Learner Assignment Submission Format

Learner Details

- **Name:H O Akash**
- **Enrollment Number:SU625MR015**
- **Batch / Class:MERN Stack**
- **Assignment:Age Calculator**
- **Date of Submission:17/07/2025**

Problem Solving Activity 1.1

1. Program Statement

Age calculator using Javascript

---

2. Algorithm

1. Get the date of birth (DOB) from the user input
2. Get the current date (today)
3. Calculate the difference in years between the current year and the DOB year
4. Calculate the difference in months between the current month and the DOB month
5. Calculate the difference in days between the current day and the DOB day
6. If the difference in days is negative, subtract 1 from the months and add the number of days in the previous month to the days
7. If the difference in months is negative, subtract 1 from the years and add 12 to the months
8. Display the result in the format "You are X years, Y months, and Z days old."

---

3. Pseudocode

```
BEGIN
  GET dob FROM user input
  GET today FROM current date
  CALCULATE years = today.year - dob.year
  CALCULATE months = today.month - dob.month
  CALCULATE days = today.day - dob.day
  IF days < 0 THEN
    months = months - 1
    days = days + NUMBER_OF_DAYS_IN_PREVIOUS_MONTH
  IF months < 0 THEN
    years = years - 1
    months = months + 12
  DISPLAY "You are " + years + " years, " + months + " months, and " + days + " days old."
END
```

---

4. Program Code

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Age Calculator</title>
    <style>
        body {
            font-family: Arial, sans-serif;
```

```
        margin: 20px;
    }
    form {
        margin-bottom: 20px;
    }
    input[type="date"] {
        padding: 10px;
        margin-right: 10px;
    }
    button {
        padding: 10px 15px;
    }
    #result {
        margin-top: 20px;
        font-weight: bold;
    }
    .a{
        justify-content: center;
        margin-left: 500px;
        border-style: solid;
        width: 400px;
        padding-left: 60px;
        height: 250px;
        margin-top: 230px;
        border-radius: 20px;
        background-color: aqua;
```

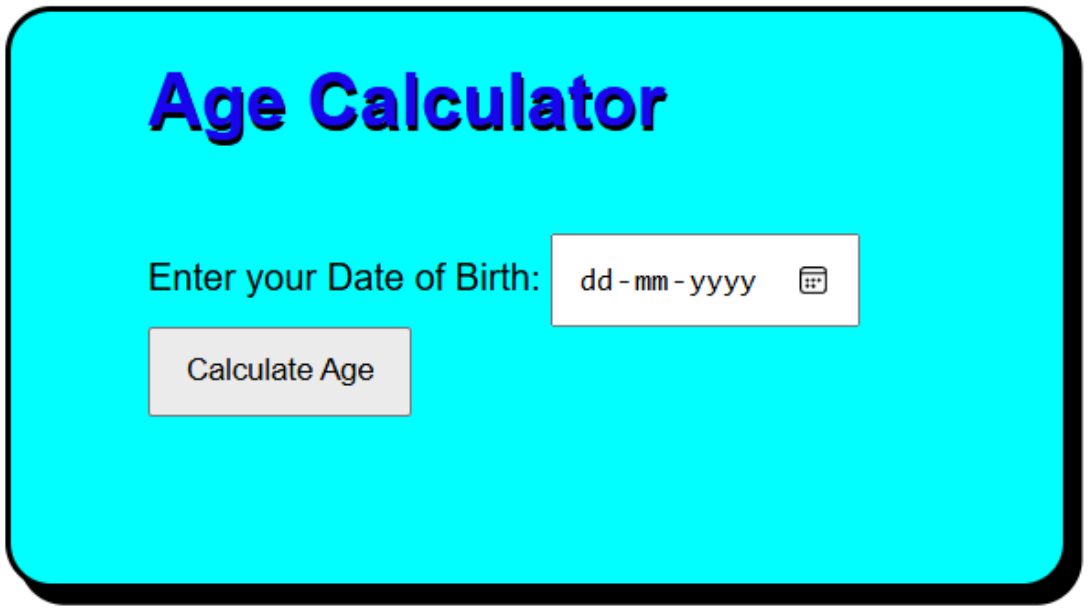
```
            box-shadow: 6px 7px black;



        }
         .document{
            margin-left: 100px;
         }
         h1{
            color: rgb(27, 8, 237);
            text-shadow: 1px 2px black;

         }
    </style>
</head>
<body>
   <div class="a"><h1>Age Calculator</h1><br>
    <form id="ageForm">
        <label for="dob">Enter your Date of Birth:</label>
        <input type="date" id="dob" required><br>
        <button type="submit">Calculate Age</button>
    </form>
     <div id="result"></div>
</div>



    <script>
```

```
        document.getElementById('ageForm').addEventListener('submit',
function(event) {
            event.preventDefault();

            const dob = new Date(document.getElementById('dob').value);
            const today = new Date();

            let years = today.getFullYear() - dob.getFullYear();
            let months = today.getMonth() - dob.getMonth();
            let days = today.getDate() - dob.getDate();

            if (days < 0) {
                months--;
                days += new Date(today.getFullYear(), today.getMonth(),
0).getDate();
            }

            if (months < 0) {
                years--;
                months += 12;
            }

            document.getElementById('result').innerText = `You are
${years} years, ${months} months, and ${days} days old.`;
        });
    </script>
</body>
```

```
</html>
```

---

6. Screenshots of Output



---

7. Observation / Reflection

The given JavaScript code is a simple age calculator that calculates the age of a person based on their date of birth. The code is well-structured and easy to understand. Here are some reflections:

- The code uses a modular approach by handling the form submission event, making it easy to maintain and extend.

- The code calculates the age by taking into account the differences in years, months, and days, making it accurate and reliable.

- The code handles edge cases, such as when the difference in days or months is negative, making it robust and error-free.

- The code displays the result in a clear and concise manner.