WEBTECHNOLOGY

CALCULATOR

CASE STUDY:

<html>

<head>

<title>Calculator</title>

<link rel="stylesheet" href="style.css" />

</head>

<body>

<div class="calculator-container">

<h1>Calculator</h1>

<div class="container">

<input type="text" class="display" />

<div class="buttons">

<button class="operator" data-value="AC">AC</button>

<button class="operator" data-value="DEL">DEL</button>

<button class="operator" data-value="%">%</button>

<button class="operator" data-value="/">/</button>

<button data-value="7">7</button>

<button data-value="8">8</button>

<button data-value="9">9</button>

<button class="operator" data-value="\*">\*</button>

<button data-value="4">4</button>

<button data-value="5">5</button>

<button data-value="6">6</button>

<button class="operator" data-value="-">-</button>

<button data-value="1">1</button>

<button data-value="2">2</button>

<button data-value="3">3</button>

<button class="operator" data-value="+">+</button>

<button data-value="0">0</button>

<button data-value="00">00</button>

<button data-value=".">.</button>

<button class="operator" data-value="=">=</button>

</div>

</div>

</div>

<script src="script.js"></script>

</body>

</html>

/\*css stytle\*/

\* {

box-sizing: border-box;

font-family: "Poppins", sans-serif;

}

body {

height: 100vh;

display: flex;

align-items: center;

justify-content: center;

background-color:fuchsia; /\* Light background color for the page \*/

}

.calculator-container {

text-align: center;

background-color:yellow ; /\* White background for the container \*/

padding: 20px;

border-radius: 15px;

box-shadow: 0 5px 15px rgba(0, 0, 0, 0.1); /\* Subtle shadow around the container \*/

width: 350px;

border: 2px solid black; /\* Light gray border around the whole container \*/

}

h1 {

font-size: 32px;

color: purple; /\* Dark gray color for the title \*/

margin-bottom: 20px;

}

.container {

position: relative;

max-width: 300px;

width: 100%;

padding: 10px 20px 20px;

background-color: green; /\* White background for the calculator container \*/

border-radius: 12px;

box-shadow: 0 5px 10px rgba(0, 0, 0, 0.05); /\* Subtle shadow for the calculator \*/

border: 2px solid purple; /\* Light gray border \*/

}

.display {

height: 80px;

width: 100%;

outline: none;

border: 2px solid #e0e3eb; /\* Light gray border for the display \*/

text-align: right;

margin-bottom: 10px;

font-size: 25px;

color: #000e1a; /\* Dark text color \*/

background-color: #f9f9f9; /\* Light background color for the display \*/

padding: 10px;

border-radius: 6px;

pointer-events: none;

}

.buttons {

display: grid;

grid-gap: 10px;

grid-template-columns: repeat(4, 1fr);

}

.buttons button {

padding: 15px;

border-radius: 6px;

border: none;

font-size: 20px;

cursor: pointer;

background-color: lightpink; /\* Light button background \*/

transition: background-color 0.2s ease;

}

.buttons button:active {

transform: scale(0.98); /\* Slight shrinking effect on button press \*/

}

.operator {

color:maroon; /\* Light blue color for operator buttons \*/

background-color: #dcdfe2; /\* Slightly darker background for operator buttons \*/

}

.buttons button:hover {

background-color: #d1d1d1; /\* Darker background when hovering over buttons \*/

}

/\*javascript\*/

const display = document.querySelector(".display");

const buttons = document.querySelectorAll("button");

const specialChars = ["%", "\*", "/", "-", "+", "="];

let output = "";

//Define function to calculate based on button clicked.

const calculate = (btnValue) => {

display.focus();

if (btnValue === "=" && output !== "") {

//If output has '%', replace with '/100' before evaluating.

output = eval(output.replace("%", "/100"));

} else if (btnValue === "AC") {

output = "";

} else if (btnValue === "DEL") {

//If DEL button is clicked, remove the last character from the output.

output = output.toString().slice(0, -1);

} else {

//If output is empty and button is specialChars then return

if (output === "" && specialChars.includes(btnValue)) return;

output += btnValue;

}

display.value = output;

};

//Add event listener to buttons, call calculate() on click.

buttons.forEach((button) => {

//Button click listener calls calculate() with dataset value as argument.

button.addEventListener("click", (e) => calculate(e.target.dataset.value));

});

**OUTPUT:**





