```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Basic Calculator</title>
    <style>
      /* General Reset */
      * {
          margin: 0;
          padding: 0;
          box-sizing: border-box;
      }

      body {
          font-family: Arial, sans-serif;
          background-color: #f1f1f1;
          display: flex;
          justify-content: center;
          align-items: center;
          height: 100vh;
      }

      /* Calculator Container */
      .calculator {
          width: 280px;
          background-color: #fff;
          border-radius: 10px;
          box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
          padding: 20px;
      }
```

```css
/* Display Section */
.display {
    width: 100%;
    height: 50px;
    background-color: #222;
    color: #fff;
    text-align: right;
    padding: 10px;
    border-radius: 5px;
    font-size: 24px;
    overflow: hidden;
}

/* Button Grid */
.buttons {
    display: grid;
    grid-template-columns: repeat(4, 1fr);
    gap: 10px;
    margin-top: 20px;
}

.buttons button {
    background-color: #f1f1f1;
    border: 1px solid #ccc;
    font-size: 20px;
    padding: 20px;
    cursor: pointer;
    border-radius: 5px;
    transition: background-color 0.3s;
}
```

```css
.buttons button:hover {
    background-color: #ddd;
}


.buttons button:active {
    background-color: #bbb;
}


.buttons button.operator {
    background-color: #f97e1d;
    color: #fff;
}


.buttons button.operator:hover {
    background-color: #e26b14;
}


.buttons button.equal {
    background-color: #28a745;
    color: #fff;
    grid-column: span 2;
}


.buttons button.equal:hover {
    background-color: #218838;
}


.buttons button.clear {
    background-color: #dc3545;
    color: #fff;
```

```
          grid-column: span 2;

        }


      .buttons button.clear:hover {

        background-color: #c82333;

      }
    </style>
</head>
<body>
    <div class="calculator">
      <!-- Display Section -->
      <div class="display" id="display">0</div>


      <!-- Button Grid -->
      <div class="buttons">
        <button class="clear">C</button>

        <button>/</button>

        <button>*</button>

        <button>-</button>


        <button>7</button>

        <button>8</button>

        <button>9</button>

        <button>+</button>


        <button>4</button>

        <button>5</button>

        <button>6</button>

        <button>.</button>


        <button>1</button>
```

```html
        <button>2</button>
        <button>3</button>
        <button class="equal">=</button>


        <button>0</button>
      </div>
    </div>

    <script>
        const display = document.getElementById('display');
        let currentInput = '';
        let previousInput = '';
        let operator = '';


        // Update the display
        function updateDisplay(value) {
            display.textContent = value;
        }


        // Handle button clicks
        const buttons = document.querySelectorAll('.buttons button');
        buttons.forEach(button => {
            button.addEventListener('click', () => {
                const buttonText = button.textContent;


                // Clear the display
                if (buttonText === 'C') {
                    currentInput = '';
                    previousInput = '';
                    operator = '';
                    updateDisplay('0');
```

```javascript
      return;
    }


    // Handle number buttons
    if (/\d/.test(buttonText) || buttonText === '.') {
      currentInput += buttonText;
      updateDisplay(currentInput);
    }


    // Handle operators
    if (['+', '-', '*', '/'].includes(buttonText)) {
      if (currentInput === '') return;
      if (previousInput !== '') {
        currentInput = calculate(previousInput, currentInput, operator);
        updateDisplay(currentInput);
      }
      operator = buttonText;
      previousInput = currentInput;
      currentInput = '';
    }


    // Handle equal button
    if (buttonText === '=') {
      if (previousInput !== '' && currentInput !== '') {
        currentInput = calculate(previousInput, currentInput, operator);
        updateDisplay(currentInput);
        previousInput = '';
        operator = '';
      }
    }
  });
```

```
        });

        // Perform the calculation
        function calculate(num1, num2, operator) {
            num1 = parseFloat(num1);
            num2 = parseFloat(num2);
            switch (operator) {
                case '+':
                    return num1 + num2;
                case '-':
                    return num1 - num2;
                case '*':
                    return num1 * num2;
                case '/':
                    if (num2 === 0) {
                        return 'Error'; // Prevent division by zero
                    }
                    return num1 / num2;
                default:
                    return num2;
            }
        }
    </script>
</body>
</html>
```