WEB ANSWERS

1. **Show the structure of an html document.**

This basic structure includes:

- <!DOCTYPE html>: Declares the document type and version of HTML.
- <html>: The root element of the document.
- <head>: Contains meta-information about the document, such as the title.
- <body>: Contains the content of the document, including a heading and a paragraph.

```
<!DOCTYPE html>
    <html>
    <head>
      <title>Simple Page</title>
    </head>
    <body>
      <h1>Hello, World!</h1>
      <p>This is a simple HTML document.</p>
    </body>
    </html>
```

2. **Why do you call MIME as an extension feature.**

MIME, which stands for Multipurpose Internet Mail Extensions, is called an extension feature because it extends the capabilities of the original email protocol, SMTP (Simple Mail Transfer Protocol). Here are the key reasons:

1. **Enhanced Functionality**: MIME allows for the transmission of non-ASCII data, such as images, audio, video, and application files, which SMTP alone cannot handle[1].
2. **Support for Multiple Content Types**: It defines a method to register new content types and other MIME attribute values, making it extensible[2].
3. **Compatibility**: MIME transforms non-ASCII data into a format that can be sent via SMTP and then converts it back to its original form at the recipient's end[1].

3.What is a Web Browser and How is it Different from a Web Server? 3mark

A web browser is an application used to access and display web pages. Examples include Google Chrome and Mozilla Firefox. A web server is software or hardware that stores, processes, and delivers web pages to clients (browsers). Examples include Apache and Nginx. The key differences are:

1. Purpose: Browsers display web content; servers provide it.
2. Role: Browsers act as interfaces for users; servers host and serve content.
3. Data Handling: Browsers send requests and receive responses; servers receive requests and send responses[1][2].

4.define www ?

The World Wide Web (abbreviated as WWW or W3, commonly known as the Web) is a system of interlinked hypertext documents that are accessed via the Internet. With a web browser, one can view web pages that may contain text, images, videos, and other multimedia and navigate between them via hyperlinks.
• The World Wide Web allows computer users to execute web-based applications and to locate and view multimedia-based documents on almost any subject over the Internet.

5.what is url, what are its parts ?

• The URL (Uniform Resource Locator) specifies the address (i.e., location) of the web page displayed in the browser window. Each web page on the Internet is associated with a unique URL. URLs usually begin with http:/ It consists of several parts:

1. **Scheme**: Indicates the protocol used, such as `http` or `https`.
2. **Domain Name**: Specifies the server hosting the resource, like `example.com`.

3. **Path**: Points to a specific resource on the server, such as `/about.html`.

6.what is mime , list any two features offered by mime to email service .

MIME (Multipurpose Internet Mail Extensions) is a standard that extends the capabilities of email by allowing the transmission of various types of data beyond plain text. It was introduced to overcome the limitations of the original email protocol (SMTP), which could only handle text in the ASCII format.

Two Features Offered by MIME to Email Service:

1. Support for Multiple Attachments: MIME allows a single email to contain multiple attachments, such as documents, images, audio, and video files, all within one message[1].
2. Support for Non-ASCII Ch
3. aracters: MIME enables the use of character sets other than ASCII, allowing emails to include text in various languages and special characters[2].

7.what is absolute positioning?

Absolute positioning is a CSS property that allows you to position an element relative to its nearest positioned ancestor. If no such ancestor exists, it positions the element relative to the initial containing block (usually the document body). Key points include:

1. Removal from Normal Flow: The element is removed from the normal document flow, meaning it does not affect the position of other elements and vice versa.
2. Positioning Properties: You can use `top`, `right`, `bottom`, and `left` properties to specify the exact position of the element within its containing block[1][2].

8.what is callback function in js how it is different from other functions.

A callback function in JavaScript is a function passed as an argument to another function, to be executed after the completion of that function. This allows for asynchronous operations, such as handling events or making API calls.

Differences from Other Functions:

1. Execution Context: Callback functions are executed within the context of another function, whereas regular functions are called independently.
2. Asynchronous Behavior: Callbacks are often used for asynchronous tasks, while regular functions are typically used for synchronous operations.
3. Higher-Order Functions: Callbacks are often used with higher-order functions, which accept other functions as arguments[1][2].

9.what are different ways of adjusting space in a text ?

There are several ways to adjust space in text using CSS:

Line Spacing: Use the `line-height` property to control the space between lines of text. For example:

```
p {
  line-height: 1.5;
}
```

1.

Letter Spacing: Use the `letter-spacing` property to adjust the space between characters. For example:

```
p {
  letter-spacing: 2px;
}
```

Word Spacing: Use the `word-spacing` property to control the space between words. For example:

```
p {
  word-spacing: 5px;
}
```

10.what is relative positioning ?

For a relatively positioned box, the inset properties move the box inward from the respective edge, without changing its size. left moves the box to the right, right moves it to the left, etc.

Since boxes are not split or stretched as a result of relative positioning opposing used values in a given axis must be negations of each other:

If opposing inset properties in an axis both compute to auto (their initial values), their used values are zero (i.e., the boxes stay in their original position in that axis). If only one is auto, its used value becomes the negation of the other, and the box is shifted by the specified amount.

If neither is auto, the position is over-constrained; (with respect to the writing mode of its containing block) the computed end side value is ignored, and its used value becomes the negation of the start side.

1.explain MIME and its type with example .Describe why should MIME type information be essentially included in http response.

MIME* (Multipurpose Internet Mail Extensions) types are a way to specify the nature and format of a file or data. They are used in various internet protocols, including HTTP, to help browsers and other clients understand how to process the data they receive.

Common MIME Types
MIME types are composed of a type and a subtype, separated by a slash. Here are some common examples:

1. *Text Types*:
   - text/plain: Plain text files (e.g., .txt).
   - text/html: HTML files (e.g., .html).

2. *Image Types*:
   - image/jpeg: JPEG images (e.g., .jpg, .jpeg).
   - image/png: PNG images (e.g., .png).

3. *Audio Types*:
   - audio/mpeg: MP3 audio files (e.g., .mp3).
   - audio/ogg: Ogg audio files (e.g., .ogg).

4. *Video Types*:
   - video/mp4: MP4 video files (e.g., .mp4).
   - video/mpeg: MPEG video files (e.g., .mpeg).

5. *Application Types*:
   - application/json: JSON data (e.g., .json).
   - application/pdf: PDF documents (e.g., .pdf).

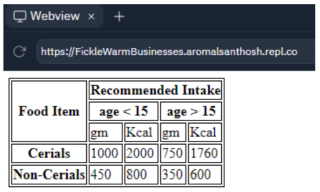
Importance of MIME Type in HTTP Responses
Including MIME type information in HTTP responses is crucial for several reasons:

1. *Content Handling*: Browsers use the MIME type to determine how to handle and display the content. For example, text/html will be rendered as a web page, while application/pdf will be handled by a PDF viewer[1].

2. *Security*: Correct MIME types help prevent security issues. For instance, if a server incorrectly labels a script file as plain text, it could be executed in a way that compromises security[2].

3. *Interoperability*: Ensuring that the correct MIME type is sent helps maintain compatibility across different browsers and devices, ensuring that content is displayed or processed as intended[3].

5.design a web page that displays the table



Food Item	Recommended Intake			
	age < 15		age > 15	
	gm	Kcal	gm	Kcal
Cerials	1000	2000	750	1760
Non-Cerials	450	800	350	600

Ans:
```
<!DOCTYPE html>
<html>
   <head>
      <meta charset="utf-8">
      <title>table</table></title>
   </head>
   <body>
      <table border="1">
         <thead>
            <tr>
               <th rowspan="3">food item</th>
               <th colspan="4">recommended intake</th>

            </tr>
            <tr>
```

```
            <th colspan="2"> age < 15 </th>
            <th colspan="2">age > 15</th>
            </tr>
            <tr>
            <th>gm</th>
                <th>kcal</th>
                <th>gm</th>
                <th>kcal</th>
        </tr>

        <tbody>
            <tr>
                <td>cereals</td>
                <td>1000</td>
                <td>2000</td>
                <td>750</td>
                <td>1760</td>
            </tr>
            <tr>
                <td>non-cereals</td>
                <td>450</td>
                <td>800</td>
                <td>350</td>
                <td>600</td>

            </tr>
        </tbody>
    </thead>
</table>
</body>
</html>
```

3.explain http and its significance .describe request and response phase.

HTTP (Hypertext Transfer Protocol)* is the foundation of data communication on the World Wide Web. It is an application layer protocol used for transmitting hypermedia documents, such as HTML. Here's a breakdown of its significance and the request-response phases:

Significance of HTTP

1. *Client-Server Model*: HTTP operates on a client-server model where the client (usually a web browser) sends requests to the server, which then sends back responses. This model is fundamental to how the web functions[1].

2. *Stateless Protocol*: Each HTTP request from a client to server is independent, meaning the server does not retain any state information between requests. This simplifies the design and implementation of web services[2].
3. *Extensibility*: HTTP is designed to be extensible, allowing for the addition of new methods and headers as needed. This has enabled the protocol to evolve and support new features over time[1].
4. *Foundation for Web Technologies*: HTTP is the backbone of web technologies, enabling the transfer of web pages, images, videos, and other resources. It also supports secure communication through HTTPS[3].

Request and Response Phases
HTTP Request
1. *Client Sends Request*: The client initiates communication by sending an HTTP request to the server. This request includes:
   - *Request Line*: Contains the HTTP method (e.g., GET, POST), the URL, and the HTTP version.
   - *Headers*: Provide additional information about the request, such as the type of content the client can accept.
   - *Body*: Optional, used primarily with methods like POST to send data to the server[6].

HTTP Response
1. *Server Processes Request*: The server receives the request and processes it based on the requested method and resource.
2. *Server Sends Response*: After processing, the server sends back an HTTP response, which includes:
   - *Status Line*: Indicates the HTTP version, status code (e.g., 200 OK, 404 Not Found), and a reason phrase.
   - *Headers*: Provide additional information about the response, such as the type of content being returned.
   - *Body*: Contains the requested resource or data, such as an HTML document, image, or JSON data[7].

4.write an equivalent html code
a)an image titled flowers.jpg with a height of 150 px and width of 250 px if the image cannot be accessed no image available should be displayed
b)a hyperlink to the url www.mysite.com the hyperlink should have label click here
c)an unordered list with value tea coffee milk

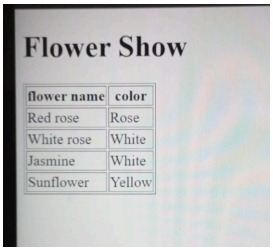
Ans:
<!DOCTYPE html>
<html lang="en">

```
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>HTML Example</title>
</head>
<body>
   <!-- a) Image with alt text -->
   <img src="flowers.jpg" alt="No image available" height="150" width="250">

   <!-- b) Hyperlink -->
   <a href="http://www.mysite.com">Click here</a>

   <!-- c) Unordered list -->
   <ul>
      <li>Tea</li>
      <li>Coffee</li>
      <li>Milk</li>
   </ul>
</body>
</html>
```

5.create a table.



Ans:

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset="utf-8">
      <title>flower show</table></title>
   </head>

   <body>
      <h1>Flower Show</h1>
      <table border="1">
         <thead>
```

```
            <tr>
                <th>Flower name</th>
                <th>color</th>

            </tr>


            <tbody>
                <tr>
                    <td>Red rose</td>
                    <td>Rose</td>

                </tr>
                <tr>
                    <td>White rose</td>
                    <td>White</td>

                </tr>
                <tr>
                    <td>Jasmine</td>
                    <td>White</td>

                </tr>
                <tr>
                    <td>Sunflower</td>
                    <td>Yellow</td>

                </tr>
            </tbody>
        </thead>
    </table>
</body>
</html>
```

6.WRITE HTML code for following
a)a textbox that can accept a maximum of 25 characters
b)3 radio buttons with valid label,names and values
c)three checkboxes with valid label,names and values d)selection list with 4 items d)a submit button to send the form data to sever "http://www.mysite.com/reg.php" using "POST" method and reset button to clear its contents you can use any text of your choice

Ans:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sample Form</title>
</head>
<body>
    <form action="http://www.mysite.com/reg.php" method="POST">
        <!-- Textbox -->
        <label for="username">Username:</label>
        <input type="text" id="username" name="username"
maxlength="25"><br><br>

        <!-- Radio Buttons -->
        <label>Gender:</label><br>
        <input type="radio" id="male" name="gender" value="male">
        <label for="male">Male</label><br>
        <input type="radio" id="female" name="gender" value="female">
        <label for="female">Female</label><br>
        <input type="radio" id="other" name="gender" value="other">
        <label for="other">Other</label><br><br>

        <!-- Checkboxes -->
        <label>Interests:</label><br>
        <input type="checkbox" id="sports" name="interests" value="sports">
        <label for="sports">Sports</label><br>
        <input type="checkbox" id="music" name="interests" value="music">
        <label for="music">Music</label><br>
        <input type="checkbox" id="reading" name="interests" value="reading">
        <label for="reading">Reading</label><br><br>

        <!-- Selection List -->
        <label for="country">Country:</label>
        <select id="country" name="country">
            <option value="india">India</option>
            <option value="usa">USA</option>
```

```
        <option value="uk">UK</option>
        <option value="australia">Australia</option>
    </select><br><br>

    <!-- Submit and Reset Buttons -->
    <input type="submit" value="Submit">
    <input type="reset" value="Reset">
  </form></body></html>
```

7.how are event handlers handled in js.

Ans:

8.what are class selectors in css?

Ans:

A class selector is defined by a period (.) followed by the class name. For example:

CSS

```
.myClass {
    color: blue;
    font-size: 20px;
```

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Class Selectors Example</title>
    <style>
        .highlight {
            background-color: yellow;
        }
        .bold {
            font-weight: bold;
        }
    </style>
</head>
<body>
```

```
    <p class="highlight">This paragraph has a yellow background.</p>
    <p class="bold">This paragraph is bold.</p>
    <p class="highlight bold">This paragraph is bold and has a yellow
background.</p>
</body>
</html>
```
AI-generated code. Review and use carefully. More info on FAQ.
Key Points
Multiple Classes: You can assign multiple classes to an element by separating them with spaces. For example, ... will apply both the highlight and bold styles to the paragraph1.
Specificity: Class selectors have a higher specificity than element selectors but lower than ID selectors. This means they will override element styles but can be overridden by ID styles2. Reusability: Using class selectors promotes reusability and maintainability of your CSS code, as you can apply the same styles to multiple elements without duplicating code3

9.explain css and its types how can css be used to display an xml doc

Ans:
CSS (Cascading Style Sheets) is a language used to describe the presentation of a document written in HTML or XML. CSS controls the layout, colors, fonts, and overall visual appearance of web pages.

Types of CSS
Inline CSS: Styles are applied directly within an HTML element using the style attribute.
HTML

```
<p style="color: blue; font-size: 20px;">This is a paragraph.</p>
```
AI-generated code. Review and use carefully. More info on FAQ.
Internal CSS: Styles are defined within a <style> tag in the <head> section of an HTML document.
HTML

```
<head>
    <style>
        p {
            color: blue;
            font-size: 20px;
        }
    </style>
</head>
```

External CSS: Styles are defined in an external .css file, which is linked to the HTML document using the <link> tag.
HTML

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

CSS can be used to style XML documents to make them more readable and visually appealing. Here's how you can do it:

Create an XML file:
XML

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="styles.css"?>
<books>
   <book>
      <title>Web Programming</title>
      <author>Chris Bates</author>
      <publisher>Wiley</publisher>
      <price>300</price>
   </book>
   <book>
      <title>Computer Networks</title>
      <author>Forouzan</author>
      <publisher>McGraw Hill</publisher>
      <price>700</price>
   </book>
</books>
```

Create a CSS file (styles.css):
CSS

```
books {
   display: block;
   background-color: #f0f0f0;
   padding: 10px;
}
book {
   display: block;
   margin-bottom: 10px;
   border: 1px solid #ccc;
   padding: 10px;
}
title {
```

```
    font-size: 20px;
    font-weight: bold;
    color: #333;
}
author, publisher, price {
    display: block;
    color: #666;
}
```

Link the CSS file to the XML file: The XML file already includes the line <?xml-stylesheet type="text/css" href="styles.css"?> to link the CSS file.

10.Write a code for HTML document with embedded javascript which initially displays the paragraph with text welcome and button titled click. When the button is clicked, the message hello from Javascript in bold should be replace the text paragraph.

Ans:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>JavaScript Example</title>
    <script>
        function changeText() {
            document.getElementById('message').innerHTML = '<b>Hello from
JavaScript</b>';
        }
    </script>
</head>
<body>
    <p id="message">Welcome</p>
    <button onclick="changeText()">Click</button>
</body>
</html>
```

11.explain doc obj model.

Ans:

18.The *Document Object Model (DOM)* is a programming interface for web documents. It represents the structure of a document as a tree of objects, allowing

programs to manipulate the document's structure, style, and content. The DOM is essential for creating dynamic and interactive web pages.

Key Concepts of the DOM
1. *Tree Structure*: The DOM represents a document as a tree of nodes. Each node can be an element, attribute, text, or other types of nodes.
2. *Nodes*: The basic units of the DOM. Examples include element nodes (e.g.,), text nodes (e.g., text inside a <p>), and attribute nodes (e.g., class="example").
3. *Methods and Properties*: The DOM provides methods (e.g., getElementById, createElement) and properties (e.g., innerHTML, style) to access and manipulate nodes.

Example
Let's consider a simple HTML document and how it can be manipulated using the DOM.

HTML Document
html
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>DOM Example</title>
</head>
<body>
    <h1 id="title">Welcome to the DOM</h1>
    <p>This is a paragraph.</p>
    <button onclick="changeContent()">Click Me</button>

    <script>
        function changeContent() {
            // Access the element with id 'title'
            var title = document.getElementById('title');
            // Change the content of the element
            title.innerHTML = 'Hello from the DOM!';
            // Change the style of the element
            title.style.color = 'blue';
        }
    </script>
</body>
</html>
```

Explanation
1. *HTML Structure*: The document includes a heading (<h1>), a paragraph (<p>), and a button (<button>).
2. *JavaScript Function*: The changeContent function is defined to change the content and style of the heading when the button is clicked.
3. *DOM Manipulation*:
   - document.getElementById('title'): Accesses the <h1> element with the id title.
   - title.innerHTML = 'Hello from the DOM!': Changes the content of the <h1> element.
   - title.style.color = 'blue': Changes the color of the <h1> element to blue.

Why the DOM is Important
- *Dynamic Content*: Allows web pages to update content without reloading the entire page.
- *Interactivity*: Enables interaction with user actions (e.g., clicks, inputs).
- *Accessibility*: Provides a structured way to access and manipulate document elements programmatically.

12.Write CSS for the responding code.a)Set the background colour for the hover and active link state to yellow
b)Set list style for ordered list to lowercase alphabet.
c) Set boat dot jpeg as the background image of the page,Set 3 percentage margin for the page.
d)set dotted border for the document.

Ans:
19./* a) Set the background color for the hover and active link state to yellow */
```
a:hover, a:active {
    background-color: yellow;
}

/* b) Set list style for ordered list to lowercase alphabet */
ol {
    list-style-type: lower-alpha;
}

/* c) Set boat.jpg as the background image of the page, with a 3% margin */
body {
```

```
  background-image: url('boat.jpg');
  background-size: cover;
  margin: 3%;
}

/* d) Set a dotted border for the document */
html {
  border: 2px dotted black;
}
```

13.a)Explain different levels of CSS
b). Write javascript programme to find factorial of a number. Use prompt dialogue box to get the input from user.

Ans:

CSS (Cascading Style Sheets) can be applied to HTML documents in three main ways:

1. Inline CSS:
    - Applied directly to HTML elements using the `style` attribute.
    - Example:
    - HTML

```
<p style="color: blue; font-size: 20px;">This is a paragraph.</p>
```

2. Internal CSS:
    - Defined within a `<style>` tag in the `<head>` section of an HTML document.
    - Example:
    - HTML

```
<head>
  <style>
    p {
      color: blue;
      font-size: 20px;
    }
  </style>
```

</head>

3. External CSS:
    - Defined in a separate `.css` file and linked to the HTML document using the `<link>` tag.
    - Example:
    - HTML

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

JavaScript Program to Find Factorial of a Number

Here's a JavaScript program that prompts the user to enter a number and then calculates the factorial of that number:

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Factorial Calculator</title>
  <script>
    function calculateFactorial() {
      // Prompt the user to enter a number
      const number = parseInt(prompt('Enter a positive integer: '));

      // Check if the number is valid
      if (isNaN(number) || number < 0) {
        alert('Please enter a valid positive integer.');
        return;
      }

      // Calculate the factorial
      let factorial = 1;
```

```
        for (let i = 1; i <= number; i++) {
            factorial *= i;
        }

        // Display the result
        alert(`The factorial of ${number} is ${factorial}.`);
    }
  </script>
</head>
<body>
  <button onclick="calculateFactorial()">Calculate Factorial</button>
</body>

</html>
```

14.Explain various types of control statements in Javascript.

Ans:
21.1. Conditional Statements
These statements execute different blocks of code based on certain conditions.

if Statement: Executes a block of code if a specified condition is true.
JavaScript

```
if (condition) {
  // code to be executed if condition is true
}
```

if...else Statement: Executes one block of code if a condition is true, and another block if it is false.
JavaScript

```
if (condition) {
  // code to be executed if condition is true
} else {
  // code to be executed if condition is false
}
```

else if Statement: Allows multiple conditions to be checked in sequence.
JavaScript

```
if (condition1) {
    // code to be executed if condition1 is true
} else if (condition2) {
    // code to be executed if condition2 is true
} else {
    // code to be executed if none of the conditions are true
}
```

switch Statement: Selects one of many blocks of code to be executed.
JavaScript

```
switch (expression) {
    case value1:
        // code to be executed if expression === value1
        break;
    case value2:
        // code to be executed if expression === value2
        break;
    default:
        // code to be executed if expression doesn't match any case
}
```

2. Iterative Statements (Loops)
These statements repeatedly execute a block of code as long as a specified condition is true.

for Loop: Repeats a block of code a specified number of times.
JavaScript

```
for (initialization; condition; increment) {
    // code to be executed
}
```

while Loop: Repeats a block of code as long as a specified condition is true.
JavaScript

```
while (condition) {
    // code to be executed
}
```

do...while Loop: Similar to the while loop, but the block of code is executed at least once before the condition is tested.
JavaScript

```
do {
    // code to be executed
} while (condition);
```

for...in Loop: Iterates over the properties of an object.
JavaScript

```
for (let key in object) {
    // code to be executed for each property
}
```

for...of Loop: Iterates over the values of an iterable object (like an array).
JavaScript

```
for (let value of iterable) {
    // code to be executed for each value
}
```

3. Jump Statements
These statements change the flow of execution by jumping to another part of the program.

break Statement: Exits a loop or a switch statement.
JavaScript

```
for (let i = 0; i < 10; i++) {
    if (i === 5) {
        break; // exits the loop when i is 5
    }
}
```

AI-generated code. Review and use carefully. More info on FAQ.

continue Statement: Skips the current iteration of a loop and continues with the next iteration.
JavaScript

```
for (let i = 0; i < 10; i++) {
    if (i === 5) {
        continue; // skips the iteration when i is 5
    }
    console.log(i);
}
```

return Statement: Exits a function and optionally returns a value.
JavaScript

```
function sum(a, b) {
    return a + b; // exits the function and returns the sum
}
```