

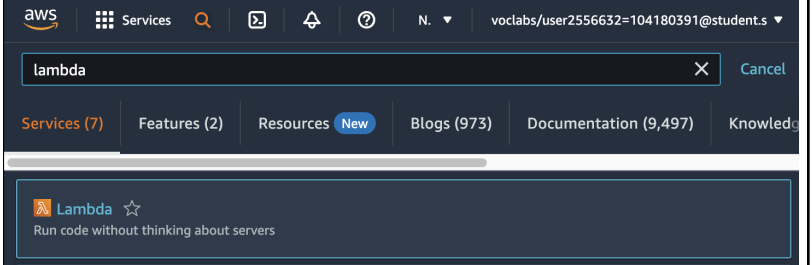
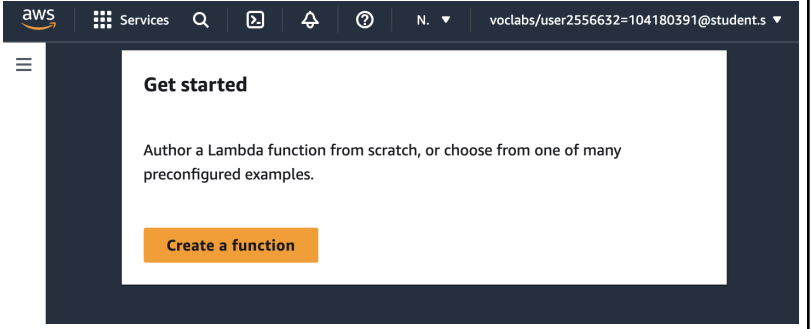
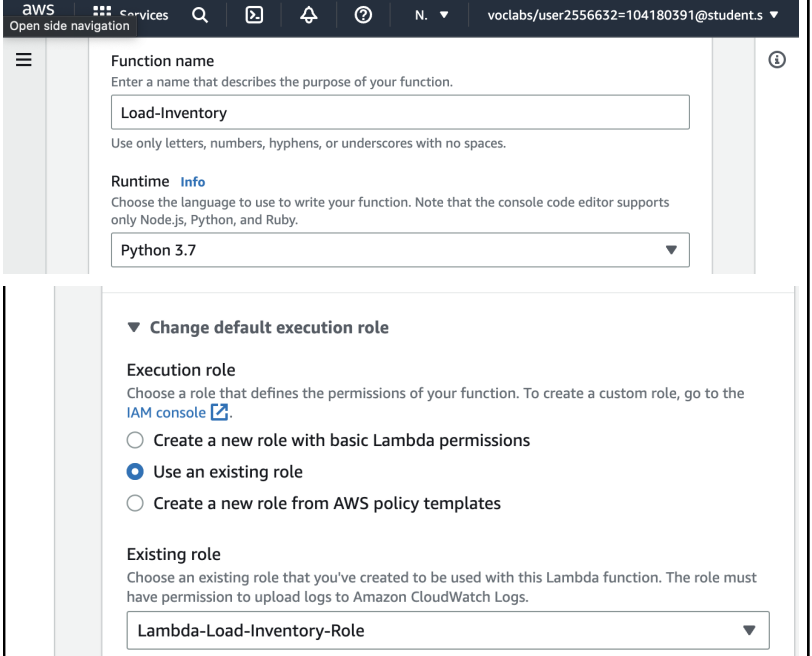


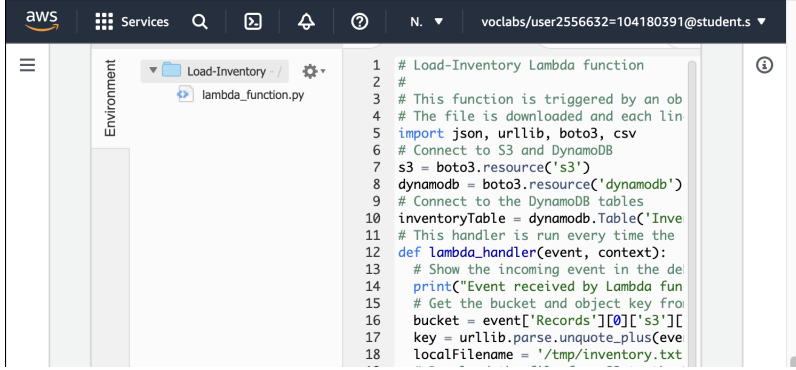
Module 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

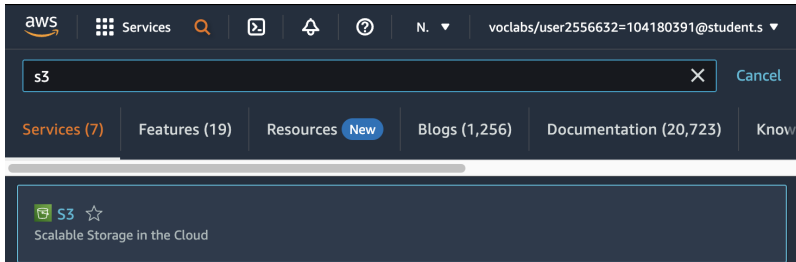
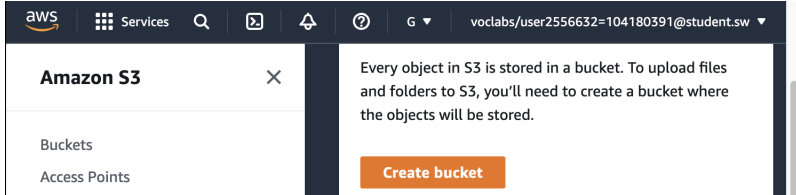
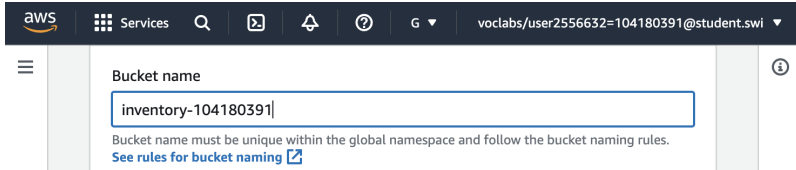
July 6, 2023

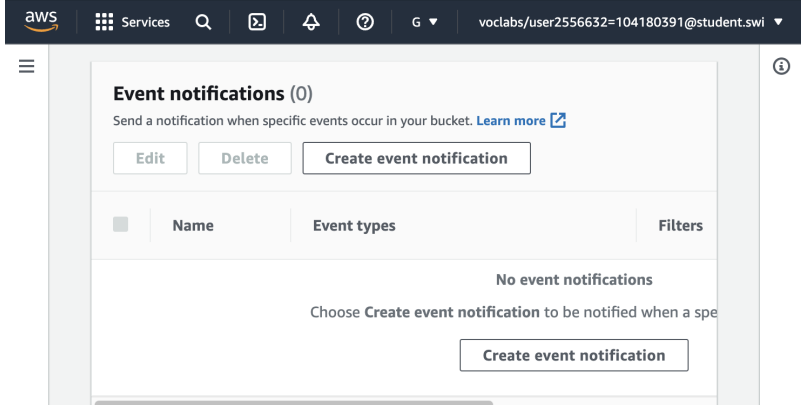
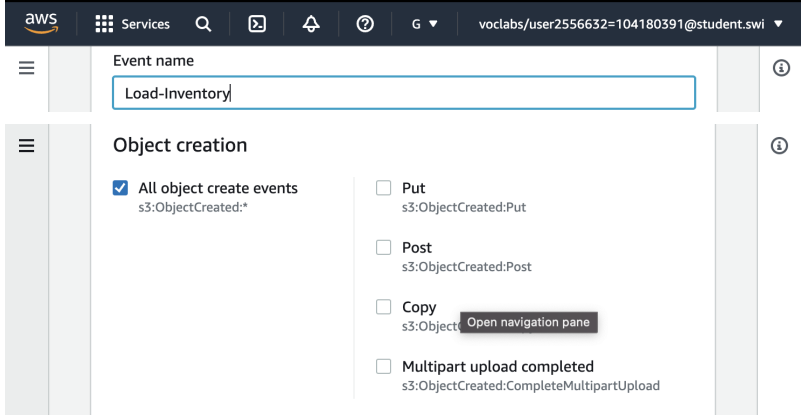
Luu Tuan Hoang
Student ID: 104180391

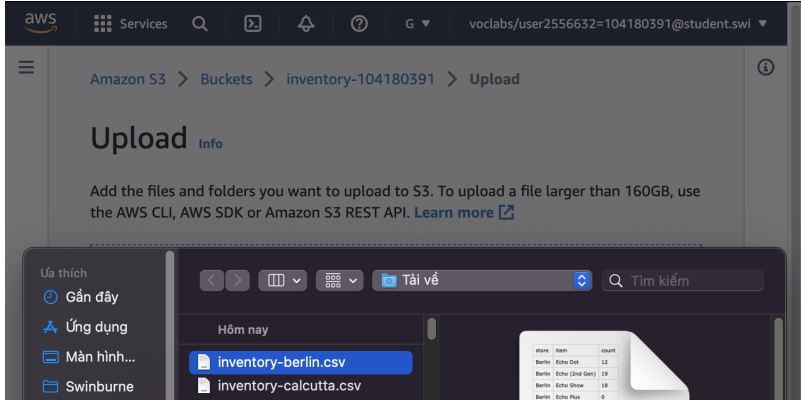
Task 1: Creating a Lambda function to load data

Step	Description	Screenshot
1	In the AWS Management Console, on the Services menu, choose Lambda .	 A screenshot of the AWS Management Console showing the search results for 'lambda'. The search bar at the top contains the text 'lambda'. Below the search bar, there are tabs for 'Services (7)', 'Features (2)', 'Resources', 'Blogs (973)', 'Documentation (9,497)', and 'Knowledge'. The 'Services (7)' tab is selected, and it shows a card for 'Lambda' with a star icon and the text 'Run code without thinking about servers'.
2	Choose Create function	 A screenshot of the 'Get started' page for creating a Lambda function. The page has a dark blue header with the AWS logo and navigation icons. The main content area is white and contains the text 'Author a Lambda function from scratch, or choose from one of many preconfigured examples.' Below this text is an orange button labeled 'Create a function'.
3	<p>Use the Author from scratch option, configure the following settings:</p> <ul style="list-style-type: none">- Function name: Load-Inventory- Runtime: Python 3.7 <p>Expand Choose or create an execution role.</p> <ul style="list-style-type: none">- Execution role: Use an existing role- Existing role: Lambda-Load-Inventory-Role <p>Choose Create function</p>	 A screenshot of the 'Create new function' page in the AWS Management Console. The page is divided into two main sections. The top section is for configuring the function details, and the bottom section is for selecting an execution role. In the top section, the 'Function name' is set to 'Load-Inventory' and the 'Runtime' is set to 'Python 3.7'. In the bottom section, under 'Change default execution role', the 'Execution role' is set to 'Use an existing role', and the 'Existing role' is set to 'Lambda-Load-Inventory-Role'.

4	<p>In the Code source editor, copy and paste the code.</p> <p>Choose Deploy to save your changes.</p>	 <pre> 1 # Load-Inventory Lambda function 2 # 3 # This function is triggered by an ob 4 # The file is downloaded and each lin 5 import json, urllib, boto3, csv 6 # Connect to S3 and DynamoDB 7 s3 = boto3.resource('s3') 8 dynamodb = boto3.resource('dynamodb') 9 # Connect to the DynamoDB tables 10 inventoryTable = dynamodb.Table('Inve 11 # This handler is run every time the 12 def lambda_handler(event, context): 13 # Show the incoming event in the de 14 print("Event received by Lambda fun 15 # Get the bucket and object key fro 16 bucket = event['Records'][0]['s3'] 17 key = urllib.parse.unquote_plus(eve 18 localFilename = '/tmp/inventory.txt </pre>
---	---	---

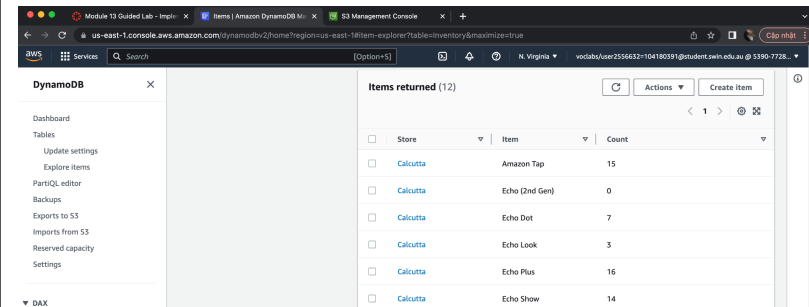
Task 2: Configuring an Amazon S3 event		
Step	Description	Screenshot
1	In the AWS Management Console, on the Services menu, choose S3 .	
2	Choose Create bucket	
3	<p>For Bucket name enter: inventory-<number> (Replace with a random number)</p> <p>Choose Create bucket</p>	

4	<p>Choose the name of your inventory-bucket.</p> <p>Choose the Properties tab.</p> <p>Scroll down to Event notifications.</p>	
5	<p>Click Create event notification then configure these settings:</p> <ul style="list-style-type: none"> - 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		
Step	Description	Screenshot
1	<p>Download the inventory files</p> <p>In the console, return to your S3 bucket by choosing the Objects tab.</p> <p>Choose Upload</p> <p>Choose Add files, and select one of the inventory CSV files. (You can choose any inventory file.)</p> <p>Choose Upload</p>	

2

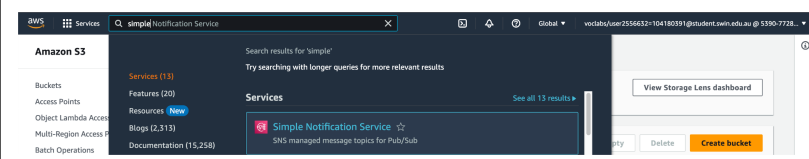
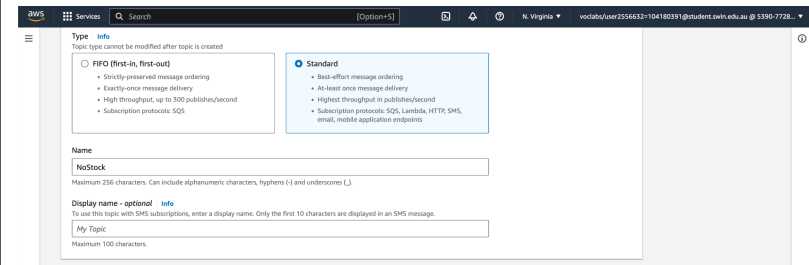
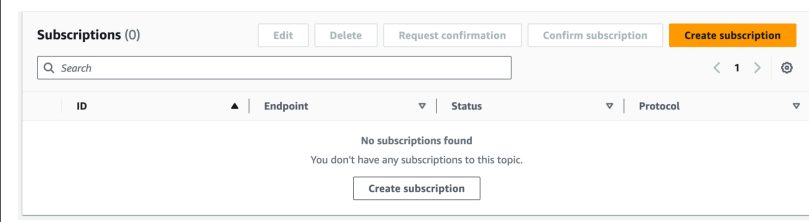
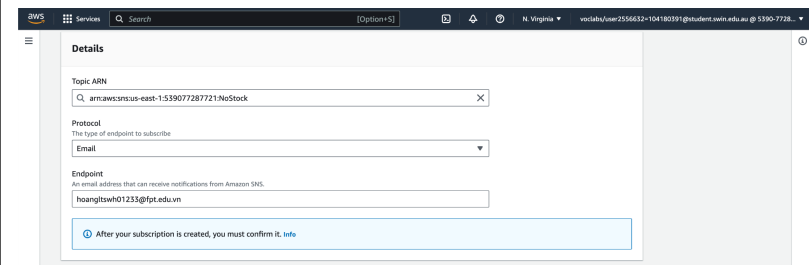
Since I cannot open the dashboard, this is the data from the inventory file will be displayed. It shows the store, item and inventory count in DynamoDB.



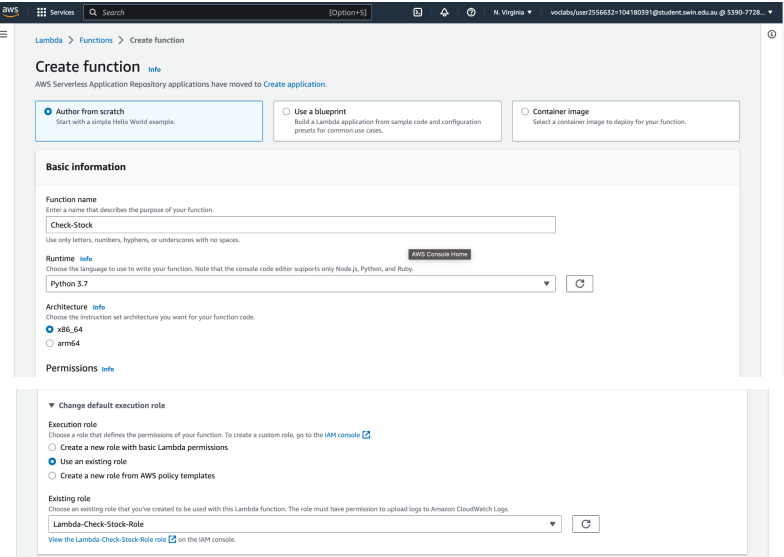
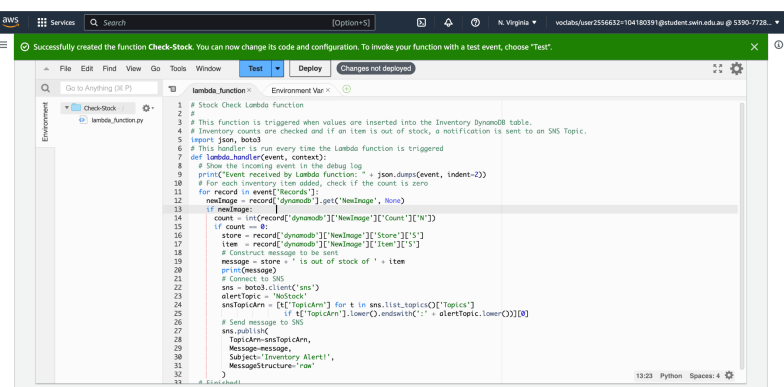
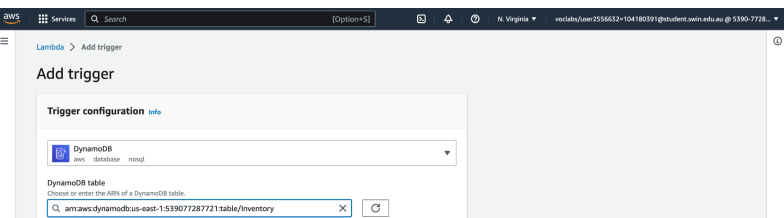
The screenshot shows the AWS Management Console interface for a DynamoDB table named 'inventory'. The left sidebar contains navigation links for Dashboard, Tables, Update settings, Explore items, PartiQL editor, Backups, Exports to S3, Imports from S3, Reserved capacity, and Settings. The main content area displays 'Items returned (12)' in a table format. The table has three columns: 'Store', 'Item', and 'Count'. The data rows are as follows:

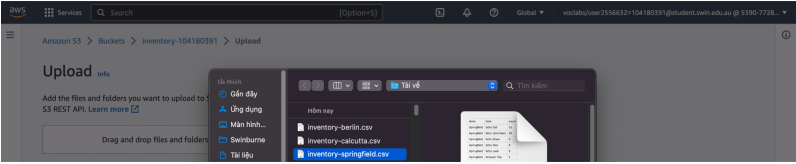
Store	Item	Count
Calcutta	Amazon Tap	15
Calcutta	Echo (2nd Gen)	0
Calcutta	Echo Dot	7
Calcutta	Echo Look	3
Calcutta	Echo Plus	16
Calcutta	Echo Show	14

Task 4: Configuring notifications

Step	Description	Screenshot
1	On the Services menu, choose Simple Notification Service .	
2	In the Create topic box, for Topic name , enter: NoStock . Keep Standard selected. Choose Create topic	
3	In the lower half of the page, choose Create subscription	
4	Configure these settings: <ul style="list-style-type: none">- Protocol: Email- Endpoint: Enter your email address- Choose Create subscription	

Task 5: Creating a Lambda function to send notifications

Step	Description	Screenshot
1	<p>On the Services menu, choose Lambda.</p> <p>Choose Create function and configure these settings:</p> <ul style="list-style-type: none">- Function name: Check-Stock- Runtime: Python 3.7 <p>Expand Choose or create an execution role.</p> <ul style="list-style-type: none">- Execution role: Use an existing role- Existing role: Lambda-Check-Stock-Role <p>Choose Create function</p>	
2	<p>Copy the code to the Code Source editor</p> <p>Choose Deploy to save your code changes</p>	
3	<p>Scroll to the Designer section (which is at the top of the page).</p> <p>Choose Add trigger and then configure these settings:</p> <ul style="list-style-type: none">- Select a trigger: DynamoDB- DynamoDB Table: Inventory- Choose Add	

Task 5: Creating a Lambda function to send notifications		
Step	Description	Screenshot
1	<p>On the Services menu, choose S3.</p> <p>Choose the name of your inventory-bucket.</p> <p>Choose Upload and upload a different inventory file</p>	
2	<p>Lab completed with all tasks finished.</p>	