing and discovering content. They enhance user experience by making information more accessible and searchable, though they also come with challenges like inconsistency and potential for misuse. Despite these challenges, their flexibility and user-driven nature make them essential tools in the digital landscape.

d. JavaScript DOM (4)

Ans:

The JavaScript Document Object Model (DOM) is a programming interface for web documents. It represents the structure of a web page as a tree of objects, allowing developers to access and manipulate the content, structure, and styles of a website dynamically.

- Structure: The DOM represents HTML and XML documents as a tree structure of nodes.
- Nodes: Everything in a web page is a node (e.g., elements, attributes, text).
- Manipulation: Developers can use JavaScript to change content, style, and structure without reloading the page.
- Event Handling: The DOM allows adding and managing events (like clicks, inputs) to elements.
- Methods: Common methods include getElementById, getElementsByClassName, querySelector, createElement, appendChild, and removeChild.

For example,

```
<!DOCTYPE html>
<html>
<head>
  <title>DOM Example</title>
</head>
<body>
  <p id="demo">Hello, World!</p>
  <button onclick="changeText()">Change Text</button>
  <script>
    function changeText() {
      document.getElementById("demo").innerText = "Hello, JavaScript!";
    }
  </script>
</body>
</html>
```

2078 Bhadra (Regular)

Q.2 Design HTML User Registration form as below. Validate the registration from using JavaScript. (10)

Ans:

Q.10 Short note on (3)
a. Crawling. **c. DOM** **b. AJAX** (Ans :2080 Baishakh)
Ans: See 2079 Bhadra Q.10

Q.4 Explain Dialogue Boxes in JavaScript? (6)
Ans:

JavaScript provides three types of dialogue boxes: alert, confirm, and prompt. These boxes are used to interact with the user and handle simple input and output operations.

1. Alert Box

The alert box is used to display a simple message to the user. It only has an "OK" button.

alert("This is an alert box!");

Usage: To show warnings or information.

For Example,

```
<!DOCTYPE html>
<html>
<head>
   <title>Alert Example</title>
</head>
<body>
   <button onclick="showAlert()">Show Alert</button>
   <script>
      function showAlert() {
         alert("This is an alert box!"); }
   </script>
</body>
</html>
```

2. Confirm Box

The confirm box is used to display a message and ask for user confirmation. It has "OK" and "Cancel" buttons.

```
var result = confirm("Do you want to proceed?");
if (result) {
   // User clicked "OK"
} else {
   // User clicked "Cancel"
}
```

Usage: To confirm user actions.

For Example,

```
<!DOCTYPE html>
<html>
<head>
   <title>Confirm Example</title>
</head>
<body>
   <button onclick="showConfirm()">Show Confirm</button>
   <script>
      function showConfirm() {
         var result = confirm("Do you want to proceed?");
         if (result) {
```

```
            alert("You clicked OK!");
        } else {
            alert("You clicked Cancel!");
        }}
    </script>
</body>
</html>
```

3. Prompt Box

The prompt box is used to get input from the user. It has a text input field along with "OK" and "Cancel" buttons.

```
var userInput = prompt("Please enter your name:", "Default Name");
if (userInput !== null) {
    // User entered something and clicked "OK"
} else {
    // User clicked "Cancel"
}
```

Usage: To get user input.

For Example,

```
<!DOCTYPE html>
<html>
<head>
    <title>Prompt Example</title>
</head>
<body>
    <button onclick="showPrompt()">Show Prompt</button>
    <script>
        function showPrompt() {
            var userInput = prompt("Please enter your name:", "Default Name");
            if (userInput !== null) {
                alert("Hello, " + userInput + "!");
            } else {
                alert("You clicked Cancel!");
            }}
    </script>
</body>
</html>
```

Q.6 Differentiate Web 2.0 and Web 3.0? Why Tagging-folksonomies are important? (4+4)

Ans: Importance of Tagging-Folksonomies

Tagging-folksonomies are important because they allow users to collaboratively categorize and organize content through tags, making it easier to search and discover information. Here are key points:

1. Improved Discoverability: Tags help users find related content quickly by grouping similar items together.
2. User-Generated Classification: Unlike traditional taxonomies, folksonomies are created by users, making them more adaptable and reflective of current trends and user language.
3. Enhanced Searchability: Tags improve search engine optimization (SEO) by providing additional metadata that search engines can use to index content more effectively.

4. Community Engagement: Tagging encourages user participation and interaction, fostering a sense of community and collaboration.
5. Personalization: Tags can be used to tailor content recommendations based on user interests and behaviour.

Differences Between Web 2.0 and Web 3.0

Feature	Web 2.0	Web 3.0
Content	User-generated content	Intelligent, context-aware content
Interaction	Social interaction and collaboration	Personalized and semantic interaction
Data	Shared through social media, blogs	Linked data and knowledge graphs
Technology	AJAX, JavaScript, social platforms	AI, machine learning, blockchain
Web Structure	Centralized, platform-based	Decentralized, blockchain-based
Search	Keyword-based search	Contextual and semantic search
User Experience	Rich user interfaces, dynamic content	Personalized and intelligent experiences
Examples	Facebook, Wikipedia, YouTube	Siri, Wolfram Alpha, Decentralized apps (dApps)

Q.8 Explain Service Oriented Architecture (SOA). Differentiate between SOAP and REST. (6+4)
Ans:
Service Oriented Architecture (SOA)
Definition: Service Oriented Architecture (SOA) is a design approach where different services communicate with each other to perform business functions. Each service is a discrete unit of functionality that can be independently deployed, managed, and combined to form more complex applications.

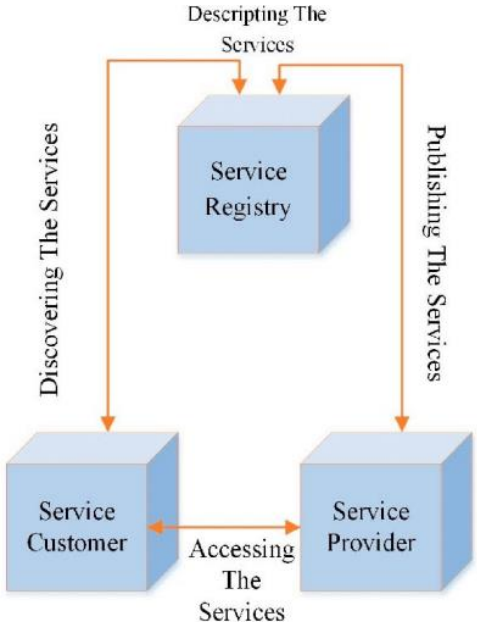


Fig: Block diagram of SOA

Key Features:
- **Modularity**: Services are independent and can be reused across different applications.
- **Interoperability**: Services can communicate using standard protocols, making them compatible across different platforms and languages.
- **Scalability**: Services can be scaled independently based on demand.

- **Loose Coupling**: Services interact with each other through well-defined interfaces, reducing dependencies.

Components:

1. Service Provider:

- Role: Creates and offers web services.
- Decisions: Determines which services to provide, prioritizes security or ease of access, and sets prices.
- Listing: Chooses the category for the service in a broker's directory and establishes agreements with users.

2. Service Broker, Registry, or Repository:

- Function: Makes web service information available to potential users.
- Scope: Can be public (accessible to everyone) or private (restricted to certain users).

3. Service Requester/Consumer:

- Role: Searches for services in the broker's directory, connects to the service provider, and uses the web services.
- Process: Finds needed services through the broker, connects (binds) to the provider, and uses the service. Multiple services can be accessed if offered.
- Agreement: The relationship between consumer and provider is defined by a standard contract covering business, functional, and technical aspects.

Benefits:

- **Flexibility**: Easy to update or replace services without affecting the entire system.
- **Reusability**: Services can be reused in different applications, reducing redundancy.
- **Maintainability**: Easier to manage and maintain individual services.

A service in Service-Oriented Architecture (SOA) has four main properties. It represents a specific business activity with a clear outcome, is self-contained, functions as a black box so users don't need to know its inner workings and can include other services within it.

Differentiate between SOAP and REST

Feature	SOAP (Simple Object Access Protocol)	REST (Representational State Transfer)
Architecture	SOAP follows a strict protocol and uses XML for message format.	REST is an architectural style that uses standard HTTP methods.
Message Format	Uses XML exclusively for request and response messages.	Can use multiple formats like JSON, XML, HTML, and plain text.
Transport Protocol	Primarily uses HTTP, but can also work over SMTP, TCP, and others.	Uses HTTP/HTTPS exclusively for communication.
Complexity	More complex due to rigid standards and extensive features.	Simpler and more flexible due to its lightweight design.
Performance	Generally slower due to the overhead of XML and extensive processing.	Typically faster due to less overhead and use of simpler formats.
Statefulness	Supports stateful operations through WS-* standards.	Typically stateless, where each request is independent.
Use Cases	Suitable for enterprise-level applications requiring high security and ACID compliance.	Ideal for web services requiring scalability and flexibility, like web APIs.

Q.9 What is semantic web? Explain the purpose of semantic web with suitable example. (6)
Ans: The Semantic Web is an advanced version of the current web designed to make data understandable and usable by computers.
The Semantic Web is an extension of the World Wide Web (Web 3.0) designed to add structure to web data. Its goal is to shift from a web of documents to a web of data, enabling computers to search, combine, and process web content based on its meaning. By providing more context to websites, the Semantic Web helps both humans and machines understand and interpret content better, leading to more accurate and coherent search results.

The purpose of the Semantic Web is to enhance the web by enabling machines to understand and interpret data more meaningfully. Here's how it achieves this:

1.Structured Data:
It adds structure to web data using standardized formats, making it easier for computers to process and link information.

2. Improved Search:
By understanding the context and relationships between data, it allows for more accurate and relevant search results.

3. Data Integration:
Facilitates the seamless combination of information from different sources, improving data integration and interoperability.

4. Enhanced Automation:
Supports automated decision-making and tasks by interpreting the meaning of data.

5. Contextual Understanding:
Provides more context to web content, enabling both humans and machines to better understand and utilize the information.

The Semantic Web transforms the web from a collection of documents into a network of interlinked data that machines can intelligently process.

2076 Chaitra (Regular)

Q.11 What is Web 2.0? What are the difference between Web 2.0 and Web 1.0? (2+3)
Ans: (See 2079 Bhadra Q.1)

Q.5 Explain the working of web crawlers. (8)
Ans: Web crawlers, also known as web spiders or bots, are automated programs used to explore and index the web. The working of web crawlers:

1. **Start with Seed URLs**: Web crawlers begin with a list of starting URLs (seed URLs). These are the initial web pages that the crawler will visit.
2. **Fetch Pages**: The crawler requests the web page from the server and downloads its content.
3. **Parse the Content**: After fetching a page, the crawler reads and analyzes the content. It looks for links to other pages (URLs) within the content.
4. **Queue Links**: The crawler adds new URLs (discovered in the previous step) to a queue for future visits. This helps the crawler to explore more pages.
5. **Index the Content**: The content of the fetched pages is then indexed. This means storing the page's data in a structured way that allows for fast searching and retrieval.
6. **Avoid Redundancy**: The crawler keeps track of visited URLs to avoid re-visiting the same pages and to manage its workload efficiently.
7. **Respect Rules**: Crawlers follow rules set by websites, such as those specified in the robots.txt file, which can restrict access to certain pages or sections.

Web crawlers systematically explore and index web pages, enabling search engines and other services to find and organize web content efficiently.

Q.2 Define XML? What is DTD (Document Type Definition)? Design well-formed XML document for "Teacher", Teacher should have data such as Name, Address, Phone, Age and Email. Name should contain FirstName and LastName. (2+3+5)

Ans:

XML (eXtensible Markup Language) is a flexible text format used to store and transport structured data. It uses tags to define elements and attributes, enabling the creation of custom tags to organize data hierarchically.

DTD (Document Type Definition) is a set of rules that defines the structure and legal elements and attributes of an XML document. It specifies what elements can appear in the XML document, their order, and how they relate to each other. DTD can be used to validate the XML document to ensure it adheres to the defined structure.

Here's a well-formed XML document for a "Teacher" with the required data fields data such as Name, Address, Phone, Age and Email :

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Teacher [
  <!ELEMENT Teacher (Name, Address, Phone, Age, Email)>
  <!ELEMENT Name (FirstName, LastName)>
  <!ELEMENT FirstName (#PCDATA)>
  <!ELEMENT LastName (#PCDATA)>
  <!ELEMENT Address (#PCDATA)>
  <!ELEMENT Phone (#PCDATA)>
  <!ELEMENT Age (#PCDATA)>
  <!ELEMENT Email (#PCDATA)>
<Teacher>
  <Name>
        <FirstName>
              Ram
        </FirstName>
       <LastName>
             Baral
      </LastName>
  </Name>
  <Address>Pokhara</Address>
  <Phone>9856056098</Phone>
  <Age>35</Age>
  <Email>ram@gmail.com</Email>
</Teacher>
```

Q.4 What is Document Object Model (DOM)? Explain DOM on the basis of HTML/JavaScript. (6)

Ans: The Document Object Model (DOM) is a programming interface that represents and interacts with the structure of a web document. It provides a structured representation of a document as a tree of objects, where each object corresponds to a part of the document, such as elements, attributes, and text.

- The DOM is a representation of the structure of an HTML document as a tree of objects.

- **Purpose**: It allows JavaScript to access and manipulate HTML elements and their attributes dynamically.

How It Works:

- **HTML Document**: When a web page is loaded, the browser creates a DOM based on the HTML content.
- **Tree Structure**: The DOM represents the HTML document as a tree, where each node is an HTML element or attribute.
- **JavaScript Interaction**: JavaScript can use the DOM to change the content, structure, or style of the web page.

Example

```
<!DOCTYPE html>
<html>
<head>
  <title>Example</title>
</head>
<body>
  <h1 id="header">Hello, World!</h1>
  <button onclick="changeText()">Change Text</button>

  <script>
    function changeText() {
      document.getElementById('header').innerText = 'Text Changed!';
    }
  </script>
</body>
</html>
```

Q.8. Explain CSS. How is XHTML different form HTML (3+3)

Ans:

XHTML vs. HTML

Feature	HTML (Hypertext Markup Language)	XHTML (Extensible Hypertext Markup Language)
Syntax	More lenient with syntax rules.	Stricter syntax rules, must be well-formed XML.
Document Structure	Can be less strict with tag nesting and closing.	Requires all tags to be properly nested and closed.
Case Sensitivity	Tag names and attributes are case-insensitive.	Tag names and attributes must be in lowercase.
Error Handling	Browsers may try to correct errors and display content.	Errors can prevent the page from being displayed.
Standards Compliance	Generally follows HTML standards but with some flexibility.	Adheres to XML standards, which ensures more rigorous compliance.

CSS (Cascading Style Sheets) is a language used to control the presentation and layout of web pages. It allows you to apply styles such as colours, fonts, and spacing to HTML elements, making it easier to create visually appealing and consistent designs.

- **Purpose**: Defines how HTML elements should be displayed on screen, paper, or in other media.
- **Selectors**: Targets HTML elements to apply styles.
- **Properties**: Includes attributes like color, font-size, margin, and padding.
- **External Style Sheets**: CSS can be included in separate files, allowing for a single style sheet to be used across multiple pages.
- **Responsive Design**: Supports media queries to adjust styles based on device screen size and resolution.

Q.7 What is the difference between (5+5)

a. GET and POST request

Ans:

Feature	GET Request	POST Request
Purpose	Retrieves data from a server.	Sends data to a server to create or update resources.
Data in URL	Data is included in the URL as query parameters.	Data is included in the request body, not the URL.
Visibility	Data is visible in the URL, making it less secure.	Data is not visible in the URL, making it more secure.
Data Length	Limited by URL length restrictions (usually up to 2048 characters).	No specific length limit, allowing for larger data.
Usage	Ideal for fetching or querying data without side effects.	Ideal for submitting forms, uploading files, or modifying data.

b. Client-side scripting and Server-side scripting

Ans:

Feature	Client-Side Scripting	Server-Side Scripting
Execution Location	Runs in the user's web browser.	Runs on the web server.
Languages	Commonly uses languages like JavaScript, HTML, CSS.	Commonly uses languages like PHP, Python, Ruby, ASP.NET.
Purpose	Enhances user interaction and provides dynamic content.	Handles data processing, database interactions, and generating web pages.
Speed	Generally faster because it runs on the client's machine.	Can be slower due to server processing time.
Security	Less secure as code is visible and accessible to users.	More secure as code is not exposed to users directly.

Q.10 What is REST? Mention the HTTP method supported by REST. (2+3)

Ans: REST (Representational State Transfer) is an architectural style for designing networked applications. It relies on standard HTTP methods and is used to build scalable and stateless web services. RESTful services interact using simple HTTP requests and responses, making them easy to use and understand.

HTTP Methods Supported by REST:

1. **GET**: Retrieves data from the server. It is used to request a resource or a collection of resources.
2. **POST**: Sends data to the server to create a new resource or submit information.
3. **PUT**: Updates an existing resource with new data. It replaces the resource or creates it if it does not exist.
4. **DELETE**: Removes a resource from the server.
5. **PATCH**: Partially updates a resource, making changes to only specific fields rather than replacing the entire resource.

2074 Chaitra (Regular)

3. Define web documents and web browser? Explain major steps of information retrieval. (2+2)

Ans:

Web Documents

Web Documents are files accessible on the internet, such as HTML pages, PDFs, and images.

Web Browser

A Web Browser is software that retrieves, displays, and interacts with web documents. Examples include Google Chrome and Mozilla Firefox.

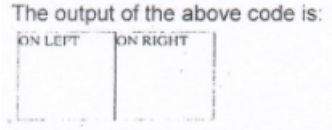
Major Steps of Information Retrieval

1. **Crawling**: Automatically collecting web pages and links.
2. **Indexing**: Organizing and storing collected data for fast retrieval.
3. **Query Processing**: Interpreting user queries to find relevant information.
4. **Ranking**: Ordering search results by relevance and quality.
5. **Retrieval and Presentation**: Displaying ranked results to the user.

Q.2 Given is the HTML code: (2+2+2)

```
<html>
<head>
</head>
<body>
 <div style="padding:10px;float:left;">
  <div id="leftbox" style="width:100px;height:100px;float:left;border: solid 1px;">
   ON LEFT
  </div>
  <div id="rightbox" style="width:100px;height:100px;float:left;border: solid 1px;">
   ON RIGHT
  </div>
 </div>
</body>
</html>
```

The output of the above code is:



Use CSS Box model (margin/padding) to do the following:

a. **Separate left and right box by total of 10 px.**
b. **Move text "ON LEFT" by 10 px towards right.**
c. **Move text "ON RIGHT" by 10 px towards right.**

Ans:

```
<html>
<head>
</head>
<body>
  <div style="padding:10px;float:left;">
    <div id="leftBox" style="width:100px;
          height:100px;float:left;
          border: solid 1px; padding-left: 10px;
          margin-right: 10px;">
      ON LEFT
    </div>
    <div id="rightBox" style="width:100px;
          height:100px;float:left;
          border: solid 1px;
          padding-left: 10px;">
      ON RIGHT
    </div>
  </div>
</body>
</html>
```

Q.4 Explain JavaScript DOM model. Make use of JavaScript function to display the date and time in either HTML document or console (2+3)

Ans: The JavaScript DOM (Document Object Model) represents the structure of an HTML document as a tree of objects, allowing JavaScript to interact with and manipulate the content and structure of web pages.

To display the date and time in either HTML document

```
<!DOCTYPE html>
<html>
<head>
   <title>Date and Time</title>
</head>
<body>
   <h1>Current Date and Time</h1>
   <div id="datetime"></div>
   <script>
      document.getElementById('datetime').innerText = new Date().toLocaleString();
   </script>
</body>
</html>
```

Q.6 What is the concept of web 2.0? Explain web 2.0 tools such as Blogs, Wiki and Social networking. (8)

Ans:

Web 2.0 marks a shift from the old, static web to a more interactive and user-driven experience. Unlike the previous version, where pages were mostly read-only and provided limited interaction, Web 2.0 emphasizes user participation and collaboration. It allows people to actively engage with websites and each other, creating and sharing their own content. This new

era supports dynamic, user-generated content and fosters community-building tools. Enhanced technologies, like AJAX, provide smoother, more responsive web experiences, making the web more engaging and collaborative.

Web 2.0 Tools

1. **Blogs:**
   Online journals or informational websites where individuals or groups can post articles, updates, and opinions.
   - Features: Allow users to create and manage content easily, often including options for comments and interaction.
   - Uses: Personal expression, professional updates, niche topics, and discussions. Blogs can serve as platforms for sharing expertise, experiences, or news.
2. **Wikis:**
   Collaborative websites that allow users to create, edit, and manage content collectively.
   - Features: Users can add and modify content with minimal technical knowledge, and changes are tracked through version control.
   - Uses: Knowledge sharing, documentation, and collaborative projects. Wikipedia is a prominent example, allowing anyone to contribute and edit entries.
3. **Social Networking:**
   Platforms that enable users to create profiles, connect with others, share content, and engage in social interactions.
   - Features: Profiles, friend or follower networks, messaging, and content sharing.
   - Uses: Personal communication, professional networking, content sharing, and community building. Examples include Facebook, Twitter, and LinkedIn.

Q.8 What is semantic web? Explain the use and purpose of semantic web. Why is it needed in present web? (8)

Ans:

The Semantic Web is an extension of the World Wide Web designed to make data more understandable by computers. It involves structuring data so that machines can process and interpret it more effectively.

Purpose and Use

1. **Better Data Sharing**: Makes it easier for different systems to share and use data.
2. **Smarter Search**: Improves search results by understanding the context of data.
3. **Automatic Processing**: Allows software to automatically handle and analyze data.
4. **Connected Data**: Links related information for a richer experience.
5. **Enhanced User Experience**: Provides more relevant and personalized services.

Why It's Needed

1. **Huge Data Volume**: Helps manage and make sense of large amounts of data.
2. **Complex Relationships**: Understands and handles complex data connections.
3. **Improved Search**: Offers better and more accurate search results.
4. **Data Integration**: Simplifies combining data from different sources.
5. **Smarter Apps**: Supports the creation of intelligent applications that understand data better.

Q.7 What do you mean by well-formed XML document? Explain DTD with example. (6)

Ans:

A **well-formed XML document** adheres to the basic syntax rules of XML. This ensures that the XML is readable and parsable by XML processors. Key rules include:

1. **Proper Tag Nesting**: Tags must be properly nested, with each opening tag having a corresponding closing tag.
2. **Case Sensitivity**: XML tags are case-sensitive. For example, <Tag> is different from <tag>.
3. **Attribute Quotation**: Attribute values must be enclosed in quotes. For example, <element attribute="value">.
4. **Root Element**: There must be a single root element that encloses all other elements.
5. **No Overlapping Tags**: Tags cannot overlap. For example, <a> is not valid.

DTD (Document Type Definition)

DTD is a set of rules that defines the structure and legal elements of an XML document. It specifies the document's allowed elements, attributes, and their relationships.

Key Components:

1. **Elements**: Defines which elements are allowed and how they are nested.
2. **Attributes**: Specifies attributes for elements and their types.
3. **Entities**: Defines reusable pieces of text or special characters.

Example

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

Q.10 Write short notes on : (2.5)

a. Web Crawling

Ans:

Web Crawling is the process by which search engines use automated bots, known as crawlers or spiders, to systematically browse the internet and index web pages.

Purpose: To discover and update the content available on the web for inclusion in search engine indexes.

Process:

1. Start: Crawlers use a list of initial URLs.
2. Fetch: They visit these URLs and download the page content.
3. Extract Links: They find and follow new links on these pages.
4. Index: The page content is analyzed and stored for search queries.

Web crawling is essential for search engines to provide up-to-date and relevant search results.

c. Server-side programming:

Ans: Server-Side Programming in web technology refers to code that runs on the web server rather than in the user's browser. It processes requests, interacts with databases, and generates dynamic web content.

Key Points:

1. **Execution Location**: Runs on the server, not the client's device.
2. **Dynamic Content**: Generates and serves web pages based on user interactions or data requests.
3. **Database Interaction**: Often used to retrieve, store, or update data in databases.
4. **Languages**: Common languages include PHP, Python, Java, Ruby, and Node.js.
5. **Security**: Handles sensitive data securely, as the code is not exposed to users.

Server-side programming is essential for creating dynamic, data-driven websites and applications by processing data and generating content on the server.

d. Cascading style sheet:

Ans:

Cascading Style Sheets (CSS) is a language used to style and layout web pages. It controls the presentation of HTML elements, making web pages visually appealing and consistent.

Key Points:

1. **Styling**: CSS defines how HTML elements should look, including colours, fonts, and layout.
2. **Selectors**: Targets HTML elements to apply styles. For example, p {color: red; } styles all <p> elements with red text.
3. **Properties**: Specifies style attributes like color, font-size, margin, and padding.
4. **External Styles**: CSS can be written in separate files (e.g., styles.css) and linked to HTML, allowing consistent styling across multiple pages.
5. **Responsive Design**: Supports media queries to adapt styles based on device screen size and resolution.

CSS is essential for designing and controlling the visual appearance of web pages, ensuring that content is both attractive and user-friendly.

Definition of Web1.0, Web2.0 and Web3.0

Ans:

Web 1.0:

The early, static web where pages were read-only and primarily displayed content without user interaction.

Web 2.0:

The interactive web that introduced dynamic content, user-generated content, and social media, allowing users to engage and collaborate online.

Web 3.0:

The semantic web that focuses on data interoperability, artificial intelligence, and decentralized technologies to make web content more intelligent and connected.