

LAB 2

AWS DynamoDB:

The screenshot shows the AWS Management Console interface for a DynamoDB table named 'Comments'. The table has a primary key 'commentId' and a secondary index 'message'. The table contains 9 items, including test messages and a real-world example of a comment.

commentId	message	pageId	userName
test-invoke-request	I really enjoyed this story!!	breaking-news-story-01-18-2016	Just Saying Thank You
3cf714c-2557-11e7-b597-0df4a9fa43ce	testMessage	testPageId	testUserName
ac852a1f-2557-11e7-972e-69d6aab3558a	testMessage1	testPageId1	testUserName1
0d34a8bc-259c-11e7-be72-a30a7301d843	I really enjoyed this story!!	breaking-news-story-01-18-2016	Just Saying Thank You
987553e6-2565-11e7-bced-0700e482fa4c	This is an example comment to be added.	sample-page-id	sample-UserName
667213bf-259b-11e7-9798-7ff6c93be4a	hello world	breaking news	welcome
2f97b7f1-259b-11e7-b85a-952d99c29bcd	hello world	breaking news	welcome
56a5fad8-2557-11e7-970b-2b7aeb2eed97	testMessage	testPageId	testUserName
b83614d7-259b-11e7-94dd-bf8a71002518	I really enjoyed this story!!	breaking-news-story-01-18-2016	Just Saying Thank You

POST: Invoke URL: <https://n7n790bbse.execute-api.us-west-2.amazonaws.com/dev/comments>

The screenshot shows the n7n (Node-RED) interface. A POST request is configured to the endpoint `https://n7n790bbse.execute-api.us-west-2.amazonaws.com/dev/comments. The request body is a JSON object with the following fields:`

```
1 {
2   "pageId": "breaking-news-story-01-18-2016",
3   "userName": "Just Saying Thank You",
4   "message": "I really enjoyed this story!!"
5 }
```

The status of the request is 200 OK, and the time taken is 137 ms.

GET: Invoke URL: <https://n7n790bbse.execute-api.us-west-2.amazonaws.com/dev/comments/{pagelId}>

GET request for pagelId = "breaking-news-story-01-18-2016"

The screenshot shows the Postman application interface. The top bar includes tabs for Runner, Import, Builder, and Team Library. The main panel is set to the 'GET' method for the URL `https://n7n790bbse.execute-api.us-west-2.amazonaws.com/dev/...`. The 'Authorization' tab is selected, showing 'No Auth'. The 'Body' tab is also selected, displaying a JSON response in 'Pretty' format. The response status is '200 OK' and the time taken is '200 ms'. The JSON body contains an array of comments:

```
1 {
2   "comments": [
3     {
4       "commentId": "test-invoke-request",
5       "userName": "Just Saying Thank You",
6       "message": "I really enjoyed this story!!"
7     },
8     {
9       "commentId": "0d34a8bc-259c-11e7-be72-a30a7301d843",
10      "userName": "Just Saying Thank You",
11      "message": "I really enjoyed this story!!"
12    },
13    {
14      "commentId": "b83614d7-259b-11e7-94dd-bf8a71002518",
15      "userName": "Just Saying Thank You",
16      "message": "I really enjoyed this story!!"
17    }
18  ]
19 }
```

GET request from browser for pagelId = "testPagelId"

The screenshot shows a web browser window with the address bar displaying `https://n7n790bbse.execute-api.us-west-2.amazonaws.com/dev/comments/testPagelId`. The page content displays a JSON response:

```
{
  "comments": [
    {
      "commentId": "3cfe714c-2557-11e7-b597-0df4a9fa43ce",
      "userName": "testUserName",
      "message": "testMessage"
    },
    {
      "commentId": "56a5fad8-2557-11e7-970b-2b7aeb2eed97",
      "userName": "testUserName",
      "message": "testMessage"
    }
  ]
}
```