

LAB-2

- Creating IAM Role and policy for accessing Dynamo DB through API

The screenshot shows the AWS IAM console interface. The left sidebar contains navigation links: Dashboard, Groups, Users, Roles (selected), Policies, Identity providers, Account settings, Credential report, and Encryption keys. The main content area is titled 'IAM > Roles > cmpe273-dharmisha-dynamodb'. Under the 'Summary' tab, the following details are displayed:

- Role ARN:** arn:aws:iam::714785976764:role/cmpe273-dharmisha-dynamodb
- Role description:** Allows API Gateway to call AWS resources on your behalf.
- Instance Profile ARNs:** /
- Path:** /
- Creation time:** 2017-04-15 15:52 PDT

Below the summary, there are tabs for 'Permissions', 'Trust relationships', 'Access Advisor', and 'Revoke sessions'. The 'Permissions' tab is active, showing a section for 'Managed Policies'. It states: 'The following managed policies are attached to this role. You can attach up to 10 managed policies.' Below this is an 'Attach Policy' button and a table of attached policies:

Policy Name	Actions
AdministratorAccess	Show Policy Detach Policy Simulate Policy
policygen-201704151509-dynamodb	Show Policy Detach Policy Simulate Policy

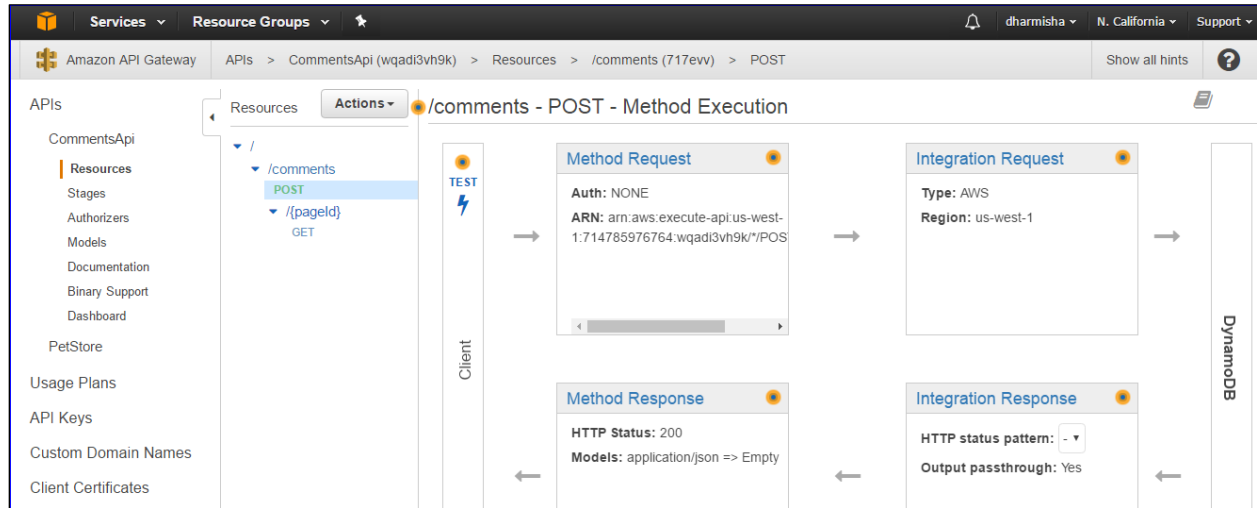
- Comments Table

The screenshot shows the AWS DynamoDB console interface. The left sidebar contains navigation links: Dashboard, Tables (selected), and Reserved capacity. The main content area is titled 'DynamoDB' and shows the 'Comments' table selected. The 'Create table' button is visible. The table details are displayed on the right side of the console:

Table details

Table name	Comments
Primary partition key	commentId (String)
Primary sort key	-
Time to live attribute	DISABLED Manage TTL
Table status	Active
Creation date	April 14, 2017 at 4:02:53 PM UTC-7
Provisioned read capacity units	5

- Resources created



- Post Integration Request

The screenshot shows the 'Integration Request' configuration page for the /comments POST method. The breadcrumb navigation is: 3vh9k) > Resources > /comments (717evw) > POST. The page title is '/comments - POST - Integration Request'. A note states: 'Provide information about the target backend that this method will call and whether the incoming request data should be modified.' The configuration fields are as follows:

- Integration type:** Radio buttons for Lambda Function, HTTP, Mock, and AWS Service (selected).
- AWS Region:** us-west-1 (with an edit icon).
- AWS Service:** DynamoDB (with an edit icon).
- AWS Subdomain:** (with an edit icon).
- HTTP method:** POST (with an edit icon).
- Action:** PutItem (with an edit icon).
- Execution role:** arn:aws:iam::714785976764:role/cmpe273-dharmisha-dynamodb (with an edit icon).

- Post Integration request: application/json template

Resources

Actions

Content-Type

application/json

+ Add mapping template

application/json

Generate template:

```
1 {
2   "TableName": "Comments",
3   "Item": {
4     "commentId": {
5       "S": "$context.requestId"
6     },
7     "pageId": {
8       "S": "$input.path('$.pageId')"
9     },
10    "userName": {
11      "S": "$input.path('$.userName')"
12    },
13    "message": {
14      "S": "$input.path('$.message')"
15    }
16  }
17 }
```

- Request Body for POST

Request Body

```
1 {
2   "pageId": "01",
3   "userName": "Dharmisha",
4   "message": "Lab 2 Completed successfully"
5 }
```

Test

- POST Logs

```

*****
*****7fabbb, X-Amz-Date=20170420T060007Z, x-amzn-apigateway-api-id=wqadi3vh9
k, Accept=application/json, User-Agent=AmazonAPIGateway_wqadi3vh9k, X-Amz-Security-Token
=AgoGb3JpZ2luEHOaCXVzLXdlc3QtMSKAAnV4aMIAm6woCCpa6BgFLOb1v68FZnBQX3xcHupMf5cQczFYtj/r09e
e8pUM7WJ1hf3NIGm4ewMnAXOIIBMhqAYB9RCwIDvQYtQx1LmFGIzayeuzoVftGBXn1mYhUP8R8t00D0iCxs5Xssy
6WdxL9N0AktEs+zSNyeGiDooGceWGN9qYIXpb/pftUgYWH9tI7LPCyX33IavGfEMCVWBpH7gccMA1Qa5j1YxC3Jo
czZxIEBmsbSH1wLUSJaORdtbNdEpZvzkYOGERkOTGYH0We0VZe1Ir/vQVLj3859hZVdox4IcgPcpvIjfhV8mUMN
W4+YxooWqnAoWge4XMSTq0KIqgQMIbxAAGgw3MTQ3ODU5NzY3NjQiDC/Ww1znM/yk9E3YhireApsorMQ5c0icN/S
fMOFPTjM/EOI24CS80f0iaaghXrsNAtpdV5MSPY2xnKxdHKu1BNTyPjM26EZrEWziadf6/utz9gRrohXe08p5nR
[TRUNCATED]
Thu Apr 20 06:00:07 UTC 2017 : Endpoint request body after transformations: {
  "TableName": "Comments",
  "Item": {
    "commentId": {
      "S": "test-invoke-request"
    },
    "pageId": {
      "S": "01"
    },
    "userName": {
      "S": "Dharmisha"
    },
    "message": {
      "S": "Lab 2 Completed successfully"
    }
  }
}
Thu Apr 20 06:00:07 UTC 2017 : Endpoint response body before transformations: {}
Thu Apr 20 06:00:07 UTC 2017 : Endpoint response headers: {Server=Server, Connection=kee
p-alive, x-amzn-RequestId=A0NBB3IRQHFLFK6GHL0QNUC6V3VV4KQNSO5AEMVJF66Q9ASUAAJG, x-amz-cr
c32=2745614147, Content-Length=2, Date=Thu, 20 Apr 2017 06:00:07 GMT, Content-Type=appli
cation/x-amz-json-1.0}
Thu Apr 20 06:00:07 UTC 2017 : Method response body after transformations: {}
Thu Apr 20 06:00:07 UTC 2017 : Method response headers: {X-Amzn-Trace-Id=Root=1-58f84e67
-7b6c8dc948c78567ed3f4af9, Content-Type=application/json}
Thu Apr 20 06:00:07 UTC 2017 : Successfully completed execution
Thu Apr 20 06:00:07 UTC 2017 : Method completed with status: 200

```

- Post inserted in dynamo table

dharmisha					
Comments Close					
Overview Items Metrics Alarms Capacity Indexes Access control Tags					
Create item Actions					
Scan: [Table] Comments: commentId					
<input checked="" type="checkbox"/>	commentId	message	pageId	userName	
<input checked="" type="checkbox"/>	test-invoke-requ	Lab 2 Completed successfully	01	Dharmisha	

- GET: Integration Request

Resources

Actions

Method Execution

/comments/{pageId} - GET - Integration Request

Provide information about the target backend that this method will call and whether the incoming request data should be modified.

Integration type

☐ Lambda Function ⓘ

☐ HTTP ⓘ

☐ Mock ⓘ

☒ AWS Service ⓘ

AWS Region

us-west-1 ⓘ

AWS Service

DynamoDB ⓘ

AWS Subdomain

ⓘ

HTTP method

POST ⓘ

Action

Query ⓘ

Execution role

arn:aws:iam::714785976764:role/cmpe273-dharmisha-dynamodb ⓘ

Credentials cache

Do not add caller credentials to cache key ⓘ

Content Handling

Passthrough ⓘ ⓘ

- GET: Mapping Template

application/json

Generate template:

1

{

2

"TableName": "Comments",

3

"IndexName": "pageId-index",

4

"KeyConditionExpression": "pageId = :v1",

5

"ExpressionAttributeValues": {

6

":v1": {

7

"S": "\${input.params('pageId')}"

8

}

9

}

10

}

- GET: Test

9k) > Resources > /comments/{pageld} (qz3cla) > GET

Show all hints?

Method Execution

/comments/{pageld} - GET - Method Test

Make a test call to your method with the provided input

Path

{pageld}

Query Strings

No query string parameters exist for this method. You can add them via Method Request.

Headers

No header parameters exist for this method. You can add them via Method Request.

Stage Variables

No [stage variables](#) exist for this method.

Request: /comments/01

Status: 200

Latency: 96 ms

Response Body

```
{  "comments": [    {      "commentId": "test-invoke-request",      "userName": "Dharmisha",      "message": "Lab 2 Completed successfully"    }  ]}
```

Response Headers

```
{"X-Amzn-Trace-Id": "Root=1-58f8620b-65d9fbbcd88128f4ffe0969", "Content-Type": "application/json"}
```

- GET: Logs

```

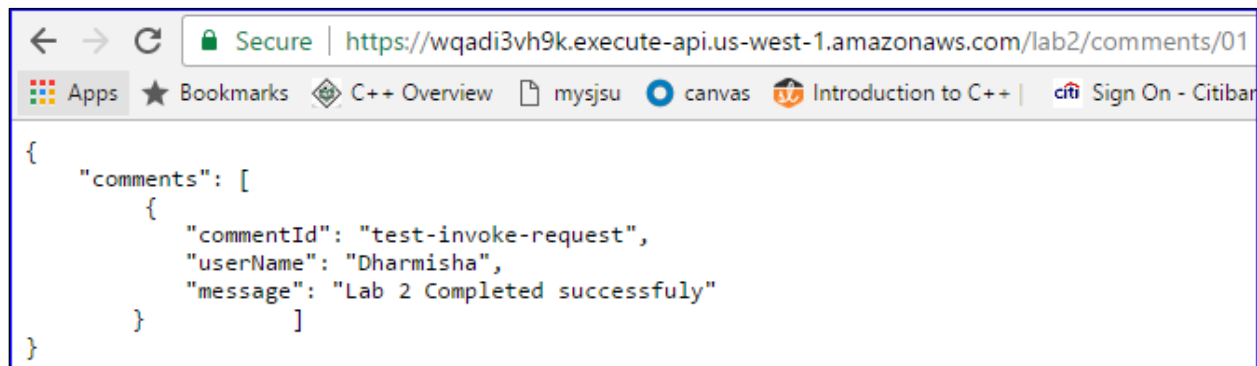
Thu Apr 20 07:23:55 UTC 2017 : Endpoint request body after transformations: {
  "TableName": "Comments",
  "IndexName": "pageId-index",
  "KeyConditionExpression": "pageId = :v1",
  "ExpressionAttributeValues": {
    ":v1": {
      "S": "01"
    }
  }
}
Thu Apr 20 07:23:55 UTC 2017 : Endpoint response body before transformations: {"Count":
1,"Items":[{"userName":{"S":"Dharmisha"},"message":{"S":"Lab 2 Completed successful
y"},"commentId":{"S":"test-invoke-request"},"pageId":{"S":"01"}}],"ScannedCount":1}
Thu Apr 20 07:23:55 UTC 2017 : Endpoint response headers: {Server=Server, Connection=kee
p-alive, x-amzn-RequestId=I6QN67SDP15I4E3J3TF35H4S83VV4KQNS05AEMVJF66Q9ASUAAJG, x-amz-cr
c32=684523606, Content-Length=175, Date=Thu, 20 Apr 2017 07:23:55 GMT, Content-Type=appl
ication/x-amz-json-1.0}
Thu Apr 20 07:23:55 UTC 2017 : Method response body after transformations: {
  "comments": [
    {
      "commentId": "test-invoke-request",
      "userName": "Dharmisha",
      "message": "Lab 2 Completed successfully"
    }
  ]
}
Thu Apr 20 07:23:55 UTC 2017 : Method response headers: {X-Amzn-Trace-Id=Root=1-58f8620b
-65d9fbbcd88128f4ffe0969, Content-Type=application/json}
Thu Apr 20 07:23:55 UTC 2017 : Successfully completed execution
Thu Apr 20 07:23:55 UTC 2017 : Method completed with status: 200

```

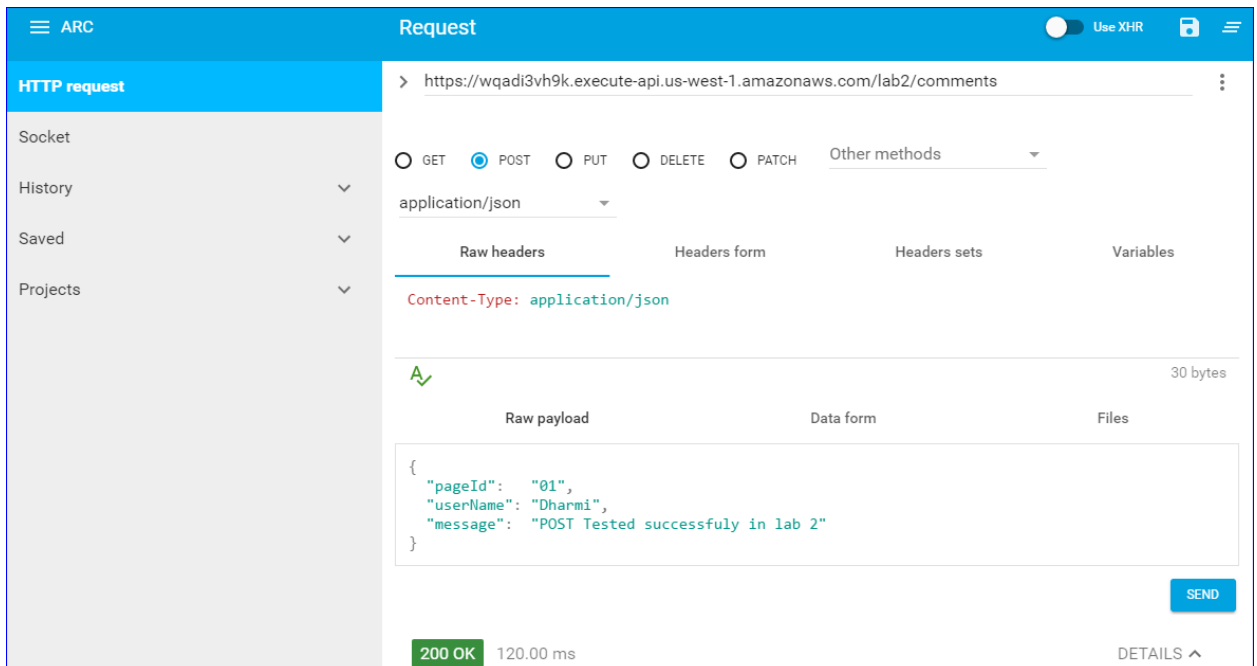
- After Deploying GET



- After accessing the URL and substituting '01' as the pageID, following output is obtained



- POST using advanced REST Client



- Executing GET after POST which shows POST was successful

```

{
  "comments": [
    {
      "commentId": "test-invoke-request",
      "userName": "Dharmisha",
      "message": "Lab 2 Completed successfully"
    },
    {
      "commentId": "b693e1ab-259c-11e7-8699-cd7b40baabf1",
      "userName": "Dharmi",
      "message": "POST Tested successfully in lab 2"
    }
  ]
}

```

- Table View after POST

commentId	message	pagelId	userName
test-invoke-requ	Lab 2 Completed successfully	01	Dharmisha
b693e1ab-259c-	POST Tested successfully in lab 2	01	Dharmi
72b2dd92-259c-	POST Tested successfully in lab 2	02	Dharmi