Wd - Javascript Basic &Amp; Dom

JavaScript Basics

1 Introduction to JavaScript:

JavaScript is a versatile scripting language used primarily for adding interactivity to web pages. It runs in the browser and can be used for tasks such as:

2. Syntax and Structure:

*Variables: Use **var**, **let**, or **const** to declare variables.*

```
let name = 'John';
const age = 30;
var isStudent = true;
```

Data Types: Numbers, strings, booleans, objects, arrays, functions, etc.

```
let number = 5; // Number
let text = 'Hello'; // String
let isTrue = false; // Boolean
let person = { name: 'John', age: 30 }; // Object
let numbers = [1, 2, 3]; // Array
```

Functions: Define reusable blocks of code.

```
function greet() {
  console.log('Hello, World!');
}

greet(); // Calls the function
```

Control Structures: if, for, while, etc.

```
if (age > 18) {
  console.log('Adult');
} else {
  console.log('Minor');
}

for (let i = 0; i < 5; i++) {
  console.log(i);
}
```

*3. **Basic DOM Manipulation***

The DOM is a programming interface for web documents. It represents the page so that programs can change the document structure, style, and content.

***Selecting Elements**: Use methods like getElementById, getElementsByClassName, querySelector, etc.*

```
let header = document.getElementById('header');
let items = document.getElementsByClassName('item');
let firstItem = document.querySelector('.item');
```

***Manipulating Elements**: Change content, attributes, styles, etc.*

```
header.textContent = 'New Header';
firstItem.style.color = 'red';
```

***Handling Events**: Respond to user actions such as clicks, form submissions, etc.*

```
header.addEventListener('click', function() {
  alert('Header clicked!');
});
```

Document Object Model (DOM)

*1. **Understanding the DOM***

The DOM is a tree-like structure where each node represents part of the document. Elements, attributes, and text are nodes in this structure.

*2. **DOM Methods and Properties***

***Traversal:** Navigate through the DOM tree.*

```
let parent = firstItem.parentNode;
let children = header.childNodes;
let nextSibling = firstItem.nextSibling;
```

Creating and Inserting Elements: Dynamically add content.

```
let newDiv = document.createElement('div');
newDiv.textContent = 'Hello!';
document.body.appendChild(newDiv);
```

Removing Elements: Remove content from the document.

```
document.body.removeChild(newDiv);
```

Practical Example

Here's a simple example that combines basic JavaScript and DOM manipulation:

HTML:

```
<!DOCTYPE html>
<html>
<head>
  <title>JavaScript and DOM Example</title>
</head>
<body>
  <h1 id="header">Hello, World!</h1>
  <ul>
```

JavaScript (script.js):

```
// Select elements
let header = document.getElementById('header');
let items = document.getElementsByClassName('item');
let button = document.getElementById('changeText');

// Change header text on button click
button.addEventListener('click', function() {
  header.textContent = 'Hello, JavaScript!';
});

// Change color of list items
for (let i = 0; i < items.length; i++) {
  items[i].style.color = 'blue';
}
```