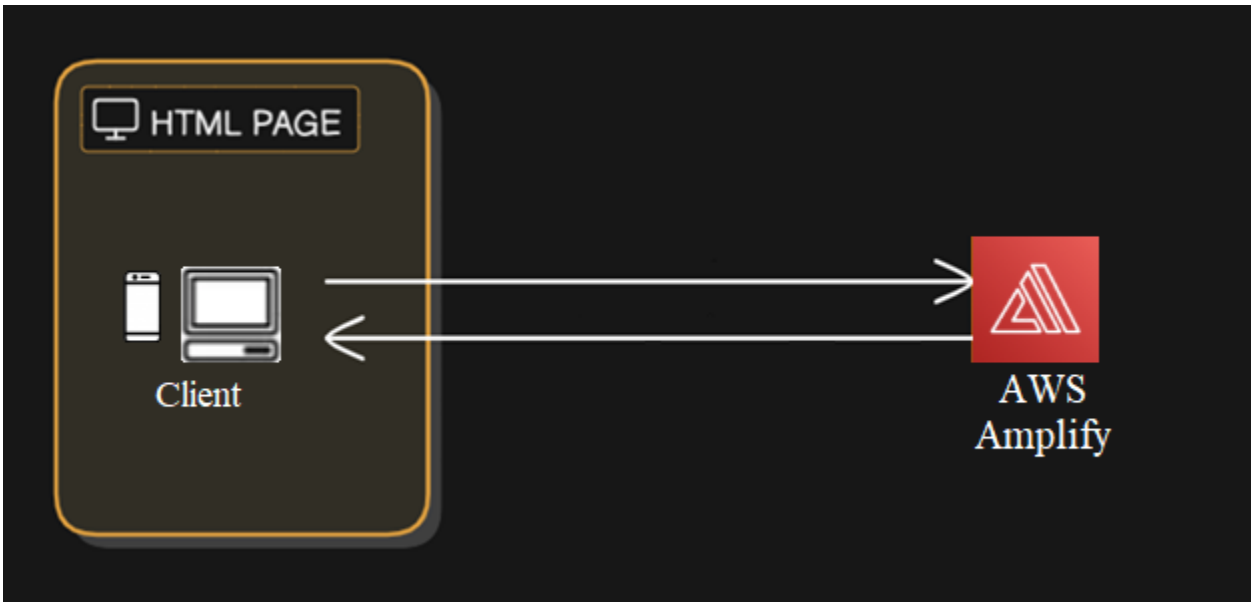
Lab: Host a static website using AWS Amplify

Lab overview and objectives

In this lab, you will use AWS Amplify to host a static website.

After completing this lab, you should be able to:

- Create a static website by using AWS Amplify.



Scenario

You will create a basic HTML page for a calculator and will store it as a static web page in AWS Amplify.

Accessing the AWS Management Console

For this lab, we will use the **AWS Academy learner** lab and the **AWS Management Console**.

Task 1: Creating a calculator HTML

In this task, you will create a basic HTML page that will later be hosted on AWS Amplify. Writing code to complete this task is optional.

1. *Write a colorful HTML calculator page*, call it **index.html**.
2. Zip the index.html file into a zip file: **calc.zip**.
   Alternatively, copy the following code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Fixed Calculator</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      text-align: center;
      margin-top: 50px;
    }
    .calculator {
      display: inline-block;
      border: 1px solid #ccc;
      border-radius: 10px;
      padding: 20px;
      box-shadow: 2px 2px 10px rgba(0, 0, 0, 0.1);
    }
    .display {
      width: 100%;
      height: 50px;
      margin-bottom: 10px;
      text-align: right;
      font-size: 1.5em;
```

```
            padding: 10px;
            border: 1px solid #ccc;
            border-radius: 5px;
            box-sizing: border-box;
        }
        .buttons {
            display: grid;
            grid-template-columns: repeat(4, 1fr);
            gap: 10px;
        }
        .button {
            padding: 15px;
            font-size: 1.2em;
            border: none;
            border-radius: 5px;
            background-color: #f0f0f0;
            cursor: pointer;
        }
        .button:hover {
            background-color: #ddd;
        }
        .button.operator {
            background-color: #f9a825;
            color: white;
        }
        .button.operator:hover {
            background-color: #f57f17;
        }
        .button.equals {
            background-color: #43a047;
            color: white;
            grid-column: span 2;
        }
        .button.equals:hover {
            background-color: #2e7d32;
        }
        .button.clear {
            background-color: #e53935;
            color: white;
            grid-column: span 2;
        }
        .button.clear:hover {
            background-color: #c62828;
        }
    </style>
</head>
<body>
    <div class="calculator">
        <input type="text" class="display" id="display" disabled />
```

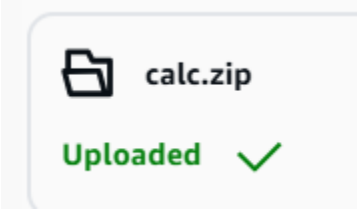
```
    <div class="buttons">
        <button class="button" onclick="appendNumber('7')">7</button>
        <button class="button" onclick="appendNumber('8')">8</button>
        <button class="button" onclick="appendNumber('9')">9</button>
        <button class="button operator" onclick="setOperator('/')">/</button>

        <button class="button" onclick="appendNumber('4')">4</button>
        <button class="button" onclick="appendNumber('5')">5</button>
        <button class="button" onclick="appendNumber('6')">6</button>
        <button class="button operator" onclick="setOperator('*')">×</button>

        <button class="button" onclick="appendNumber('1')">1</button>
        <button class="button" onclick="appendNumber('2')">2</button>
        <button class="button" onclick="appendNumber('3')">3</button>
        <button class="button operator" onclick="setOperator('-')">-</button>

        <button class="button" onclick="appendNumber('0')">0</button>
        <button class="button" onclick="appendNumber('.')">.</button>
        <button class="button clear" onclick="clearDisplay()">C</button>
        <button class="button operator" onclick="setOperator('+')">+</button>

        <button class="button equals" onclick="calculateResult()">=</button>
    </div>
</div>

<script>
    let currentNumber = '';
    let previousNumber = '';
    let operator = '';
    let resultCalculated = false; // Flag to track if result was calculated

    function appendNumber(number) {
        if (resultCalculated) {
            // If result was calculated, start a new calculation
            currentNumber = '';
            resultCalculated = false;
        }
        currentNumber += number;
        updateDisplay(currentNumber);
    }

    function setOperator(op) {
        if (currentNumber === '') return;
        if (previousNumber !== '') {
            calculateResult();
        }
        operator = op;
        previousNumber = currentNumber;
        currentNumber = '';
```

```
            resultCalculated = false; // Reset flag for new operation
        }

        function calculateResult() {
            if (currentNumber === '' || previousNumber === '' || operator === '') return;
            let result;
            const prev = parseFloat(previousNumber);
            const curr = parseFloat(currentNumber);

            switch (operator) {
                case '+':
                    result = prev + curr;
                    break;
                case '-':
                    result = prev - curr;
                    break;
                case '*':
                    result = prev * curr;
                    break;
                case '/':
                    result = curr !== 0 ? prev / curr : 'Error';
                    break;
                default:
                    return;
            }

            updateDisplay(result);
            currentNumber = result.toString();
            previousNumber = '';
            operator = '';
            resultCalculated = true; // Set flag to indicate result has been calculated
        }

        function clearDisplay() {
            currentNumber = '';
            previousNumber = '';
            operator = '';
            resultCalculated = false; // Reset flag
            updateDisplay('');
        }

        function updateDisplay(value) {
            document.getElementById('display').value = value;
        }
    </script>
</body>
</html>
```

Task 2: Creating an AWS Amplify application

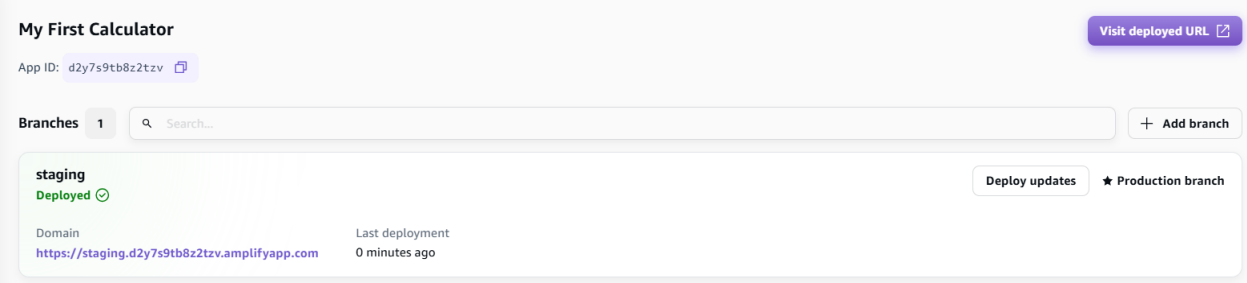
In this task, you will create an AWS Amplify application to host your website.

1. In the AWS Console search box to the right of **Services**, search for and choose **AWS Amplify** to open the **AWS Amplify** console.
2. Choose **Create new app**.
3. In the **Start building with Amplify** step, select the **Deploy without Git** option and press **Next.**
4. In the **Start a manual deployment** step, set the name of your application, i.e. *My First Calculator* and select **Choose .zip folder** option and select the calc.zip file you created.
5. You should see that the calc.zip file is uploaded successfully:

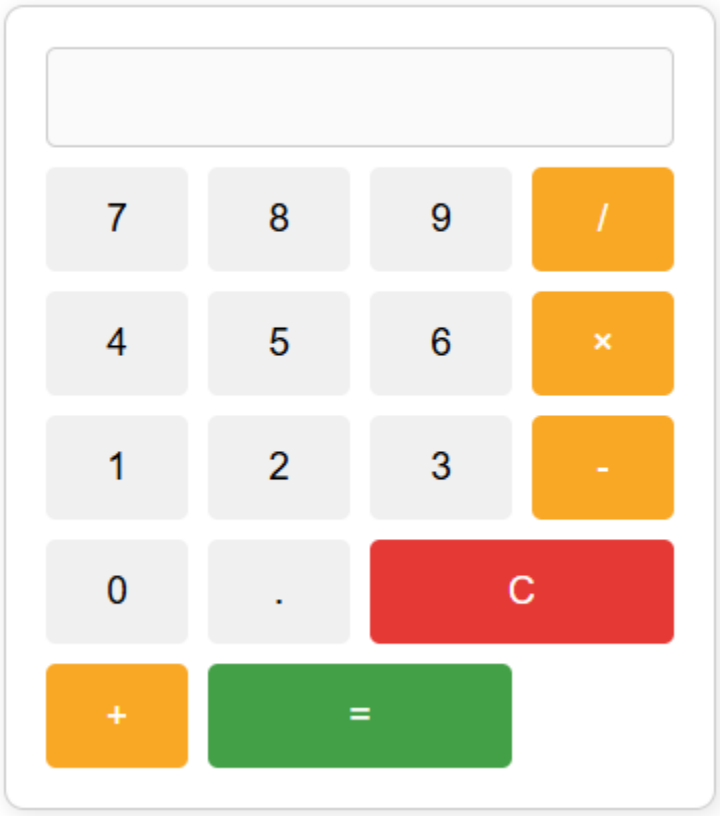


6. Press **Save and deploy** option.
   You application will now be deployed in a short process:

   Once done you will be navigated to your application page.



7. Click the **Visit Deployed URL** or the **Domain** link to launch your HTML page. Your HTML page will now be launched.



8. Open your application URL from any device such a mobile device to see it's also working on mobile.

Activity complete

Congratulations! You have successfully completed the activity.