

COMHCC-403(Hons)

Write down two characteristics of client/server computing technique.

Client/server computing is a distributed computing technique where a client machine requests and receives services from a server machine. Two characteristics of client/server computing are:

1. **Decentralized architecture:** In client/server computing, the processing power and resources are distributed between the client and server machines. The server machine hosts the central application logic and database, while the client machine provides the user interface and user input. This decentralized architecture allows for greater scalability, reliability, and performance.
2. **Client-Server Model:** The client/server computing model is based on the principle of client and server separation. The server provides services to multiple clients concurrently, while the client requests services from the server. This model allows for multiple clients to access a centralized database and application, providing a more efficient and flexible way of processing data and managing resources.

What type of operators are supported by JavaScript?

JavaScript supports several types of operators, including:

1. **Arithmetic operators:** These are used to perform arithmetic operations on numeric values, such as addition (+), subtraction (-), multiplication (*), division (/), and modulus (%).
2. **Assignment operators:** These are used to assign values to variables, such as the equals sign (=), plus equals (+=), minus equals (-=), multiply equals (*=), divide equals (/=), and modulus equals (%=).
3. **Comparison operators:** These are used to compare values and return a Boolean value (true or false), such as equals to (==), not equals to (!=), greater than (>), less than (<), greater than or equal to (>=), and less than or equal to (<=).
4. **Logical operators:** These are used to combine Boolean values and return a Boolean result, such as logical AND (&&), logical OR (||), and logical NOT (!).
5. **Bitwise operators:** These are used to manipulate the bits of a number, such as bitwise AND (&), bitwise OR (|), bitwise XOR (^), bitwise left shift (<<), bitwise right shift (>>), and bitwise zero-fill right shift (>>>).
6. **Conditional (ternary) operator:** This operator is used for conditional expressions, and it takes three operands: a condition, a value for true, and a value for false. It is represented by the symbol (?).
7. **Comma operator:** This operator is used to evaluate multiple expressions and return the result of the last expression. It is represented by the comma symbol (,).

Define Document Object used JavaScript with suitable example.

The Document Object Model (DOM) is a programming interface for web documents. It represents the page so that programs can change the document structure, style, and content. In JavaScript, the Document Object represents the HTML document that is currently displayed in the web browser. It provides access to all the elements on the page, allowing developers to manipulate and change their properties and content dynamically.

For example, let's say we have an HTML page with a paragraph element that has an ID of "myParagraph":

```
<!DOCTYPE html>
<html>
<head>
  <title>My Page</title>
</head>
<body>
  <p id="myParagraph">This is my paragraph.</p>
</body>
</html>
```

We can use JavaScript to access and change the content of the paragraph element like this:

```
// Access the paragraph element by its ID
var myParagraph = document.getElementById("myParagraph");

// Change the text content of the paragraph
myParagraph.textContent = "This is the new content of my paragraph.";
```

In this example, we use the `document.getElementById()` method to retrieve the paragraph element with the ID "myParagraph". We then use the `textContent` property to change the text content of the paragraph to "This is the new content of my paragraph." This is just one example of how the Document Object can be used in JavaScript to manipulate the content of a web page.

State basic difference between HTML and XML documents.

HTML (Hypertext Markup Language) and XML (Extensible Markup Language) are both markup languages used to create structured documents. However, there are some fundamental differences between the two:

1. Purpose: HTML is used to create web pages that are displayed in a web browser, while XML is used to store and transport data.
2. Syntax: HTML has a fixed set of tags and attributes defined by the W3C (World Wide Web Consortium), which are used to structure web pages. XML, on the other hand, allows developers to define their own tags and attributes to create custom document structures.
3. Validation: HTML does not require validation, as it is forgiving and can render pages even if they contain errors. XML, on the other hand, requires validation to ensure that the data is well-formed and conforms to a specified structure.
4. Display: HTML is designed to be displayed in a web browser, which means that it includes tags for formatting, layout, and styling. XML, on the other hand, is not designed for display, and does not include any formatting or styling tags.
5. Extensibility: XML is extensible, which means that developers can create custom document types and schema definitions to define the structure and content of their documents. HTML is not extensible in the same way, as it is based on a fixed set of tags and attributes.