Swinburne University of Technology

COS20019 Cloud Computing Architecture

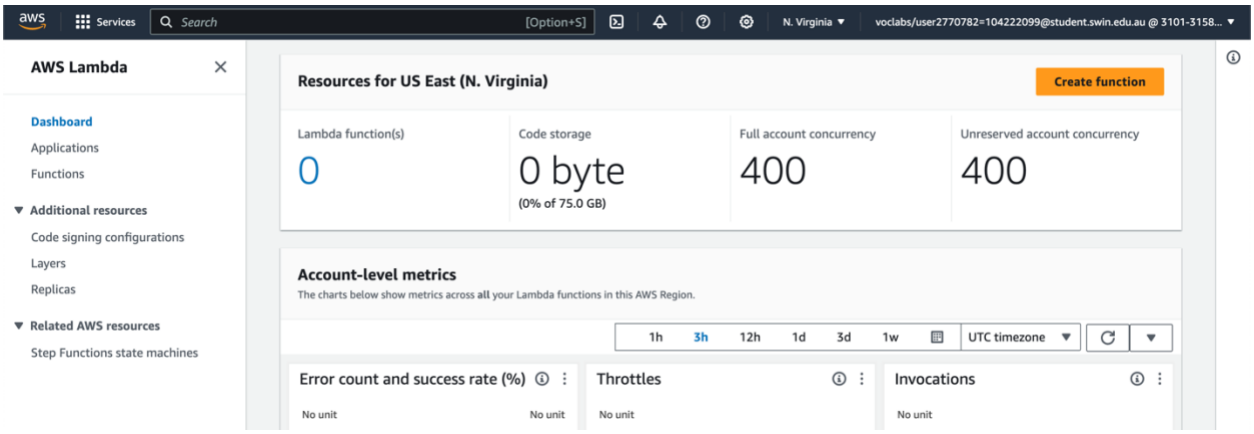
Module 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

---

Saturday 11th October, 2023

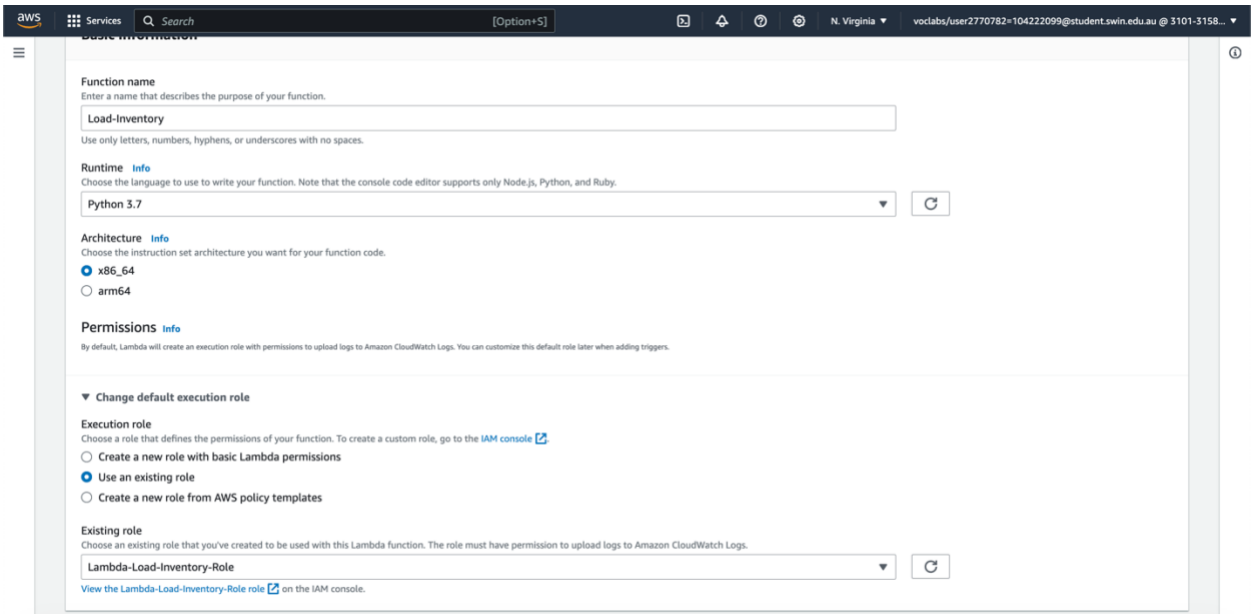
Task 1: Creating a Lambda function to load data

On the **AWS Management Console**, on the Services menu, choose **Lambda**.
Choose **Create function**

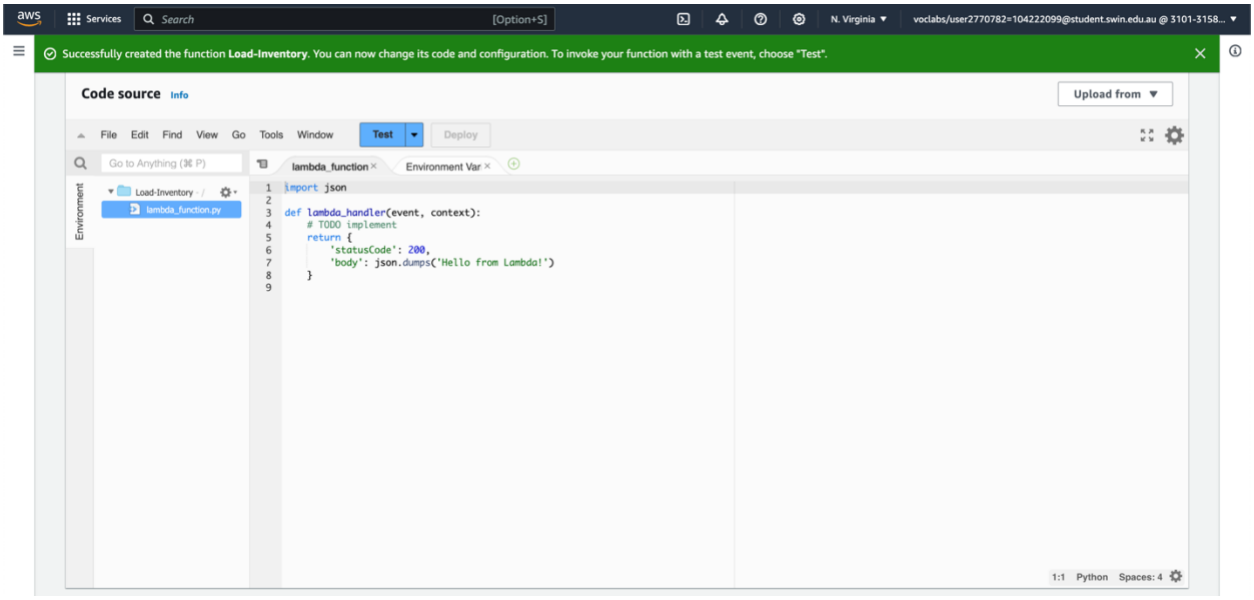


Configure the following settings:

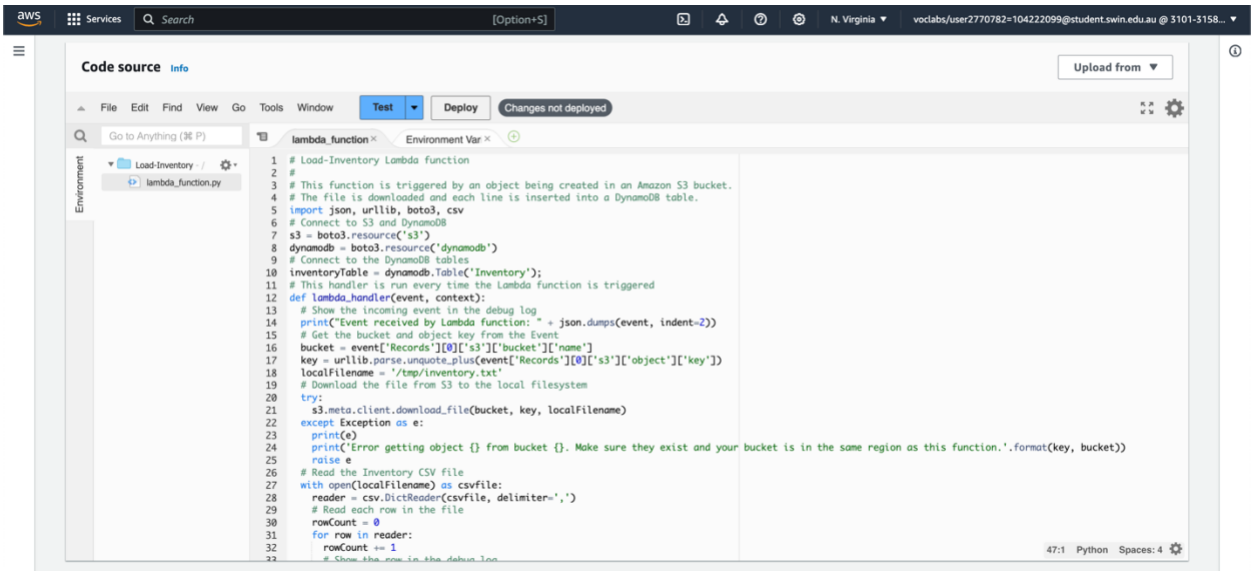
- **Function name:** Load-Inventory
- **Runtime:** *Python 3.7*
- Expand **Choose or create an execution role**.
- **Execution role:** *Use an existing role*
- **Existing role:** *Lambda-Load-Inventory-Role*



Scroll down to the **Code source** section, and in the **Environment** pane, choose lambda_function.py.



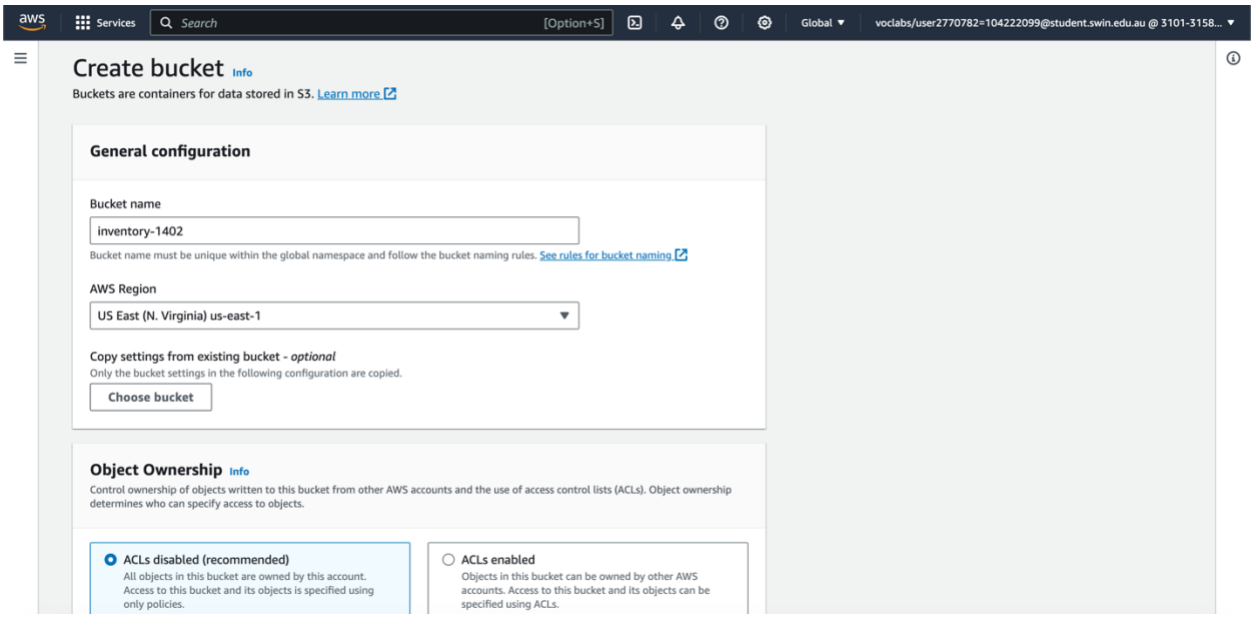
In the code editor, delete all the code.
In the **Code source** editor, copy and paste the following code:



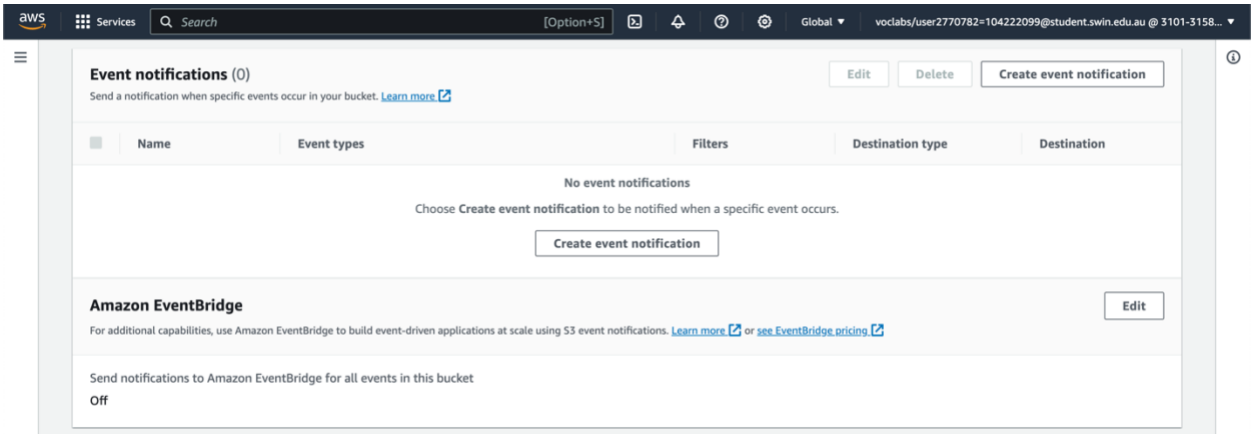
Choose **Deploy** to save your changes.

Task 2: Configuring an Amazon S3 event

On the **Services** menu, choose **S3**.
Choose **Create bucket**
For **Bucket name** enter: inventory-<number> (Replace with a random number)



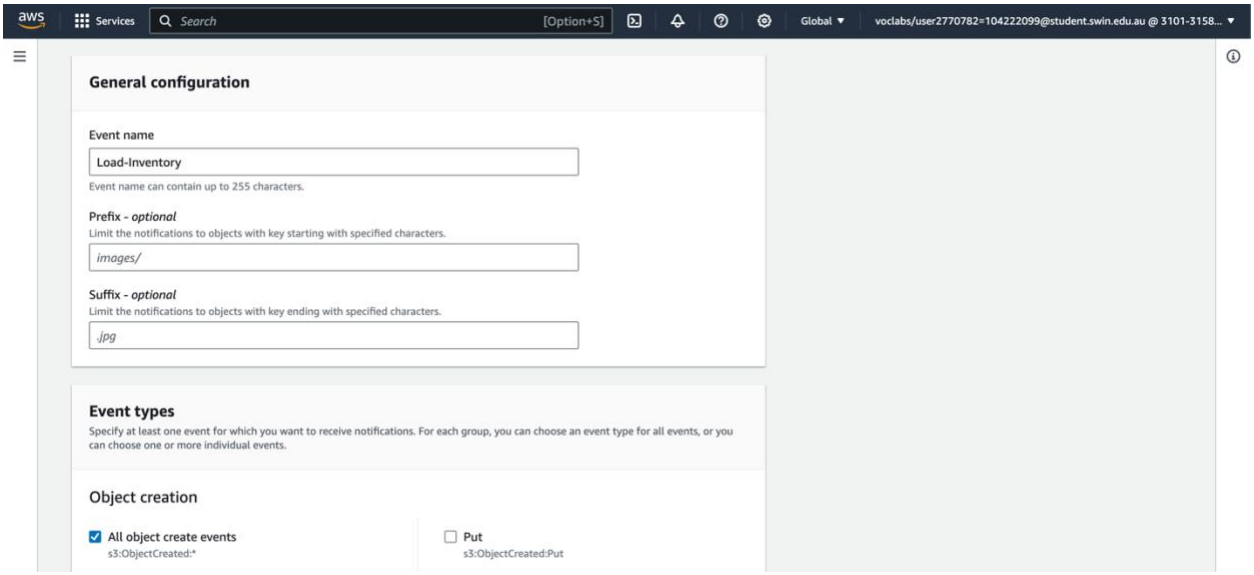
Choose the name of your *inventory*- bucket.
Choose the **Properties** tab.
Scroll down to **Event notifications**.

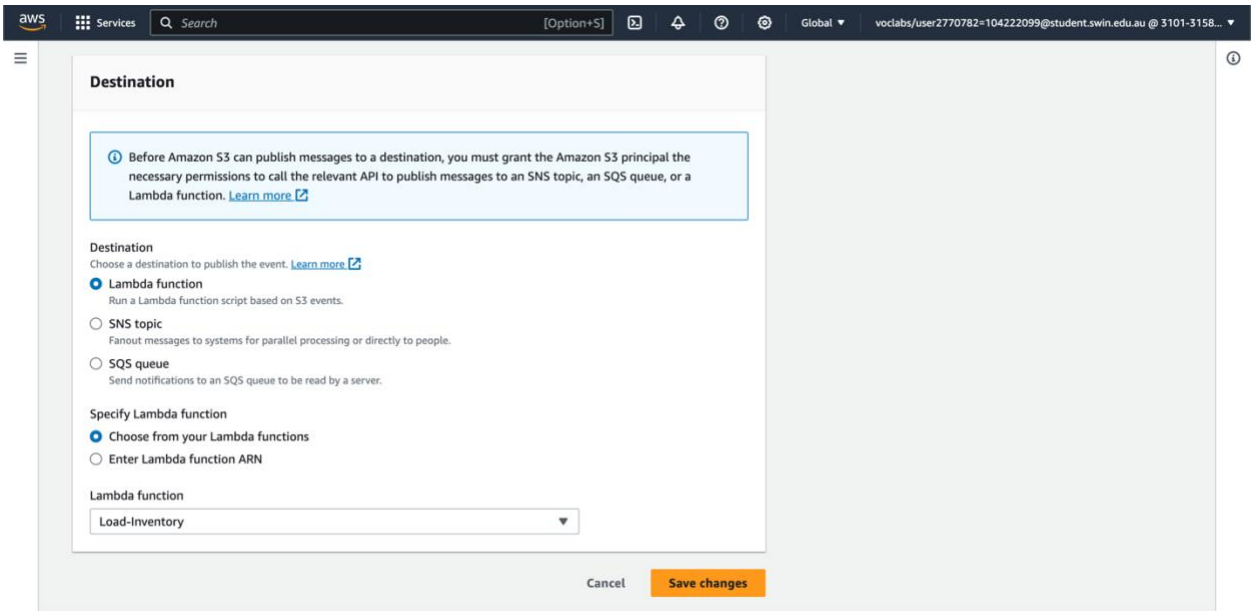


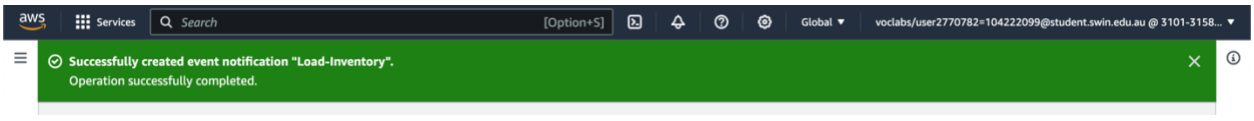
Click **Create event notification** then configure these settings:

- **Name:** Load-Inventory
- **Event types:** *All object create events*

- **Destination:** *Lambda Function*
- **Lambda function:** *Load-Inventory*
- Choose **Save changes**





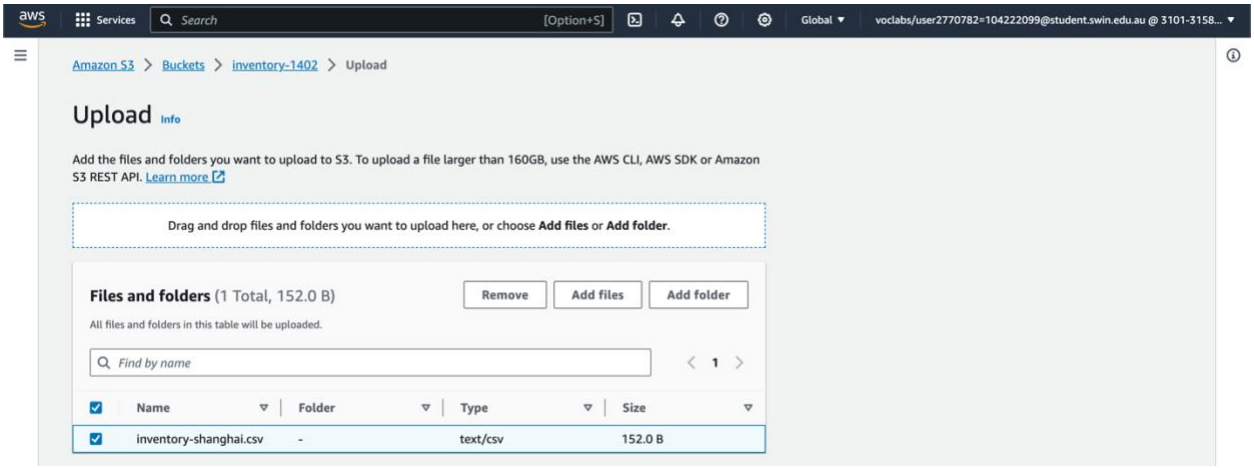


Task 3: Testing the loading process

Download the inventory files by opening the context (right-click) menu for these links:

inventory-berlin.csv
inventory-calcutta.csv
inventory-karachi.csv
inventory-pusan.csv
inventory-shanghai.csv
inventory-springfield.csv

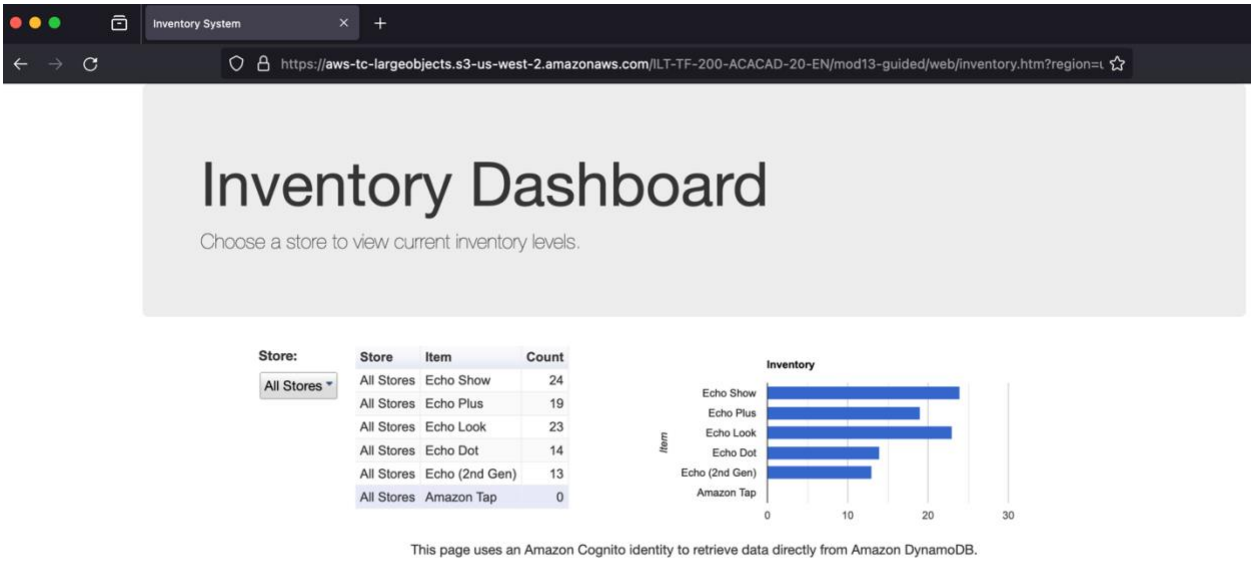
Return to your S3 bucket in the console by choosing the **Objects** tab.
Choose **Upload** > **Add files**, and select one of the inventory CSV files (You can choose any inventory file) > **Upload**



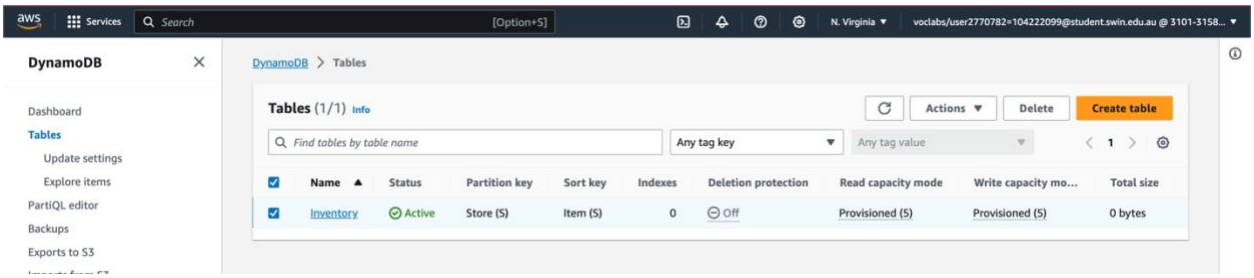
At the top of these instructions, choose the **Details** button; to the right of **AWS**, choose the **Show** button.

SecretKey	1QwWWz+NFmuNg8aEthYEzG+/ofnFPpR+5EzHUgYz
Dashboard	https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/ILT-TF-200-ACACAD-20-EN/mod13-guided/web/inventory.htm?region=us-east-1&poolId=us-east-1:e0147493-fa17-4431-9dcd-b9935b07eaae
IdentityPoolId	us-east-1:e0147493-fa17-4431-9dcd-b9935b07eaae
AccessKey	AKIAUQNKIO5AWDLBMI4Z

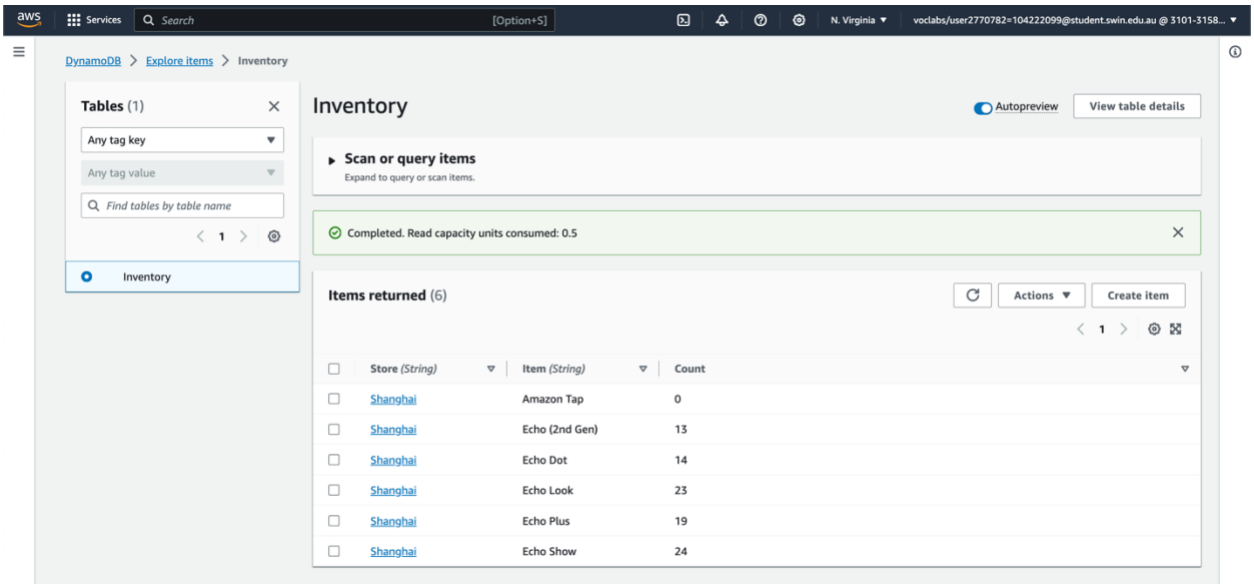
From the **Credentials** window, copy the **Dashboard** URL.
Open a new web browser tab, paste the URL, and press ENTER.



On the **Services** menu, choose **DynamoDB**.
In the left navigation pane, choose **Tables**.
Choose the **Inventory** table.

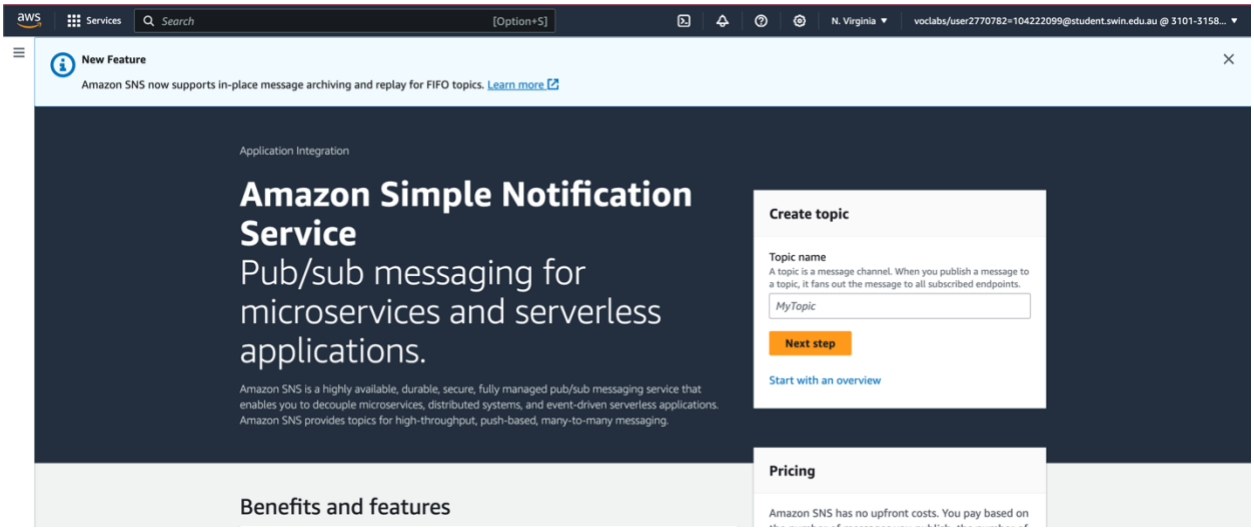


Choose the **Items** tab.

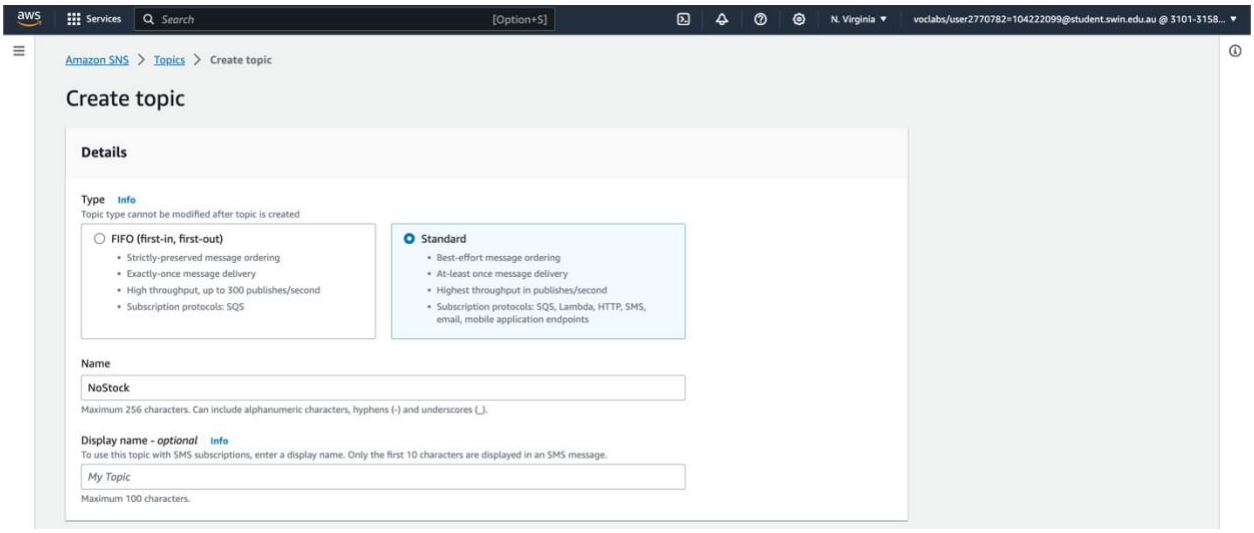


Task 4: Configuring notifications

On the **Services** menu, choose **Simple Notification Service**.

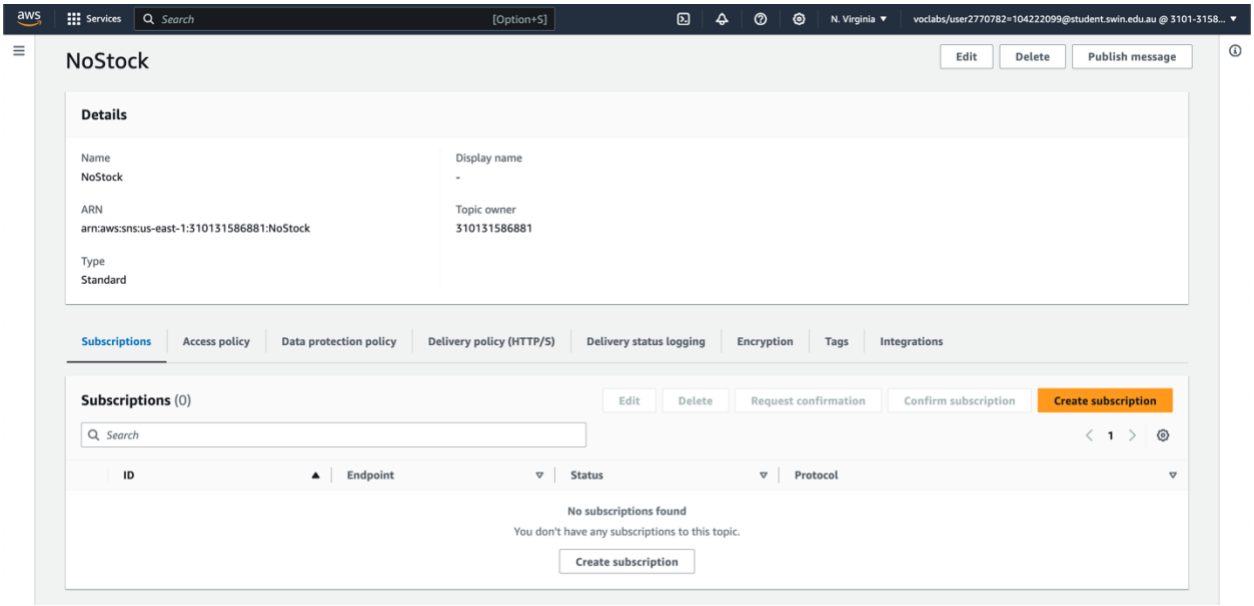


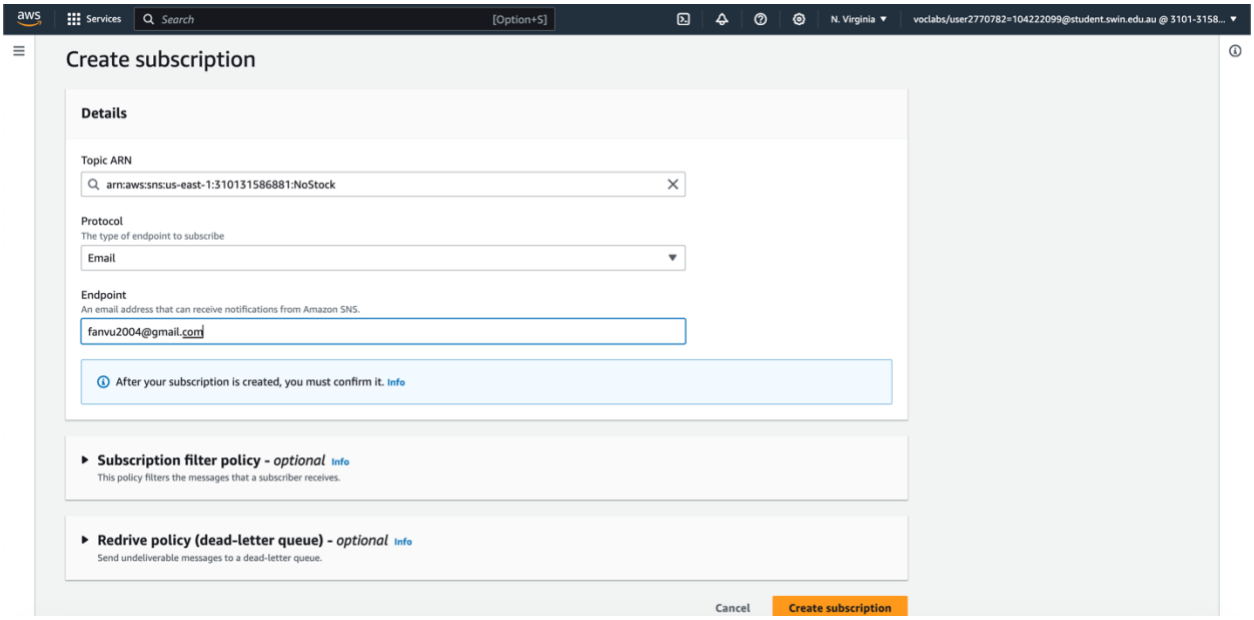
In the **Create topic** box, for **Topic name**, enter: NoStock. Keep **Standard** selected.
Choose **Create topic**

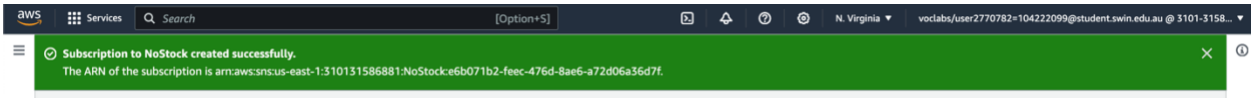


In the lower half of the page, choose **Create subscription** and configure these settings:

- **Protocol:** *Email*
- **Endpoint:** Enter your email address
- Choose **Create subscription**



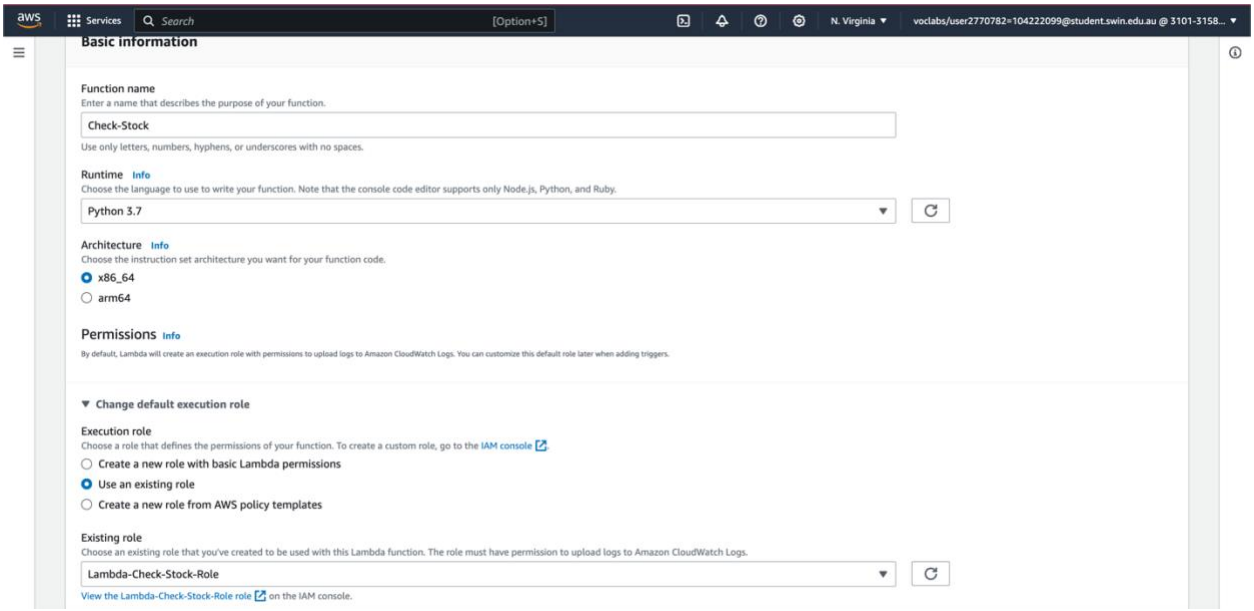




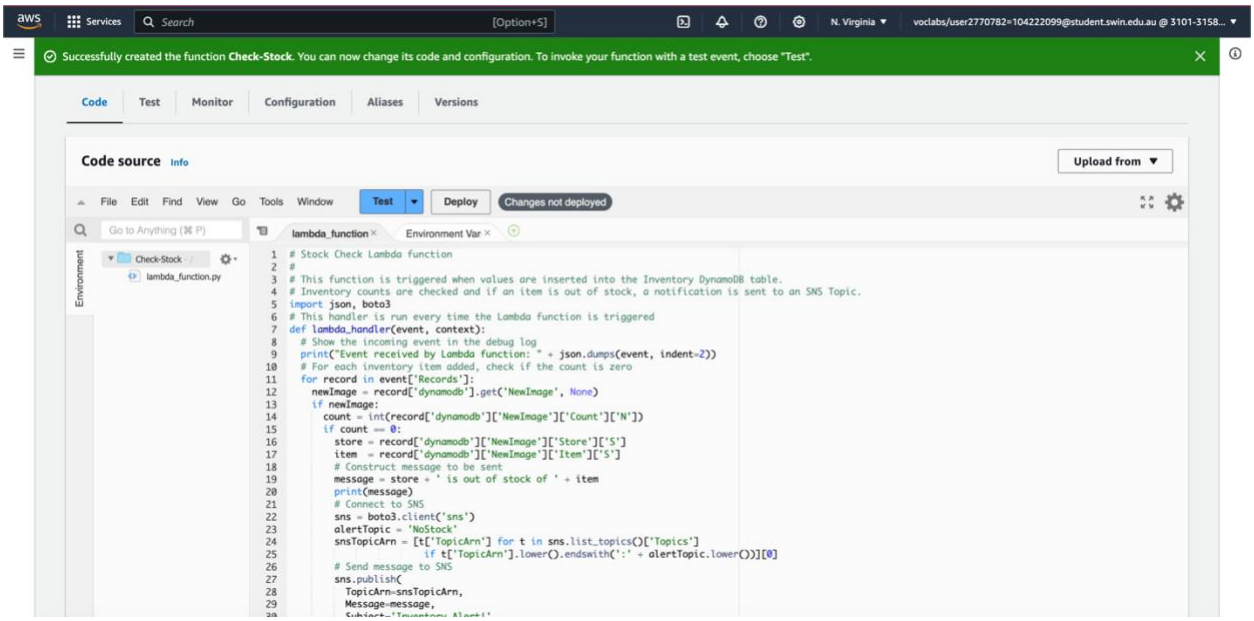
Task 5: Creating a Lambda function to send notifications

On the **Services** menu, choose **Lambda**.
Choose **Create function** and configure these settings:

- **Function name:** Check-Stock
- **Runtime:** *Python 3.7*
- Expand **Choose or create an execution role**.
- **Execution role:** *Use an existing role*
- **Existing role:** *Lambda-Check-Stock-Role*
- Choose **Create function**

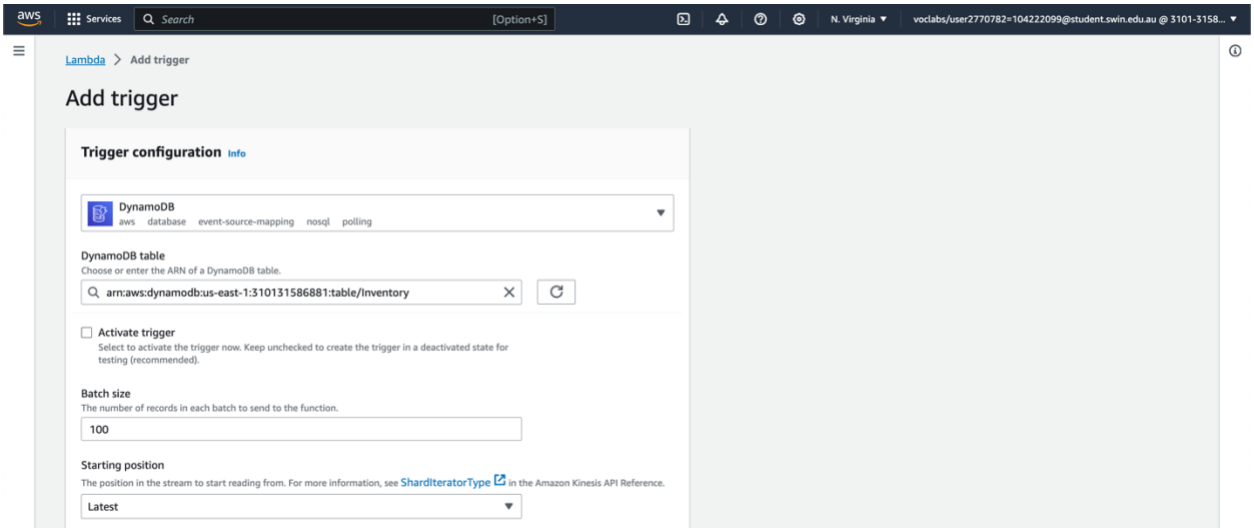


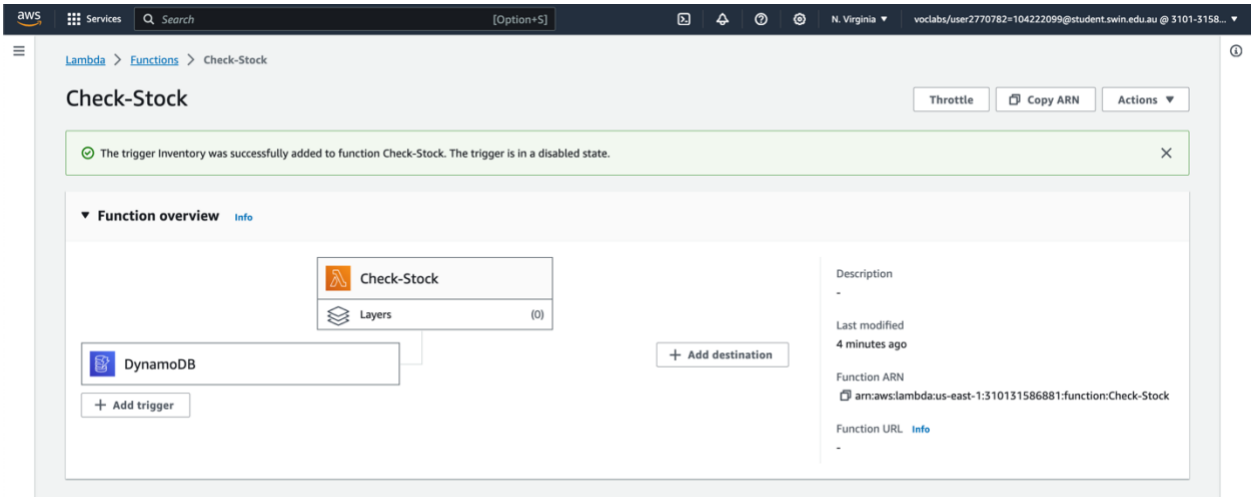
Scroll down to the **Code source** section, and in the **Environment** pane, choose lambda_function.py.
In the code editor, delete all the code.
Copy the following code, and in the **Code Source** editor, paste the copied code:



Choose **Deploy** to save your code changes
Scroll to the **Designer** section (which is at the top of the page).
Choose **Add trigger** and then configure these settings:

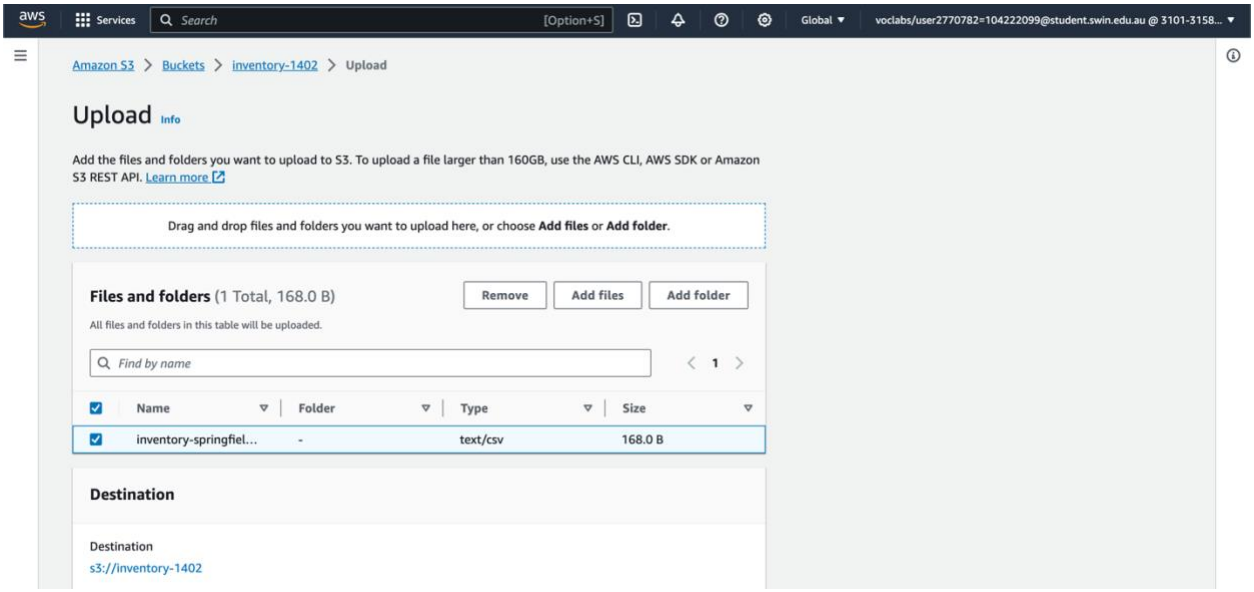
- **Select a trigger:** *DynamoDB*
- **DynamoDB Table:** *Inventory*
- Choose **Add**



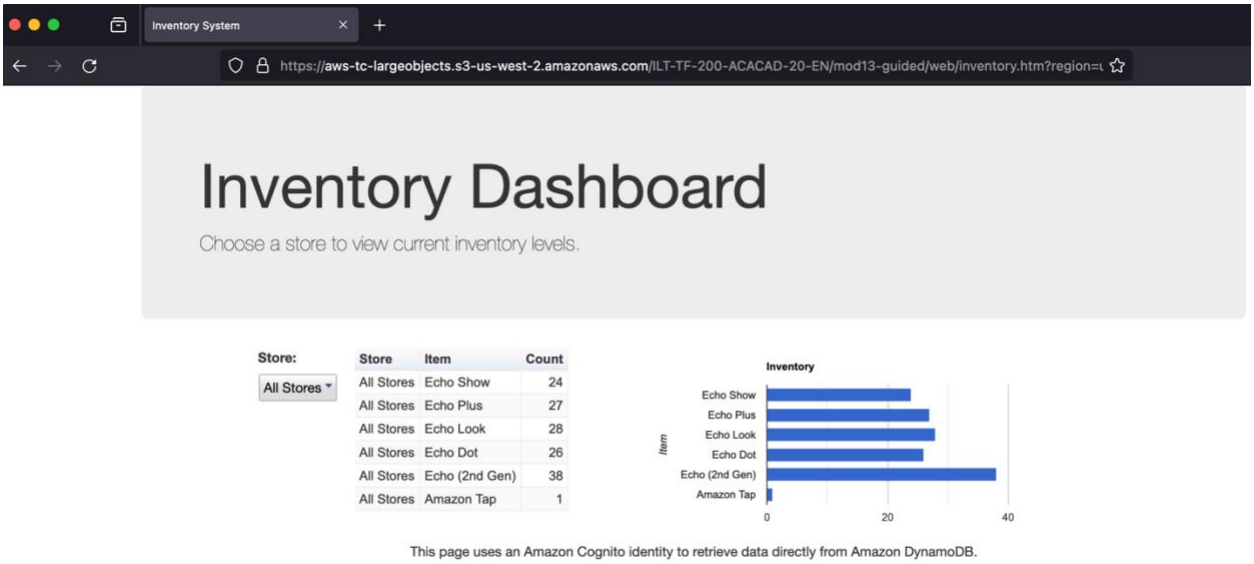


Task 6: Testing the System

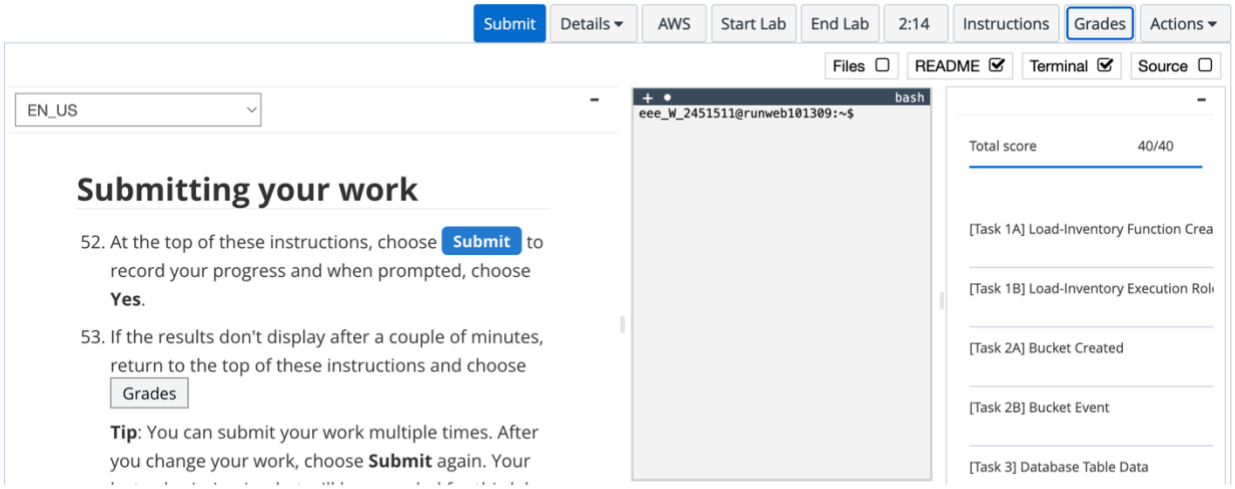
On the **Services** menu, choose **S3**.
Choose the name of your *inventory*- bucket.
Choose **Upload** and upload a different inventory file



Return to the **Inventory System Dashboard** and refresh the page.



Submitting work


Submit
Details
AWS
Start Lab
End Lab
2:14
Instructions
Grades
Actions
Files
README
Terminal
Source
EN_US
bash
eee_W_2451511@runweb101309:~$
Total score
40/40
[Task 1A] Load-Inventory Function Crea
[Task 1B] Load-Inventory Execution Rol
[Task 2A] Bucket Created
[Task 2B] Bucket Event
[Task 3] Database Table Data
Submitting your work
52. At the top of these instructions, choose Submit to record your progress and when prompted, choose Yes.
53. If the results don't display after a couple of minutes, return to the top of these instructions and choose Grades
Tip: You can submit your work multiple times. After you change your work, choose Submit again. Your

ENDLAB.