Simple Calculator — HTML, CSS & JavaScript

This PDF contains a minimal, easy-to-use calculator built with HTML, CSS and vanilla JavaScript. Copy the source code into a file named **index.html**, open it in your browser, and try it out.

Usage: Click the buttons or type numbers and operators, then press **=** to compute. The calculator supports basic operations: addition, subtraction, multiplication and division.

Source code (save as index.html)

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <title>Simple Calculator</title>
  <meta name="viewport" content="width=device-width,initial-scale=1" />
  <style>
    :root{font-family:system-ui,Segoe UI,Roboto,Helvetica,Arial,sans-serif}
    body{display:flex;min-height:100vh;align-items:center;justify-content:center;background:#f3f4f6;marg
    .calc{width:320px;background:#fff;border-radius:12px;padding:16px;box-shadow:0 6px 18px rgba(0,0,0,0
    .display{height:56px;background:#0f172a;color:#fff;border-radius:8px;padding:8px 12px;font-size:20px
    .prev{font-size:12px;opacity:0.7}
    .curr{font-weight:600;word-break:break-all}
    .keys{display:grid;grid-template-columns:repeat(4,1fr);gap:8px;margin-top:12px}
    button{height:52px;border-radius:8px;border:0;font-size:18px;background:#e6e7eb;cursor:pointer}
    button.op{background:#f97316;color:#fff}
    button.equals{background:#0ea5a4;color:#fff;grid-column:span 2}
    button.span2{grid-column:span 2}
    button:active{transform:translateY(1px)}
  </style>
</head>
<body>
  <div class="calc" role="application" aria-label="Simple calculator">
    <div class="display" id="display" aria-live="polite">
      <div class="prev" id="previous"></div>
      <div class="curr" id="current">0</div>
    </div>
    <div class="keys">
      <button class="span2" data-action="all-clear">AC</button>
      <button data-action="delete">DEL</button>
      <button class="op" data-action="/">÷</button>

      <button data-digit="7">7</button>
      <button data-digit="8">8</button>
      <button data-digit="9">9</button>
      <button class="op" data-action="*">×</button>

      <button data-digit="4">4</button>
      <button data-digit="5">5</button>
      <button data-digit="6">6</button>
      <button class="op" data-action="-">−</button>

      <button data-digit="1">1</button>
      <button data-digit="2">2</button>
      <button data-digit="3">3</button>
      <button class="op" data-action="+">+</button>

      <button data-digit="0">0</button>
      <button data-digit=".">.</button>
      <button class="equals" data-action="=">=</button>
    </div>
  </div>

  <script>
    // Simple calculator logic
    const prevEl = document.getElementById('previous');
    const currEl = document.getElementById('current');
    let current = '';
    let previous = '';
    let operation = null;

    function updateDisplay() {
      currEl.textContent = current || '0';
      prevEl.textContent = previous ? previous + (operation || '') : '';
    }

    function appendDigit(d) {
      if (d === '.' && current.includes('.')) return;
```

```
      if (current === '0' && d !== '.') current = d;
      else current += d;
    }

    function chooseOp(op) {
      if (!current && previous) {
        operation = op;
        updateDisplay();
        return;
      }
      if (!current) return;
      if (previous) {
        compute();
      }
      operation = op;
      previous = current;
      current = '';
    }

    function compute() {
      const a = parseFloat(previous);
      const b = parseFloat(current);
      if (isNaN(a) || isNaN(b)) return;
      let res = 0;
      switch (operation) {
        case '+': res = a + b; break;
        case '-': res = a - b; break;
        case '*': res = a * b; break;
        case '/': res = b === 0 ? 'Error' : a / b; break;
        default: return;
      }
      current = String(res);
      previous = '';
      operation = null;
    }

    function allClear() {
      current = '';
      previous = '';
      operation = null;
    }

    function deleteDigit() {
      current = current.slice(0, -1);
    }

    document.querySelectorAll('button').forEach(btn => {
      btn.addEventListener('click', () => {
        if (btn.dataset.digit) {
          appendDigit(btn.dataset.digit);
        } else {
          const act = btn.dataset.action;
          if (act === 'all-clear') allClear();
          else if (act === 'delete') deleteDigit();
          else if (act === '=') { compute(); }
          else if (['+','-','*','/'].includes(act)) chooseOp(act);
        }
        updateDisplay();
      });
    });

    // Keyboard support
    window.addEventListener('keydown', (e) => {
      if ((e.key >= '0' && e.key <= '9') || e.key === '.') appendDigit(e.key);
      else if (['+','-','*','/'].includes(e.key)) chooseOp(e.key);
      else if (e.key === 'Enter' || e.key === '=') { compute(); }
      else if (e.key === 'Backspace') deleteDigit();
      else if (e.key.toLowerCase() === 'c') allClear();
      updateDisplay();
    });

    updateDisplay();
  </script>
</body>
</html>
```

Notes: This is a minimal example intended for learning and small projects. For production use, add more input validation and accessibility improvements.