Serverless Development 101

**HOP08 – Build an app with REST API - READ**

5/15/2020 Created by Apiwat Chuaphan

Center for Information Assurance (CIAE) @City University of Seattle (CityU)

**Learning Outcomes**

* Learn what REST API is and how it works
* Learn basic CRUD
* Learn how items store in DynamoDB
* Build an app with AWS Lambda, API Gateway, DynamoDB, REST API

**Build the R from CRUD paradigm.**

1. Open the VSCode and open the “**myproject**” project folder that we developed a web application using serverless computing in the previous module.
2. Go to **todos** folder and get inside **functions** folder.
3. Create a new file named **read.js** to read all of our data in the database. Then update the file with the content [here](https://bit.ly/3g9ocw4)

A screenshot of a social media post

Description automatically generated

**scan(params = {}, callback)**

* The Scan operation returns one or more items and item attributes by accessing every item in a table or a secondary index.

**Note**: This is very inefficient in large scale production apps. You might want to consider the query operation in a real-world scenario.

1. After we added new function, we need to let Lambda know our new function by updating **serverless.yml** as follow

A close up of text on a black background

Description automatically generated

1. Run **sls deploy -v** in terminal to deploy our function to Lambda. This time we have to deploy the entire process because we updated **serverless.yml** (*Make sure you’re in the* ***“todos” f****older)*
2. Head to AWS Lambda page and hit refresh to see that our new function has been deployed.

A screenshot of a cell phone

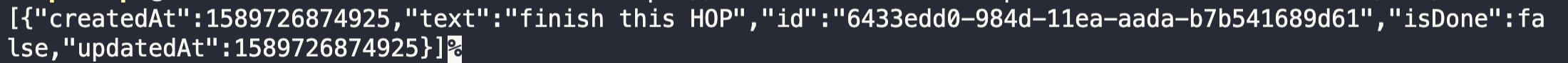
Description automatically generated

1. Verify if the function does what we expect it to do. We expect to get JSON-like text output back as a result.

Run this command in the terminal

>>> **curl -X GET <your-api-url>**

We should see the output similar to this, which is the todo task we added in the last module. Looks familiar? Of course!



Here is another way to verify with Postman. Click **New > Request > Save to your choice of collection.** After that, change HTTP method to GET.

A screenshot of a video game

Description automatically generated

This is also a way to test it, click on the function in the Lambda page > click **Test** button > name your test and leave the format as default > click **Create** button > then click **Test** button again.

This would be the result of the test.

A screenshot of a social media post

Description automatically generated

**Push your work to GitHub**

Open the terminal from the VSCode by hit the control + ~ key and type the following command:

Run the following commands to push your work to the GitHub repository:

>>> git add .

>>> git commit -m “Submission for Module 8”

>>> git push origin YOUR\_BRANCH\_NAME

**Note**: you should change the YOUR\_BRANCH\_NAME to your own branch name. It should be firstname-lastname (e.g. maria-gracia).

If you cannot remember, run the command “git status” to check