

CODING:

```
import React, { useState } from 'react';

const Calculator = () => {

  const [expression, setExpression] = useState("");
  const [displayText, setDisplayText] = useState("");
  const handleClick = (value) => {

    if (value === 'C') {
      setExpression("");
      setDisplayText("");
    } else if (value === '=') {
      try {
        const result = eval(expression);
        setDisplayText(`${expression} = ${result}`);
        setExpression(""); // Clear for next input
      } catch (error) {
        setDisplayText('Error');
      }
    } else {
      setExpression(expression + value);
      setDisplayText(expression + value);
    }
  };

  const buttons = [
    '7', '8', '9', '/',
    '4', '5', '6', '*',
    '1', '2', '3', '-',
    '0', '.', '=', '+',
    'C' ];

  const styles = {
    container: {
      display: 'flex',
```

```
justifyContent: 'center',
alignItems: 'center',
height: '100vh',
backgroundColor: '#eef2f3',
fontFamily: 'Arial, sans-serif',
},
calculator: {
  backgroundColor: '#fff',
  padding: '20px',
  borderRadius: '16px',
  boxShadow: '0 6px 12px rgba(0, 0, 0, 0.15)',
  width: '300px',
  textAlign: 'center',
},
title: {
  fontSize: '22px',
  fontWeight: 'bold',
  marginBottom: '15px',
},
display: {
  width: '100%',
  padding: '15px',
  fontSize: '22px',
  textAlign: 'right',
  marginBottom: '20px',
  border: 'none',
  backgroundColor: '#f2f2f2',
  borderRadius: '8px',
},
buttons: {
  display: 'grid',
```

```
    gridTemplateColumns: 'repeat(4, 1fr)',
    gap: '12px',
  },
  button: {
    padding: '18px',
    fontSize: '18px',
    border: 'none',
    borderRadius: '10px',
    backgroundColor: '#007bff',
    color: 'white',
    cursor: 'pointer',
  },
  clearButton: {
    gridColumn: 'span 4',
    backgroundColor: '#dc3545',
  }
};

return (
  <div style={styles.container}>
    <div style={styles.calculator}>
      <div style={styles.title}>Simple Calculator</div>
      <div style={styles.display}>{displayText}</div>
      <div style={styles.buttons}>
        {buttons.map((btn, i) => (
          <button
            key={i}
            onClick={() => handleClick(btn)}
            style={{
              ...styles.button,
              ...(btn === 'C' ? styles.clearButton : {})
            }}
          />
        ))}
      </div>
    </div>
  </div>
);
```

```

        >
        {btn}
    </button>
  )))
</div>
</div>
</div>
);
};
export default Calculator;

```

OUTPUT:

