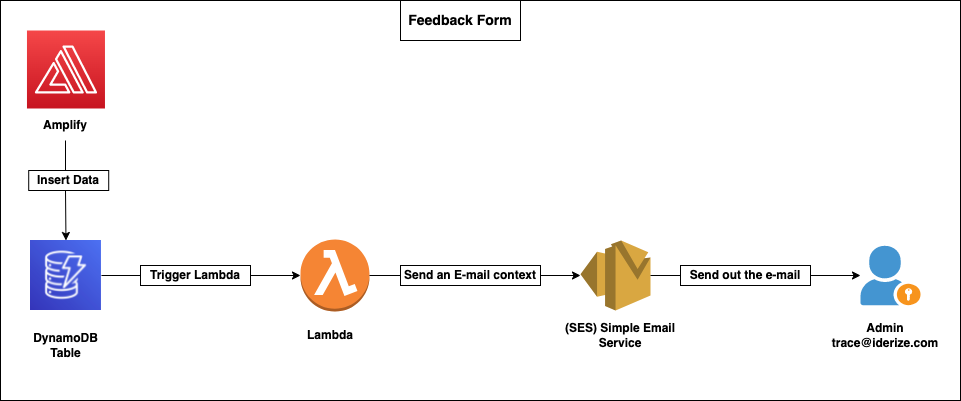
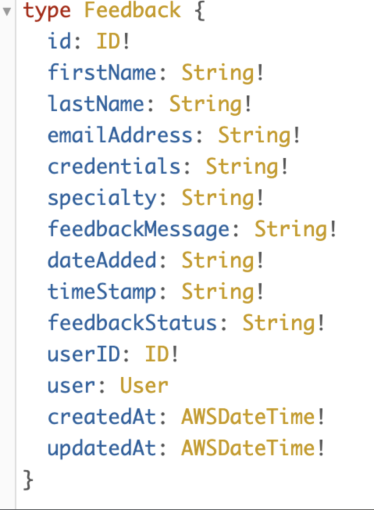
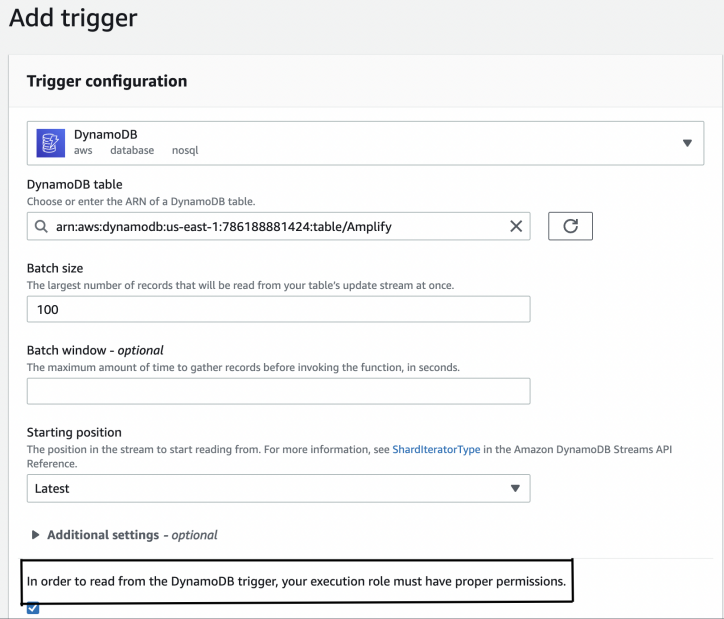
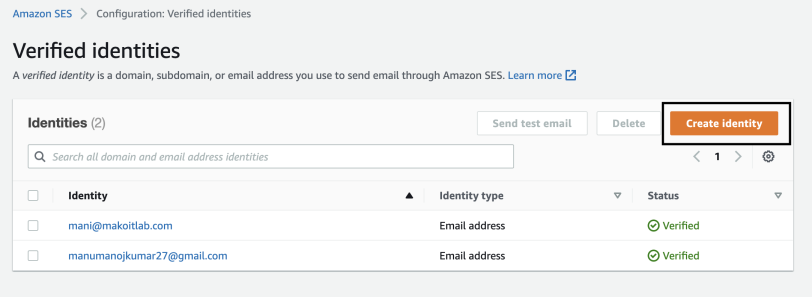
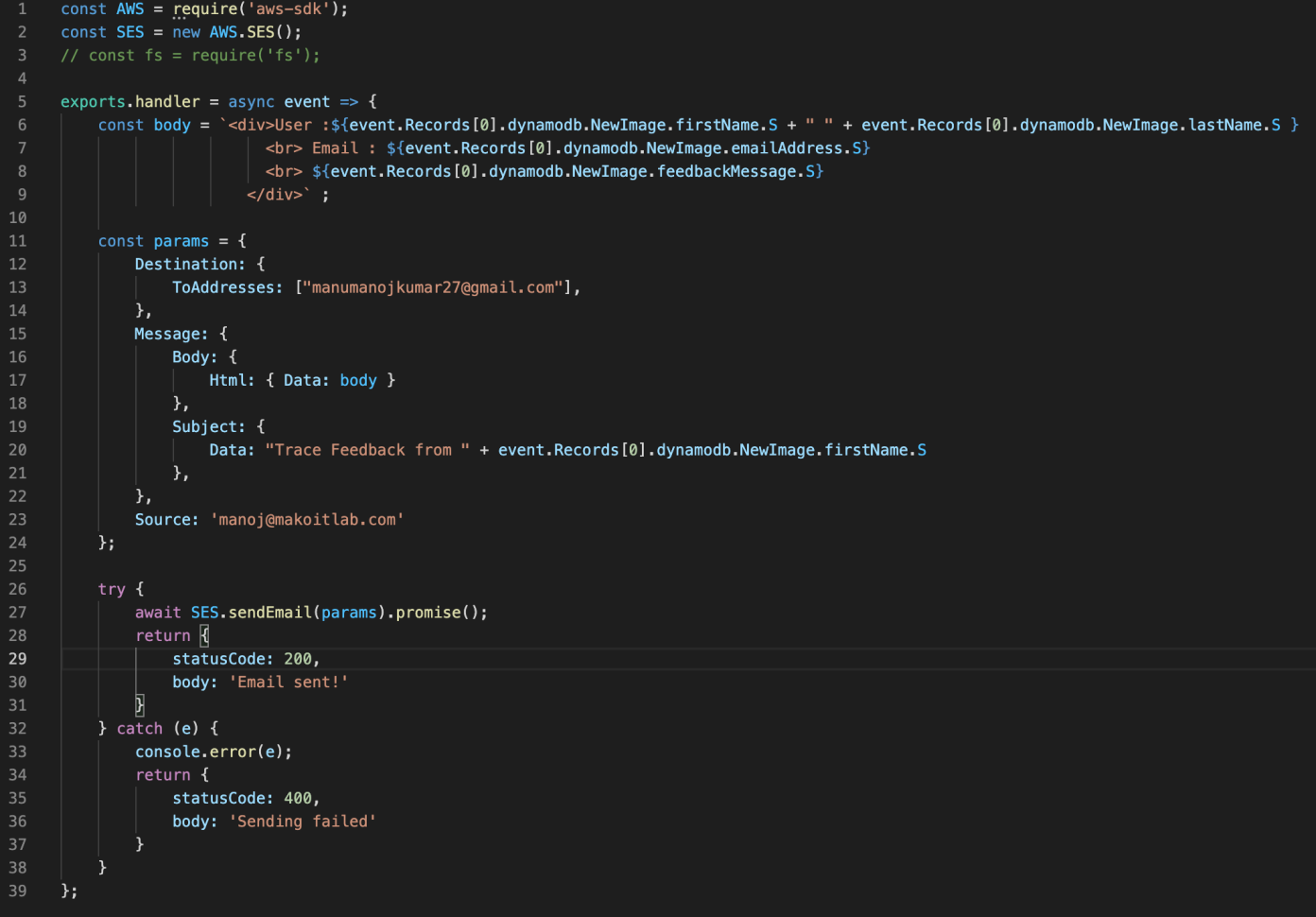
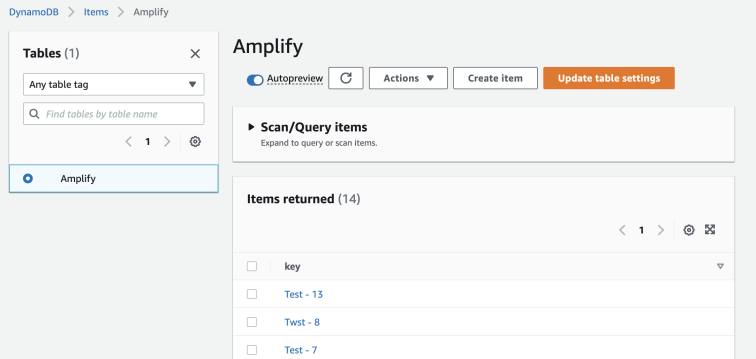
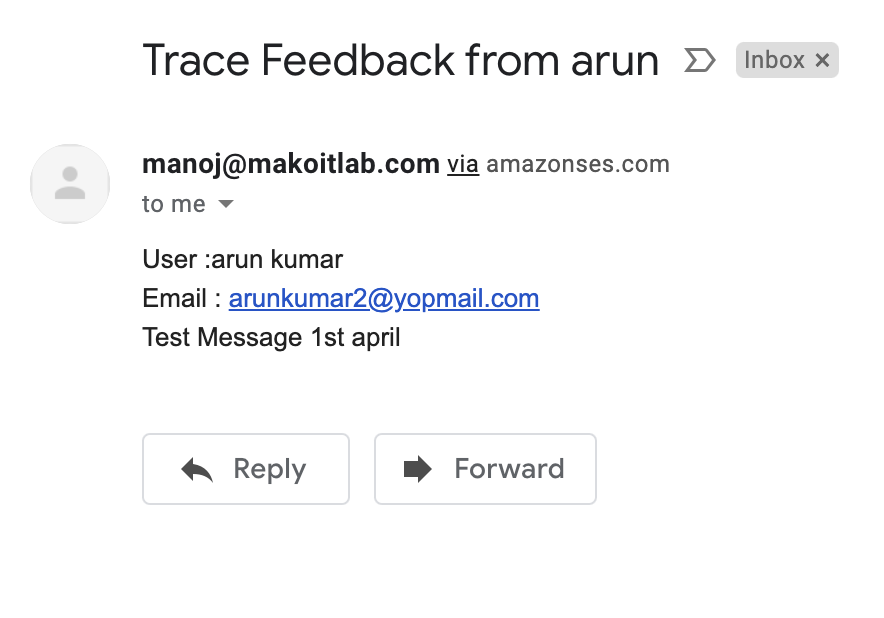
**Trace Architecture**



1. Analyze the AWS Amplify and AppSync setup. (Process of pushing data into the Dynamo-db Table).
2. Examine the format and schema of data. PF the schema below for the Feedback.
3. 
4. Spin up a lambda instance and provide LambdaExecutionRole with DynamoDBFullAceess to achieve integration.
5. Configure the trigger as Dynamo-db Table.
6. 
7. To achieve lambda integration with SES, provide SESFullAccess to LambdaExecutionRole.
8. Add identities in SES i.e., source and destination mail addresses.
9. 
10. Make sure to verify the mail address by confirming the URL in the mail which has been sent by SES.
11. Now hop on the lambda we have created, write logic to fetch the data from Dynamo Db Table and send the email context to SES.
12. Below is the logic :
13. 
14. Post that SES sends out the mail to the destination mail address (here admin)
15. Insert a sample data in the Dynamo-db table that triggers the lambda and sends out the mail.
16. 
17. You will receive the e-mail in few seconds.
18. 
19. Reference - 1) [Medium-Link](https://betterprogramming.pub/send-emails-serverlessly-with-node-js-lambda-and-aws-ses-186cba40d695) 2) [Blog](https://blog.focusotter.com/serverless-contact-form-using-aws-amplify)
20. Happy mailing!!!!!

**My take away points :**

1. Find full code in the /Trace-Amplify directory (for reference).
2. use this code snippet to check how lambda is getting the data.

exports.handler = async event => {

    console.log("DB-3", event.Records[0].dynamodb);

    return {

            statusCode: 200,

            body: 'Testing!'

        }

1. Run this snippet in lambda after configuring the Dyanamo-db trigger.
2. Quickly jump on the CloudWatch Logs and examine the log.
3. 
4. Based on the parameters and the required mail format construct the body,subject in the code by concatenating the required parameters.