Để triển khai một trang web quản lý sinh viên trên AWS sử dụng các dịch vụ Amplify, Lambda, API Gateway, DynamoDB và IAM, bạn có thể làm theo các bước sau:

1. Tạo một ứng dụng Amplify:
   * Đăng nhập vào bảng điều khiển AWS.
   * Mở Amplify Console và nhấp vào "Bắt đầu triển khai".
   * Chọn "Trang web" và chọn "Cài đặt liên kết tiếp" cho việc triển khai từ một repository Git hoặc "Bắt đầu từ scratch" để triển khai từ mã nguồn cục bộ.
   * Cấu hình các tùy chọn triển khai như tên ứng dụng, nhánh Git, môi trường triển khai và các cài đặt khác. Nhấp vào "Liên kết tiếp" để tiếp tục.
   * Amplify sẽ phát hiện mã nguồn của bạn và tạo một quy trình triển khai tự động. Xác nhận cấu hình và nhấp vào "Triển khai".
2. Tạo các chức năng Lambda:
   * Mở AWS Lambda Console.
   * Nhấp vào "Tạo chức năng" và chọn ngôn ngữ lập trình phù hợp với bạn.
   * Viết mã xử lý logic của các chức năng Lambda. Trong trường hợp này, bạn có thể có các chức năng như tạo, đọc, cập nhật và xóa sinh viên trong DynamoDB.
   * Lưu chức năng Lambda và ghi nhớ tên và ARN của chúng.
3. Tạo API Gateway:
   * Mở AWS API Gateway Console.
   * Tạo một API mới và đặt tên cho nó.
   * Tạo các tài liệu (Resources) và các phương thức (Methods) để quản lý các yêu cầu từ ứng dụng của bạn.
   * Liên kết các phương thức với các chức năng Lambda mà bạn đã tạo trước đó.
   * Lưu cấu hình API Gateway và ghi nhớ URL chính xác của API.
4. Tạo bảng DynamoDB:
   * Mở AWS DynamoDB Console.
   * Tạo một bảng mới và đặt tên cho nó, ví dụ: "Students".
   * Xác định các thuộc tính của bảng và thiết lập khóa chính.
   * Lưu bảng và ghi nhớ tên của nó.
5. Cấu hình quyền IAM:
   * Mở AWS IAM Console.
   * Tạo vai trò hoặc người dùng IAM để cung cấp quyền truy cập cho các dịch vụ Amplify, Lambda, API Gateway và DynamoDB.
   * Gán các chính sách IAM cho vai trò hoặc người dùng để cho phép truy cập, ghi và đọc dữ liệu từ các dịch vụ tương ứng.
   * Ghi nhớ ARN của vai trò hoặc người dùng IAM.
6. Cập nhật mã ứng dụng:
   * Trong mã ứng dụng của bạn, sửa đổi cấu hình để sử dụng các thông tin như ARN chức năng Lambda, URL API Gateway và thông tin bảng DynamoDB.
   * Tải lên lại mã ứng dụng lên kho lưu trữ mã nguồn hoặc cập nhật ứng dụng cục bộ của bạn.
7. Triển khai ứng dụng:
   * Quay lại Amplify Console.
   * Xác nhận cấu hình triển khai và nhấp vào "Triển khai".
   * Amplify sẽ tự động triển khai ứng dụng của bạn dựa trên cấu hình đã chỉ định.
   * Theo dõi tiến trình triển khai và kiểm tra ứng dụng của bạn khi nó hoàn thành.

Sau khi hoàn thành các bước trên, ứng dụng quản lý sinh viên của bạn sẽ được triển khai trên AWS sử dụng các dịch vụ Amplify, Lambda, API Gateway, DynamoDB và IAM.

CODE lambda:

// Import the AWS SDK

const AWS = require('aws-sdk');

// Create a DynamoDB document client

const docClient = new AWS.DynamoDB.DocumentClient();

// Define the Lambda handler function

exports.handler = async (event, context) => {

try {

// Extract the data from the event object

const { name, tx1, tx2 } = event;

// Calculate the average of the two numbers

const average = (parseFloat(tx1) + parseFloat(tx2)) / 2;

// Store the current time in a variable

const now = new Date().toISOString();

// Create a DynamoDB put operation to save the data and time

const params = {

TableName: 'student',

Item: {

ID: average.toString(),

Name: name,

TX1: parseFloat(tx1),

TX2: parseFloat(tx2),

Average: average,

LatestUpdateTime: now

}

};

// Save the data and time to the DynamoDB table

await docClient.put(params).promise();

// Return the result as a response

return {

statusCode: 200,

body: JSON.stringify(`The average is ${average}`)

};

} catch (error) {

// Handle any errors and return an error response

return {

statusCode: 500,

body: JSON.stringify('An error occurred')

};

}

};

///////

{

"Version": "2012-10-17",

"Statement": [

{

"Sid": "VisualEditor0",

"Effect": "Allow",

"Action": [

"dynamodb:PutItem",

"dynamodb:DeleteItem",

"dynamodb:GetItem",

"dynamodb:Scan",

"dynamodb:Query",

"dynamodb:UpdateItem"

],

"Resource": "YOUR-TABLE-ARN"

}

]

}