Building a Basic Web Calculator

1 **Building a Basic Web Calculator with HTML, CSS & JavaScript (and a glimpse into Java)**

2 Matam Rajini

3 **May 19, 2025**

What is a Web Calculator?

Briefly explain what a basic calculator does.
Mention that it's a common interactive element on websites.
Highlight the technologies involved:

- HTML: Structure of the calculator (buttons, display)
- .
- CSS: Styling and layout (how it looks).
- JavaScript: Interactivity and logic (how it works).
- (Optional) Java: Briefly mention its potential role in more complex web applications for server-side logic.

The Structure (Part 1)

Title: HTML - Structuring the Calculator

Show a simplified HTML code snippet focusing on the main container and the display area:

```
<!DOCTYPE html>
<html>
<head>
   <title>Basic Calculator</title>
   <link rel="stylesheet" href="style.css">
</head>
<body>
   <div class="calculator">
      <div class="display"
id="display">0</div>
      <div class="buttons">
         </div>
   </div>
   <script src="script.js"></script>
</body>
</html>
```

Key points:

<!DOCTYPE html>, <html>, <head>, <body> - Basic HTML structure.

 - The main container for the calculator.

 - Where the input and results are shown (note the id for JavaScript).

 - Container for the calculator buttons.

<link rel="stylesheet" href="style.css"> - Linking to the CSS file.

<script src="script.js"></script> - Linking to the JavaScript file.

The Structure (Part 2)

Title: HTML - Adding the Buttons

Show a snippet of the HTML for the buttons within the .buttons div:

HTML

```
<button onclick="clearDisplay()">C</button>
<button onclick="deleteLast()">DEL</button>
<button class="operator" onclick="appendOperator('/')">/</button>
<button class="operator" onclick="appendOperator('*')">*</button>
<button onclick="appendNumber('7')">7</button>
<button onclick="appendNumber('8')">8</button>
<button class="equal" onclick="calculate()">=</button>
```

Key points:

Each button is a <button> element.
onclick attribute: This is how user interaction triggers JavaScript functions.
class attribute (e.g., operator, equal): Used for styling with CSS.

CSS - Styling the Calculator

```
.calculator {
    width: 300px;
    margin: 50px auto;
    border: 1px solid #ccc;
    border-radius: 5px;
    overflow: hidden;
}

.display {
    width: 100%;
    padding: 20px;
    font-size: 24px;
    text-align: right;
    background-color: #f0f0f0;
    box-sizing: border-box;
}

.buttons {
    display: grid;
    grid-template-columns: repeat(4, 1fr);
}

button {
    padding: 20px;
    font-size: 20px;
```

JavaScript - The Logic (Part 1)

- **Title: JavaScript - Making it Interactive**

- **Show a snippet of the JavaScript code (script.js):**

JavaScript

```
letdocument'display'letrrentInput = '';
let operator = null;
let previousValue = null;

functionappendNumber(number) {
    if (currentInput === '0' && number !== '.') {
        currentInput = number;
    } elsefunctionappendOperator(op) // ... logic for handling operators ...functionclearDisplay() // ... logic to clear the display and variables ...
```

- **Key points:**
  - document.getElementById('display') gets a reference to the display element.
  - Variables (currentInput, operator, previousValue) to store the calculator's state.
  - appendNumber() function: Appends numbers to the display.

JavaScript - The Logic (Part 2)

Title: JavaScript - Performing Calculations

Show snippets of the appendOperator() and calculate() functions

```
function appendOperator(op) {
    if (currentInput === '' && previousValue === null) {
        return;
    }
    if (previousValue !== null) {
        calculate();
    }
    previousValue = parseFloat(currentInput);
    operator = op;
    currentInput = '';
}

function calculate() {
    if (operator === null || previousValue === null) {
        return;
```

(Optional) The Role of Java - A More Complex Scenario

•Title: (Optional) Java in Web Applications - Beyond the Basics

•Explain that for a basic calculator, JavaScript handles everything in the browser.

•Introduce the idea that in more complex web applications, Java (or other server-side languages) can be used for:

- ◦Handling more intricate calculations.
- ◦Storing user data or calculation history in a database.
- ◦Integrating with other server-side systems or APIs.

•Show a very simplified conceptual diagram:

•Briefly mention that JavaScript on the front-end would communicate with the Java server (e.g., using fetch API) to send data and receive results

Conclusion

Recap the key technologies used for a basic web calculator: HTML, CSS, and JavaScript. Reinforce that JavaScript is essential for the interactive logic in the browser

.

(If you included Slide 8) Briefly reiterate that Java can play a role in more complex web applications for server-side tasks.

Thank the audience

Keywords

Add your words here,according to your need to draw the text box size

Keywords

Add your words here,according to your need to draw the text box size

Keywords

Add your words here,according to your need to draw the text box size

Keywords

Add your words here,according to your need to draw the text box size

Thanks!