

## 1. Overview

- No screenshots or observations

## 2. Create a Lambda execution role

- No screenshots or observations

## 3. Create a Cloud9 Environment

- No screenshots or observations

## 4. Obtain AWS account ID

- No screenshots or observations

## 5. REST API Code

**Answer the following questions:**

- **What might go wrong when we call `scan`? Think about the way DynamoDB works, and look at the [scan](#) documentation for a hint. What could be done to address this problem?**
  - A possible issue is the limited size of the dataset. If the scanned dataset exceeds the limit of 1 MB then the scan stops and is returned as a `LastEvaluatedKey`. A possible solution would be to have this run in a loop and check to make sure the entire table has been scanned through and not missing any data.

## 6. Deploy the Lambda for viewing entries

- No screenshots or observations

## 7. Create API in API Gateway

- No screenshots or observations

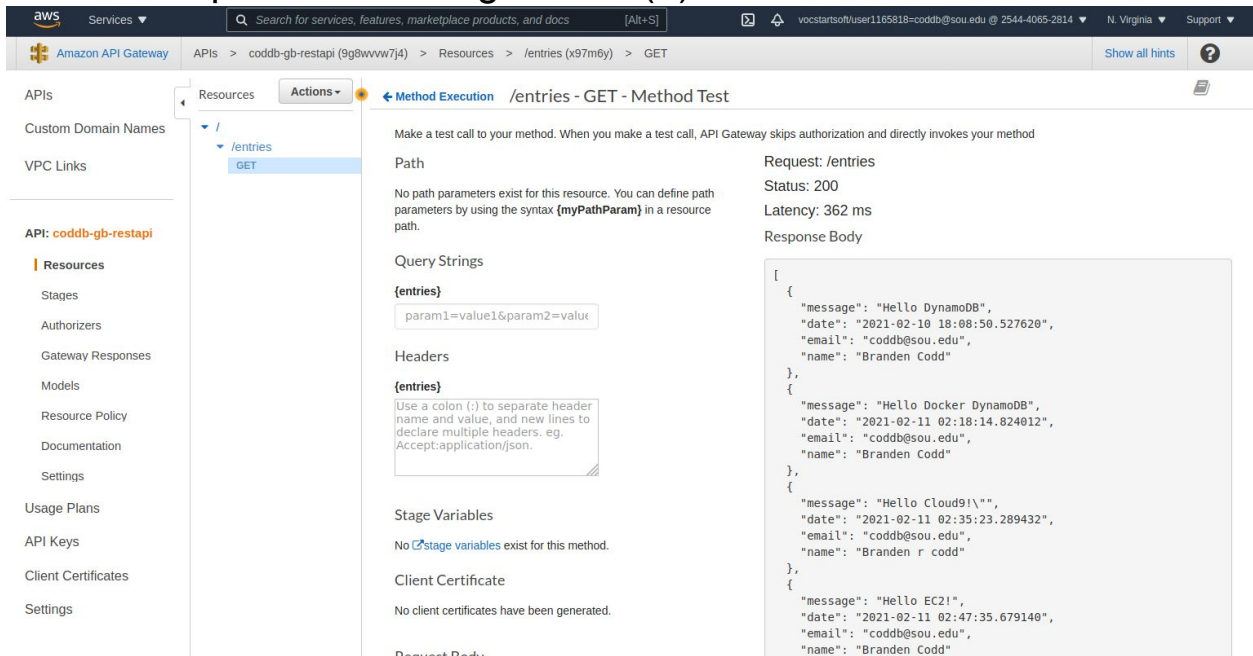
## 8. Enable API to invoke Lambda function

- No screenshots or observations

## 9. API endpoint for viewing entries (1)

- No screenshots or observations

## 10. API endpoint for viewing entries (2)



The screenshot shows the AWS API Gateway console for the API 'codd-db-restapi'. The 'Resources' section shows the path '/entries' with a 'GET' method. The 'Method Execution' page for '/entries - GET - Method Test' is displayed, showing the request path, status (200), latency (362 ms), and the response body. The response body is a JSON array of four entries, each with a message, date, email, and name.

API: codd-db-restapi

Resources

Stages

Authorizers

Gateway Responses

Models

Resource Policy

Documentation

Settings

Usage Plans

API Keys

Client Certificates

Settings

Resources

Actions

Method Execution /entries - GET - Method Test

Make a test call to your method. When you make a test call, API Gateway skips authorization and directly invokes your method.

Path

No path parameters exist for this resource. You can define path parameters by using the syntax {myPathParam} in a resource path.

Query Strings

{entries}

param1=value1&param2=value2

Headers

{entries}

Use a colon (:) to separate header name and value, and new lines to declare multiple headers. eg. Accept:application/json.

Stage Variables

No stage variables exist for this method.

Client Certificate

No client certificates have been generated.

Request

Request: /entries

Status: 200

Latency: 362 ms

Response Body

```
[
  {
    "message": "Hello DynamoDB",
    "date": "2021-02-10 18:08:50.527620",
    "email": "codd@sou.edu",
    "name": "Branden Codd"
  },
  {
    "message": "Hello Docker DynamoDB",
    "date": "2021-02-11 02:18:14.824812",
    "email": "codd@sou.edu",
    "name": "Branden Codd"
  },
  {
    "message": "Hello Cloud9!",
    "date": "2021-02-11 02:35:23.289432",
    "email": "codd@sou.edu",
    "name": "Branden r codd"
  },
  {
    "message": "Hello EC2!",
    "date": "2021-02-11 02:47:35.679140",
    "email": "codd@sou.edu",
    "name": "Branden Codd"
  }
]
```

## 11. CORS setup for viewing entries

Amazon API Gateway

APIs > coddgb-restapi (9g8www7j4) > Resources > /entries (x97m6y) > OPTIONS

Show all hints ?

APIs

Custom Domain Names

VPC Links

API: coddgb-restapi

Resources

Stages

Authorizers

Gateway Responses

Models

Resource Policy

Documentation

Settings

Usage Plans

API Keys

Client Certificates

Settings

Resources

Actions

Method Execution

/entries - OPTIONS - Method Test

Make a test call to your method. When you make a test call, API Gateway skips authorization and directly invokes your method

Path

No path parameters exist for this resource. You can define path parameters by using the syntax {myPathParam} in a resource path.

Query Strings

{entries}

param1=value1&param2=value1

Headers

{entries}

Use a colon (:) to separate header name and value, and new lines to declare multiple headers. eg. Accept:application/json.

Stage Variables

No stage variables exist for this method.

Client Certificate

No client certificates have been generated.

Request Body

1

Request: /entries

Status: 204

Latency: 3 ms

Response Body

no data

Response Headers

{ "Access-Control-Allow-Origin": "\*", "Access-Control-Allow-Methods": "GET,POST", "Access-Control-Allow-Headers": "Content-Type,X-Amz-Date,Authorization,X-Api-Key,X-Amz-Security-Token,X-Amz-User-Agent", "Content-Type": "application/json" }

Logs

Execution log for request 063ef9ee-ba10-41c9-b2ec-10999197b3fe

Thu Feb 25 01:34:32 UTC 2021 : Starting execution for request: 063ef9ee-ba10-41c9-b2ec-10999197b3fe

Thu Feb 25 01:34:32 UTC 2021 : HTTP Method: OPTIONS, Resource Path: /entries

Thu Feb 25 01:34:32 UTC 2021 : Method request path: {}

Thu Feb 25 01:34:32 UTC 2021 : Method request query string: {}

Thu Feb 25 01:34:32 UTC 2021 : Method request headers: {}

Thu Feb 25 01:34:32 UTC 2021 : Method request body before transformations: (null)

Thu Feb 25 01:34:32 UTC 2021 : Method response body after transformations: (null)

Thu Feb 25 01:34:32 UTC 2021 : Method response headers: {Access-Control-Allow-Headers=Content-Type,X-Amz-Date,Authorization,X-Api-Key,X-Amz-Security-Token,X-Amz-User-Agent, Access-Control-Allow-

## 12. Deploy API to production and view entries

AWS Account x Workbench x coddgb-lambda - Lam x API Gateway x guestbook - AWS Cloud9 x My Visitors x

file:///home/coddgb/cs356-cloud-files/06\_aws\_restapi\_lambda\_cdk/frontend/src/index.html

Guestbook

Name:

Email:

Message:

hello

Sign

Entries

Branden Codd <coddgb@sou.edu>  
signed on 2021-02-10 18:08:50.527620  
Hello DynamoDB

Branden Codd <coddgb@sou.edu>  
signed on 2021-02-11 02:18:14.824012  
Hello Docker DynamoDB

Branden r codd <coddgb@sou.edu>  
signed on 2021-02-11 02:35:23.289432  
Hello Cloud9!"

Branden Codd <coddgb@sou.edu>

## 13. API endpoint for signing (1)

- No screenshots or observations

## 14. API endpoint for signing (2)

Click on TEST, then view the response. The API returns the entire contents of the guestbook including the entry you have added.

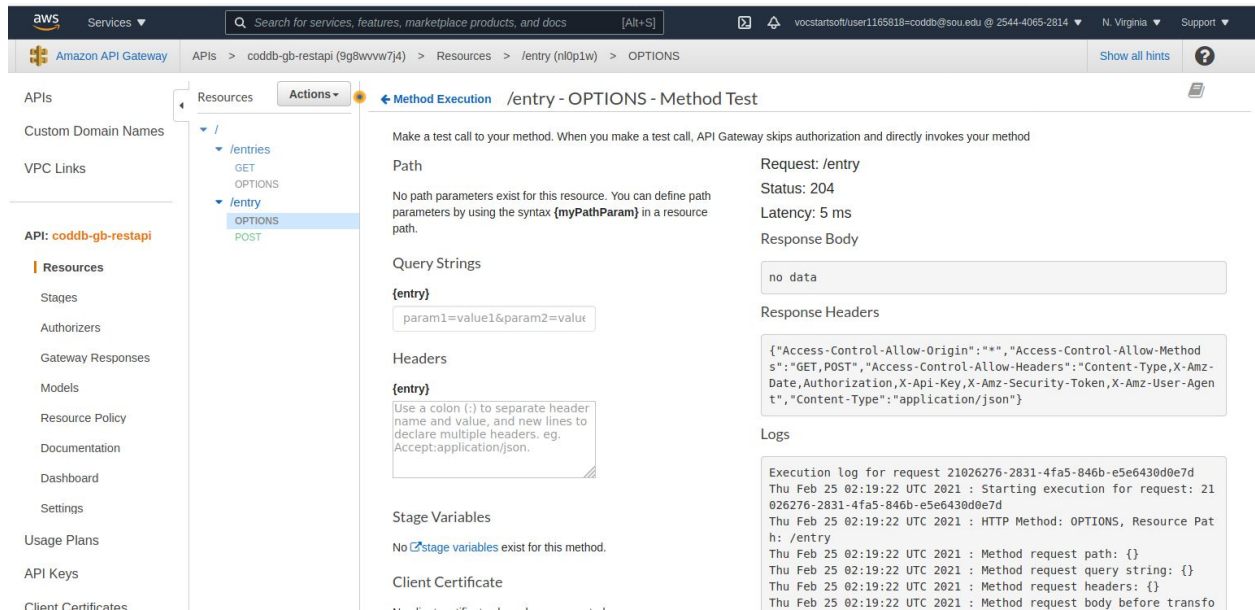
- Take a screenshot showing that the submission worked.

The screenshot displays the AWS API Gateway console interface. The left sidebar shows the navigation menu with 'APIs' selected. Under 'APIs', the 'Resources' section is expanded, showing a tree structure with '/' and '/entry'. The '/entry' resource is selected, and the 'POST' method is highlighted. The main panel shows the 'Method Execution' view for the '/entry - POST - Method Test'. It includes a 'Request' section with the path '/entry', status '200', latency '469 ms', and an empty response body. The 'Response Body' section displays a JSON array of five entries, each with 'message', 'date', 'email', and 'name' fields. The entries are: 'Hello DynamoDB', 'Hello Docker DynamoDB', 'Hello Cloud9!', 'Hello EC2!', and 'Hello Elastic Beanstalk!'. The 'Request Body' section shows a JSON object with 'name', 'email', and 'message' fields, which was used to create the new entry at the bottom of the list.

```
1 {
2   "name": "Branden Codd",
3   "email": "codd@sou.edu",
4   "message": "Hello API Gateway"
5 }
```

```
[
  {
    "message": "Hello DynamoDB",
    "date": "2021-02-10 18:08:50.527620",
    "email": "codd@sou.edu",
    "name": "Branden Codd"
  },
  {
    "message": "Hello Docker DynamoDB",
    "date": "2021-02-11 02:18:14.824012",
    "email": "codd@sou.edu",
    "name": "Branden Codd"
  },
  {
    "message": "Hello Cloud9!",
    "date": "2021-02-11 02:35:23.289432",
    "email": "codd@sou.edu",
    "name": "Branden r codd"
  },
  {
    "message": "Hello EC2!",
    "date": "2021-02-11 02:47:35.679140",
    "email": "codd@sou.edu",
    "name": "Branden Codd"
  },
  {
    "message": "Hello Elastic Beanstalk!",
    "date": "2021-02-16 05:26:59.307395",
    "email": "codd@sou.edu",
    "name": "Branden r codd"
  },
  {
    "message": "Hello ECS!",
    "date": "2021-02-17 01:37:13.463922",
    "email": "codd@sou.edu",
    "name": "Branden r codd"
  }
]
```

## 15. CORS setup for signing



## 16. Deploy API to production and sign

Branden Codd <coddgb@sou.edu>  
signed on 2021-02-25 02:16:08.830792  
Hello API Gateway

---

Branden Codd <coddgb@sou.edu>  
signed on 2021-02-25 02:34:37.561925  
hello API gateway from local HTML

---

Branden Codd <coddgb@sou.edu>  
signed on 2021-02-25 02:34:39.086334  
hello API gateway from local HTML

---

Branden Codd <coddgb@sou.edu>  
signed on 2021-02-25 02:34:43.285930  
hello API gateway from local HTML

---

## 17. Frontend Code

- No screenshots or observations

## 18. Configure and Deploy the Frontend

**Guestbook**

Name:

Email:

Message:  
Hello S3, API Gateway and Lambda!

**Entries**

Branden Codd <coddb@sou.edu>  
signed on 2021-02-10 18:08:50.527620  
Hello DynamoDB

---

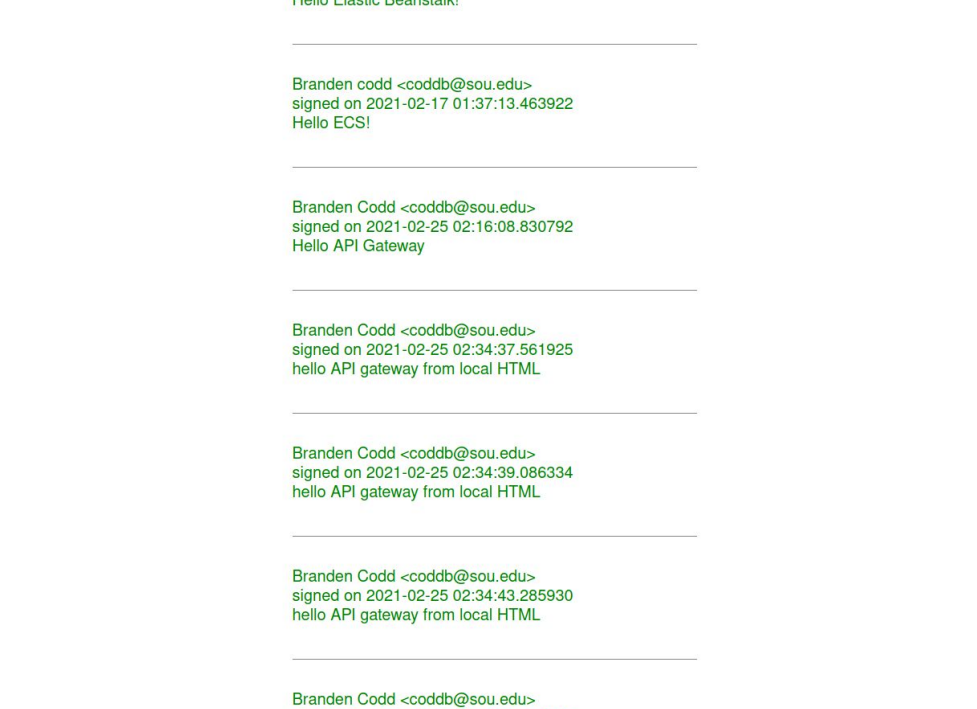
Branden Codd <coddb@sou.edu>  
signed on 2021-02-11 02:18:14.824012  
Hello Docker DynamoDB

---

Branden r codd <coddb@sou.edu>  
signed on 2021-02-11 02:35:23.289432  
Hello Cloud9!\*

---

Branden Codd <coddb@sou.edu>  
signed on 2021-02-11 02:47:35.679140  
Hello EC2!



The screenshot shows a web browser window with the address bar displaying 'codd-frontend.s3-website-us-east-1.amazonaws.com'. The page content consists of a series of green text messages, each preceded by a horizontal line. The messages are as follows:

- signed on 2021-02-16 05:26:59.307395  
Hello Elastic Beanstalk!
- Branden codd <codd@sou.edu>  
signed on 2021-02-17 01:37:13.463922  
Hello ECS!
- Branden Codd <codd@sou.edu>  
signed on 2021-02-25 02:16:08.830792  
Hello API Gateway
- Branden Codd <codd@sou.edu>  
signed on 2021-02-25 02:34:37.561925  
hello API gateway from local HTML
- Branden Codd <codd@sou.edu>  
signed on 2021-02-25 02:34:39.086334  
hello API gateway from local HTML
- Branden Codd <codd@sou.edu>  
signed on 2021-02-25 02:34:43.285930  
hello API gateway from local HTML
- Branden Codd <codd@sou.edu>  
signed on 2021-02-25 04:11:40.712477  
Hello S3, API Gateway and Lambda!