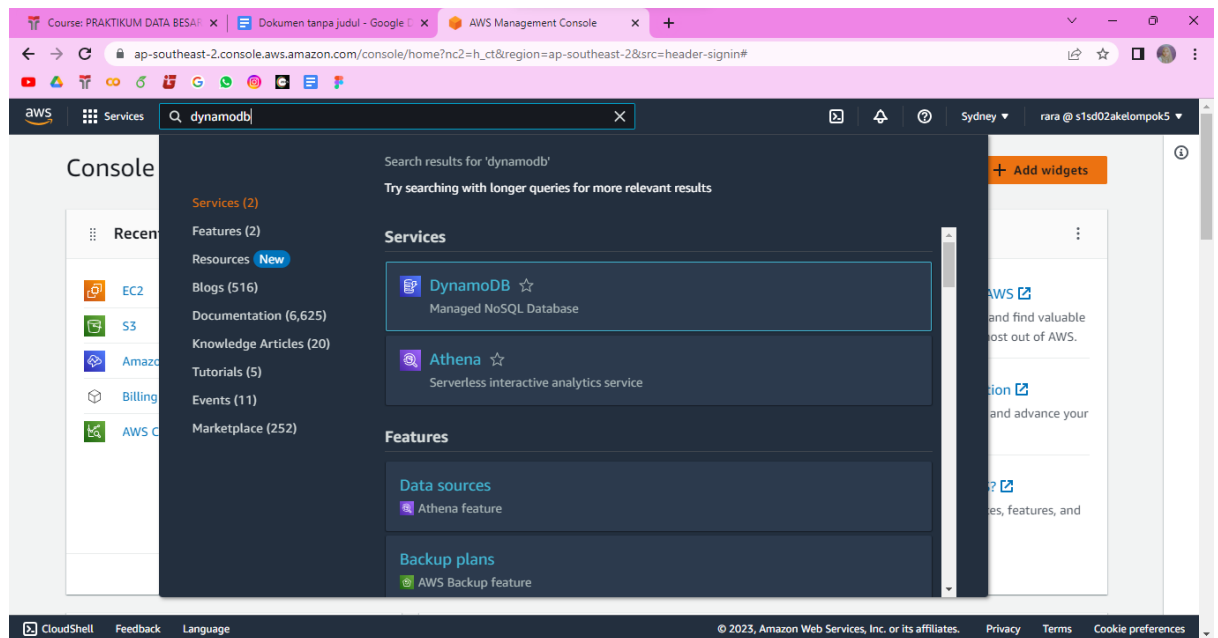


Creating, Importing, Querying, and Exporting Data with Amazon DynamoDB

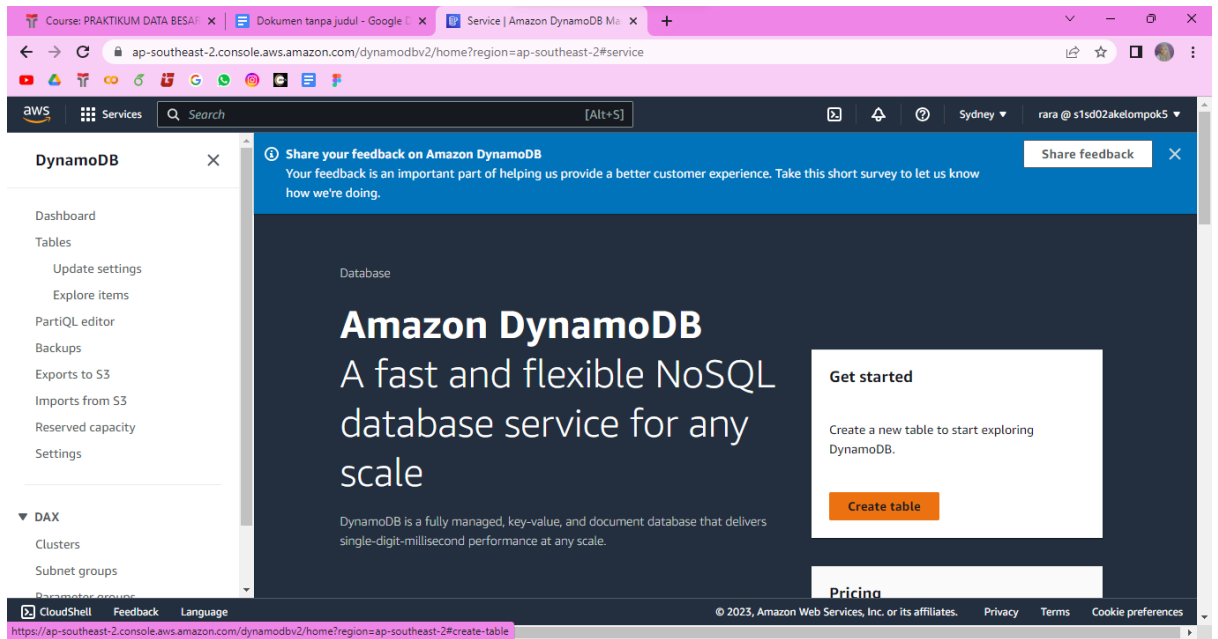
In this tutorial, we will explore how to use the Amazon DynamoDB service to create a table, import data from a CSV file, perform queries or scans with filters, and export the results. DynamoDB is a fully managed NoSQL database service provided by Amazon Web Services (AWS), known for its scalability, performance, and low latency. By the end of this tutorial, you will have a clear understanding of how to work with DynamoDB for basic data operations.

Step 1: Creating a DynamoDB Table:

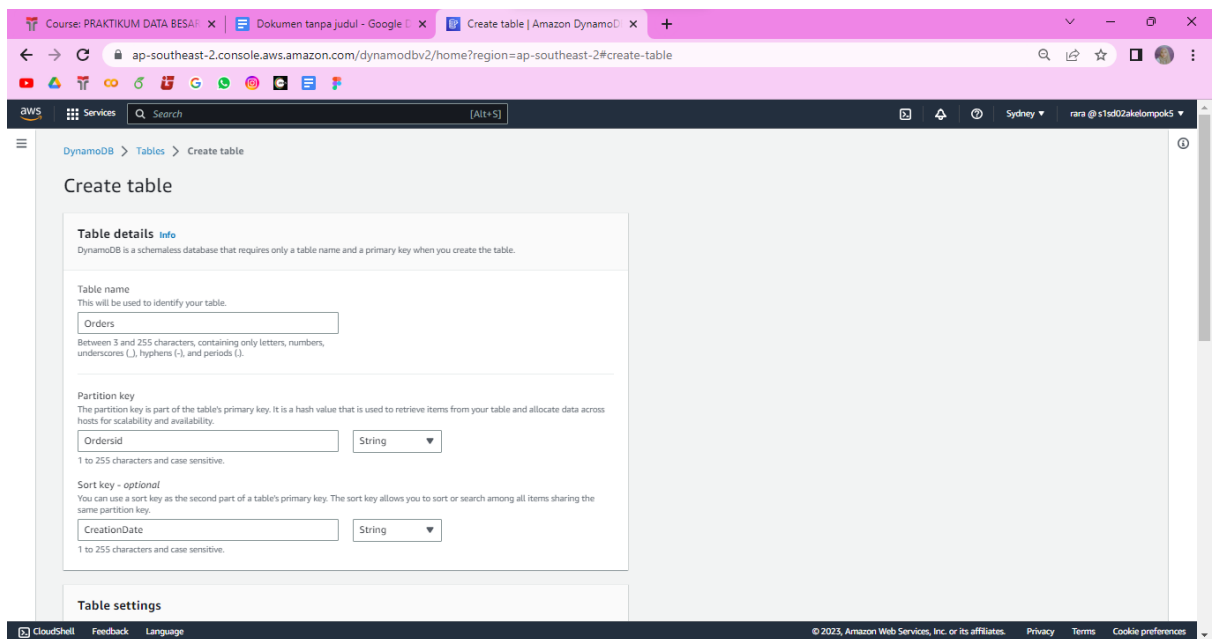
1. Launch the AWS Management Console and navigate to the DynamoDB service.



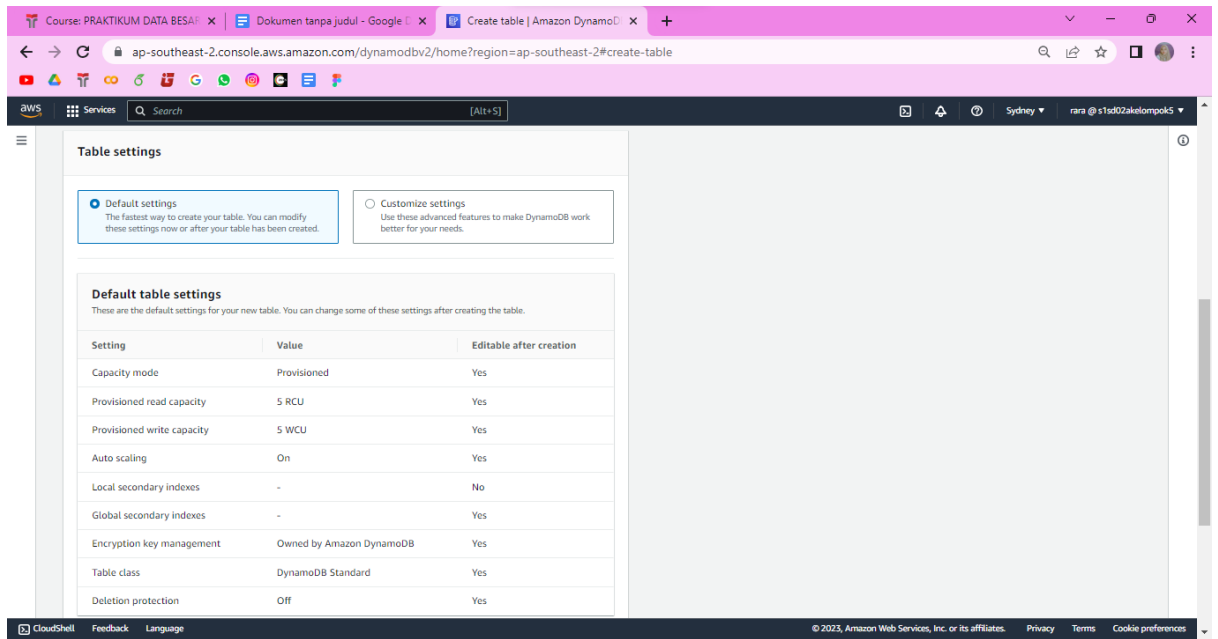
2. Click on "Create table" and provide a suitable table name.



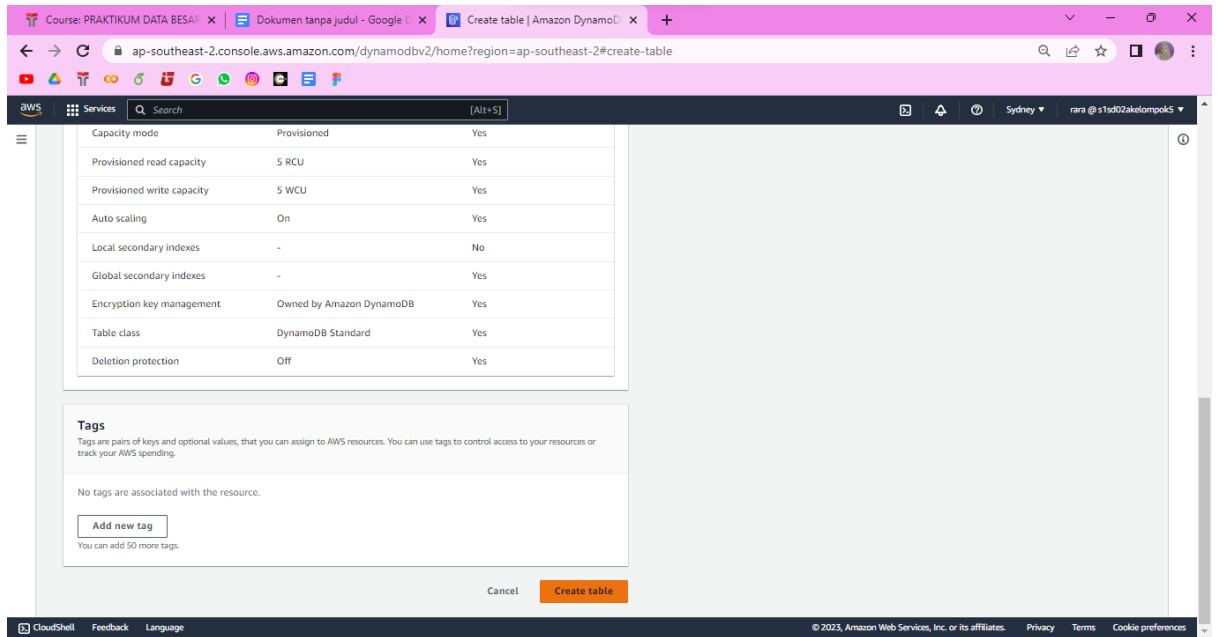
3. Specify the primary key attributes (partition key and optionally a sort key) for your table.



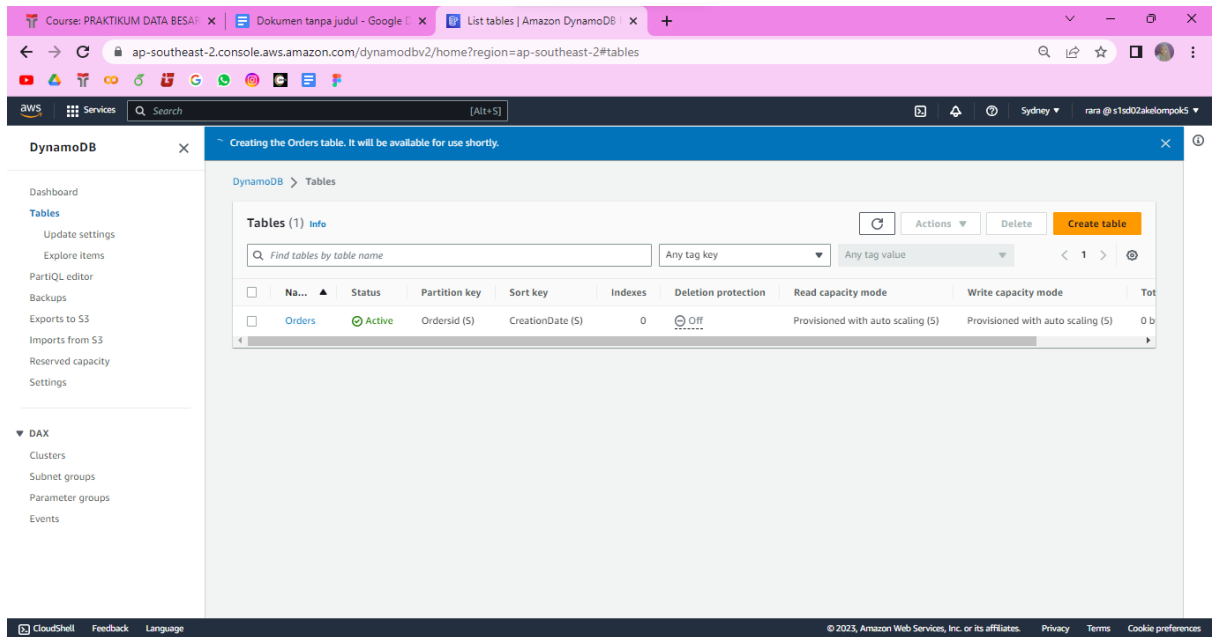
4. In the Table Settings section, select "Default Settings".



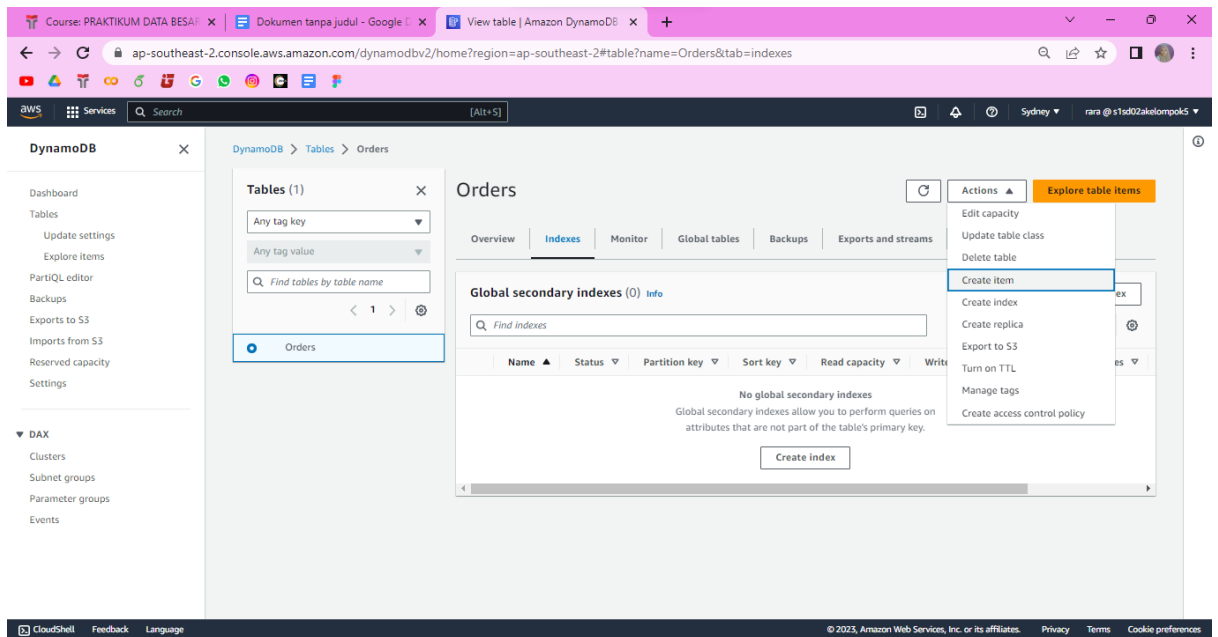
5. Click on "Create" to create the table.



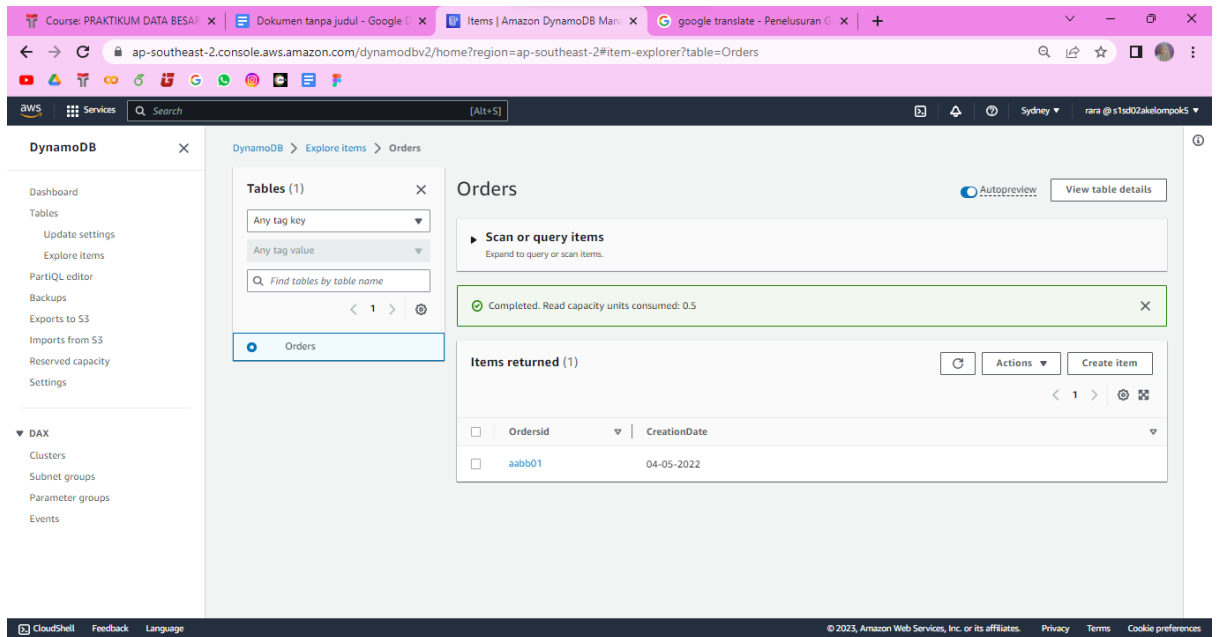
6. Wait a few moments until the table named 'Orders' is successfully created and the status appears as 'Active'.



- After that, click on 'Orders'. Open the 'Actions' menu and select 'Create Item'.

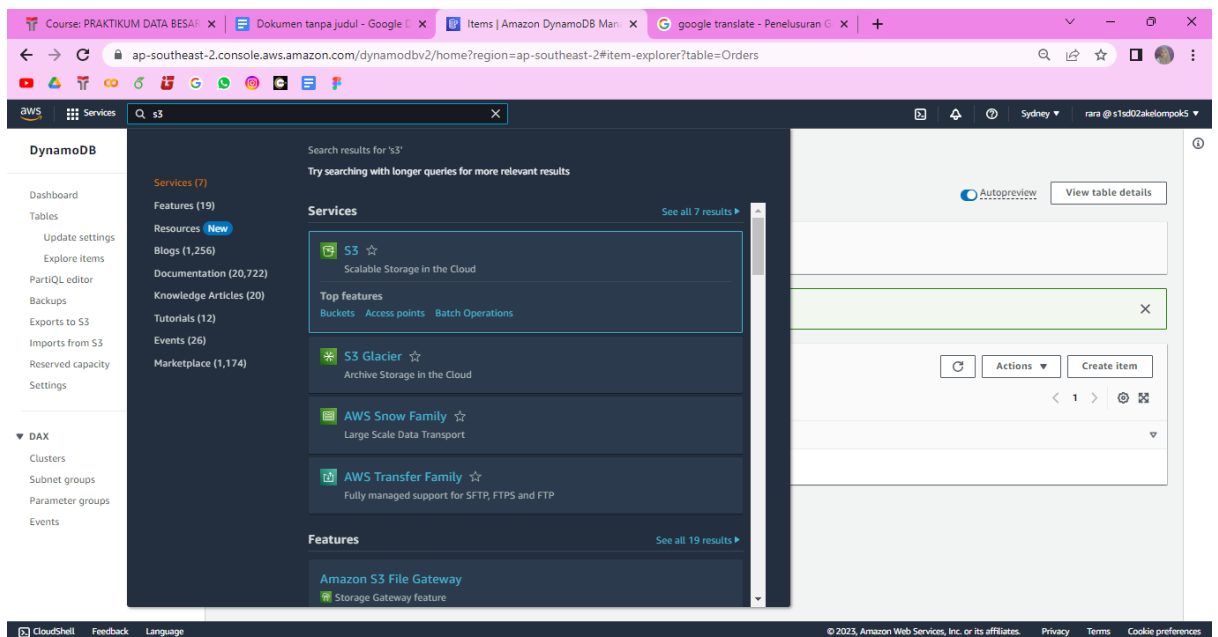


- Fill in the values for the attributes that we created earlier in the 'Orders' table, then click on 'Create Item'. Here is the display of attribute values in the 'Tabel_Mahasiswa' table that have been inputted.

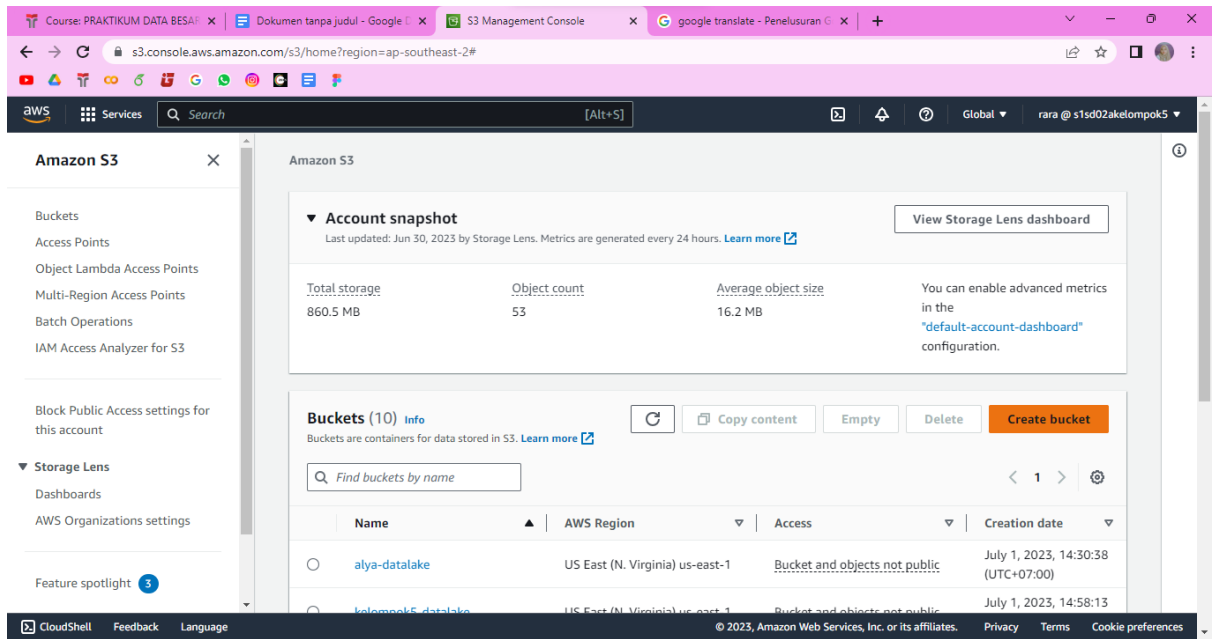


Step 2: Importing Data from a CSV file:

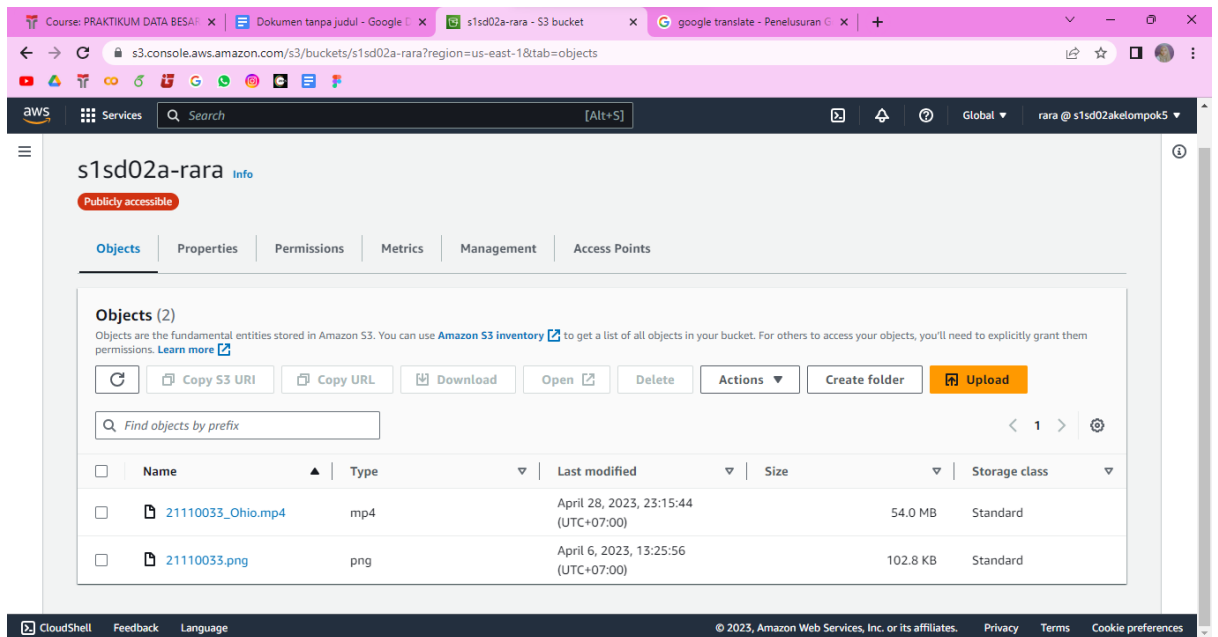
1. Search for the "S3" menu.



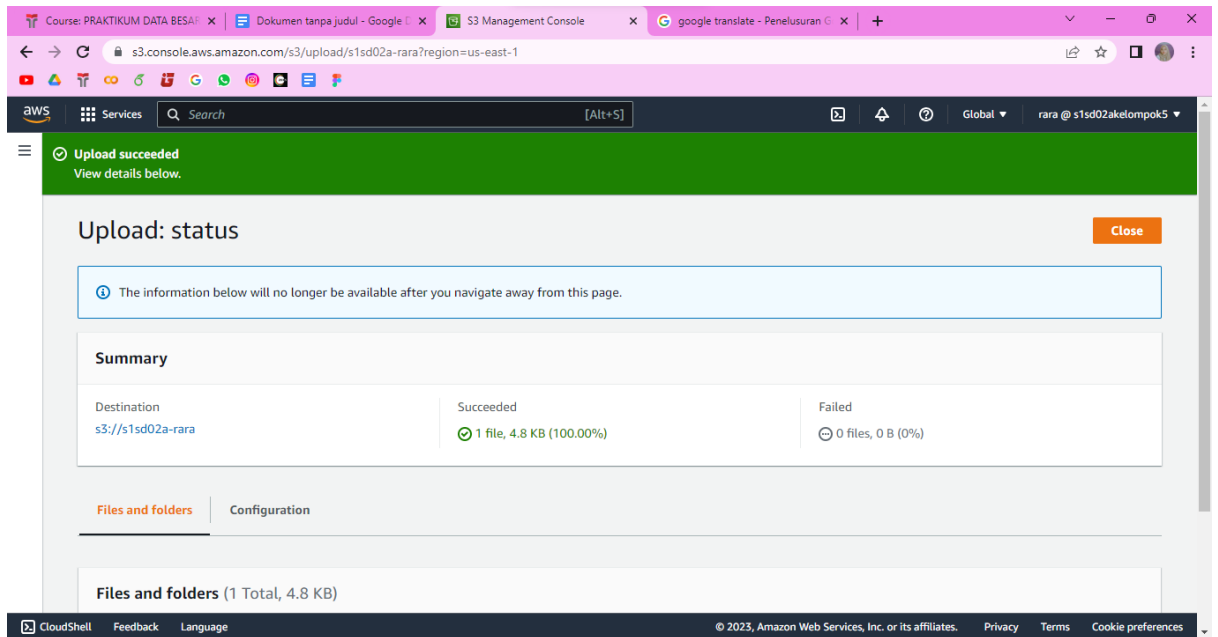
2. Open the "Amazon S3" menu and navigate to "Buckets". Here, I have already created a bucket. If you haven't created one yet, you can do so by clicking on "Create Bucket".



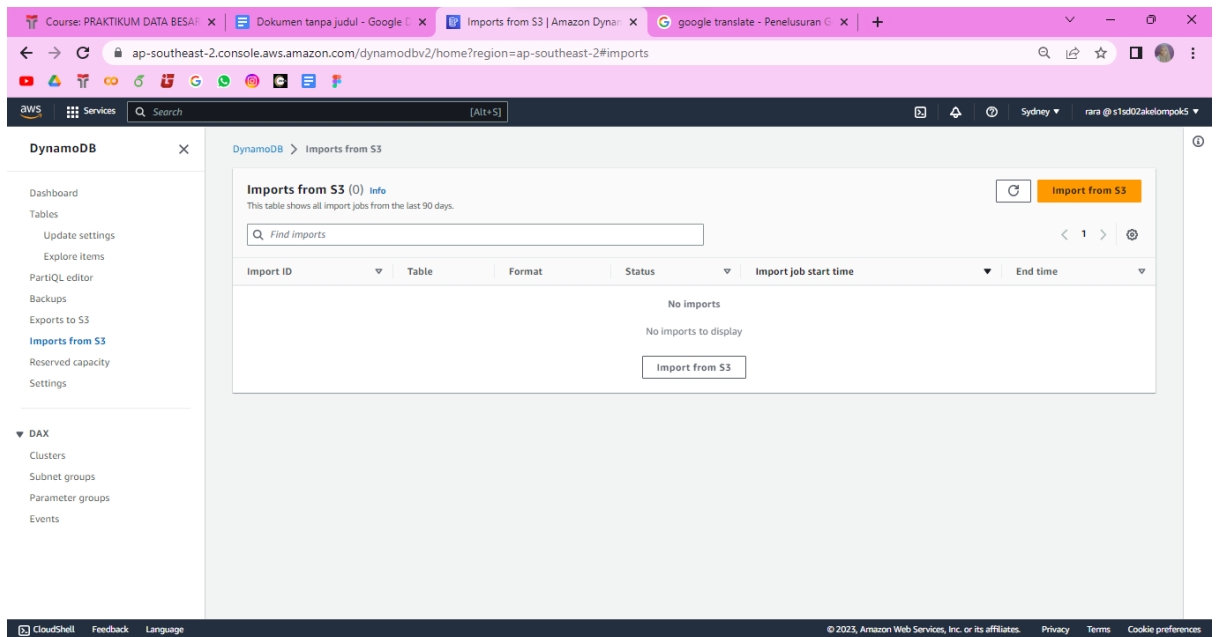
3. Open the created bucket.



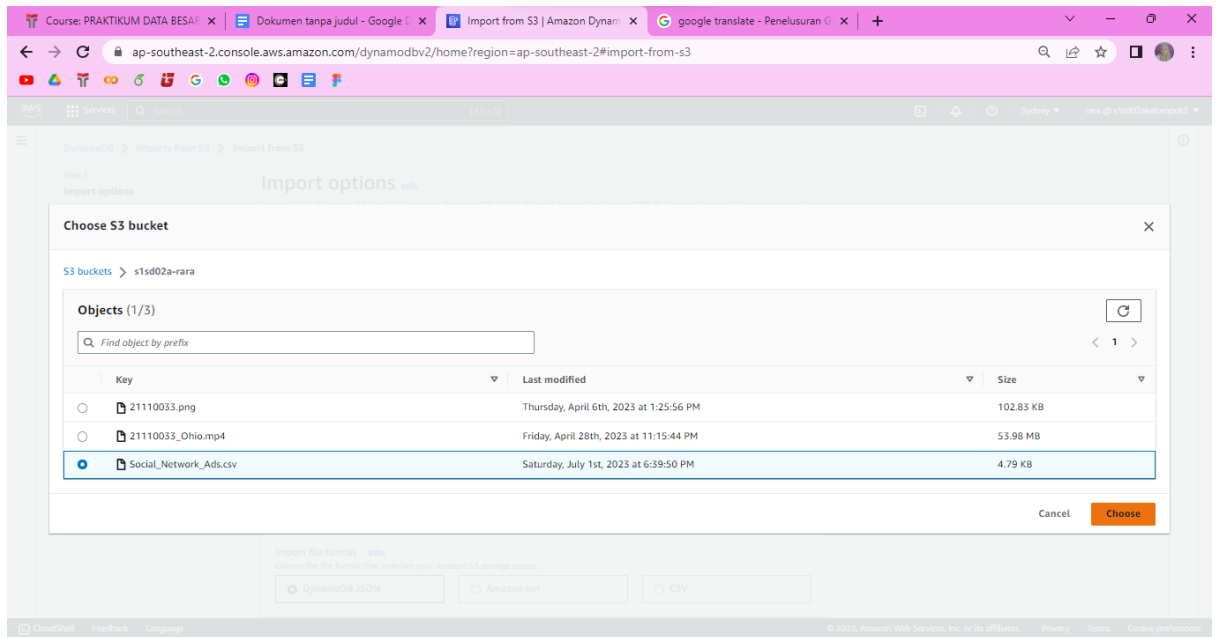
4. Upload the CSV file by clicking on "Upload" > "Add Files" > Select the CSV file > "Upload". Wait for a few moments until the file is successfully uploaded.



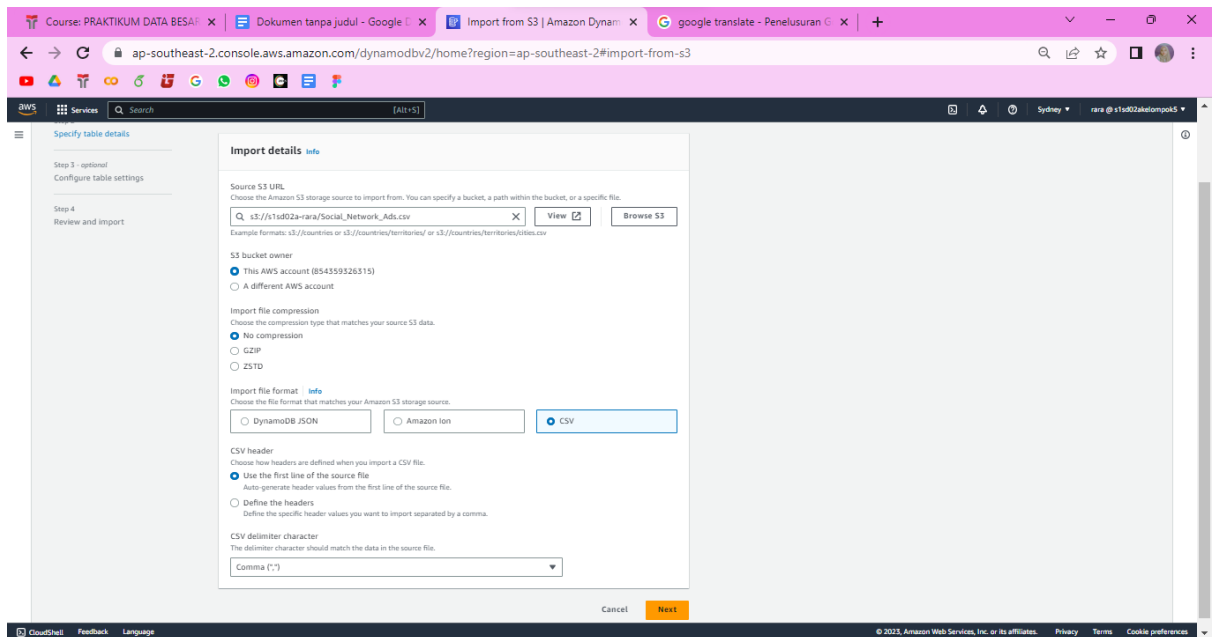
5. Search for the "DynamoDB" menu and navigate to "Import from S3".



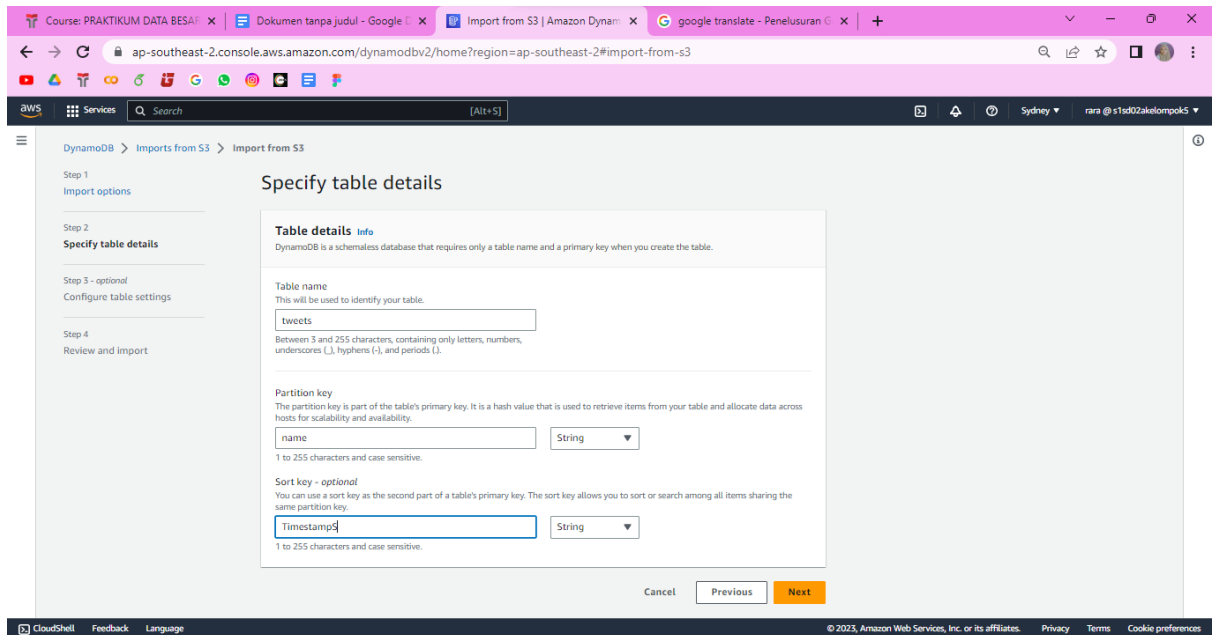
6. Click on "Import from S3" > search for the uploaded CSV file > Browse S3 > select the bucket containing the targeted CSV file > choose the desired CSV file > select.



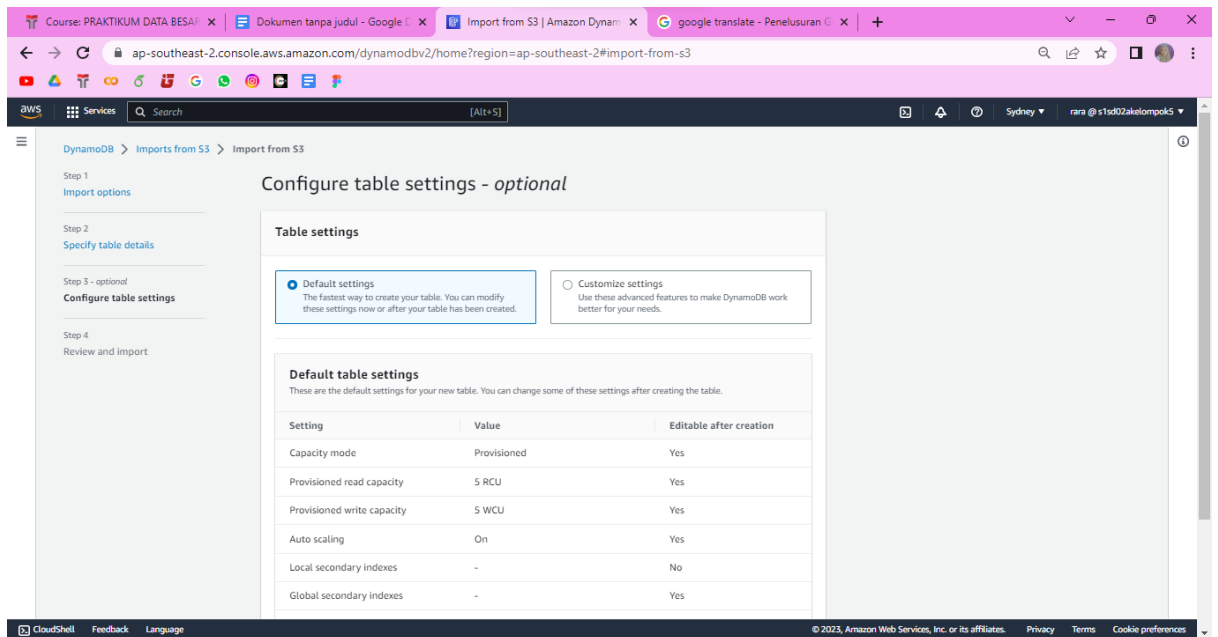
7. Configure the Import Details settings as follows, then click Next.



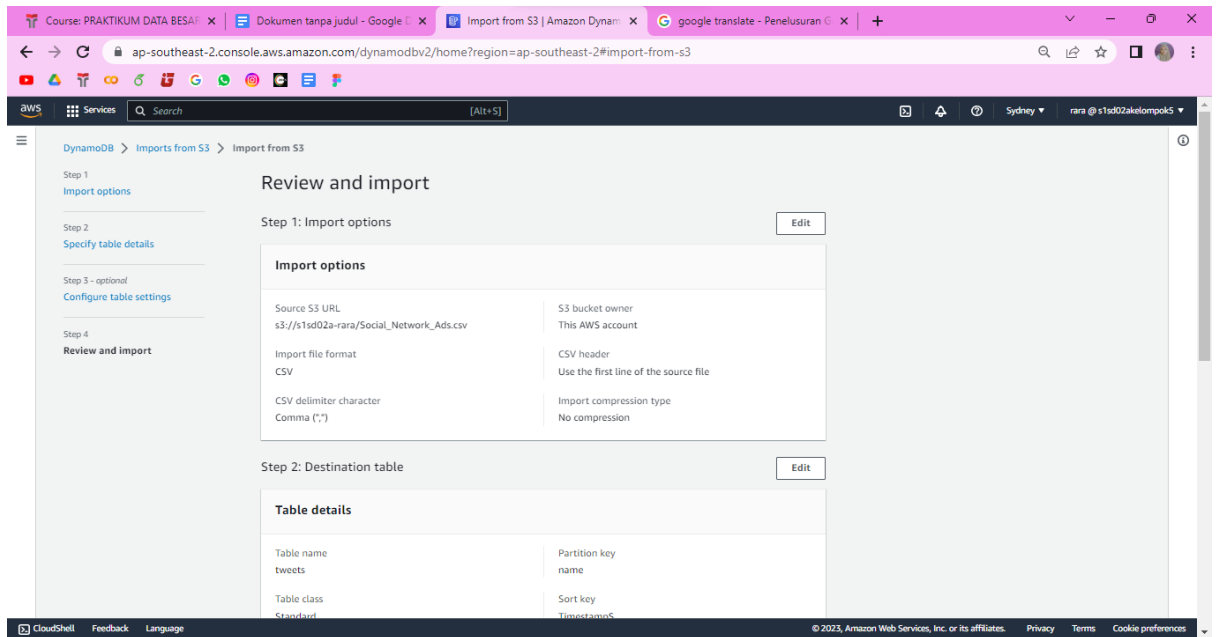
8. In the Table Details section, enter the Table Name, Partition Key, and Optional Sort Key.



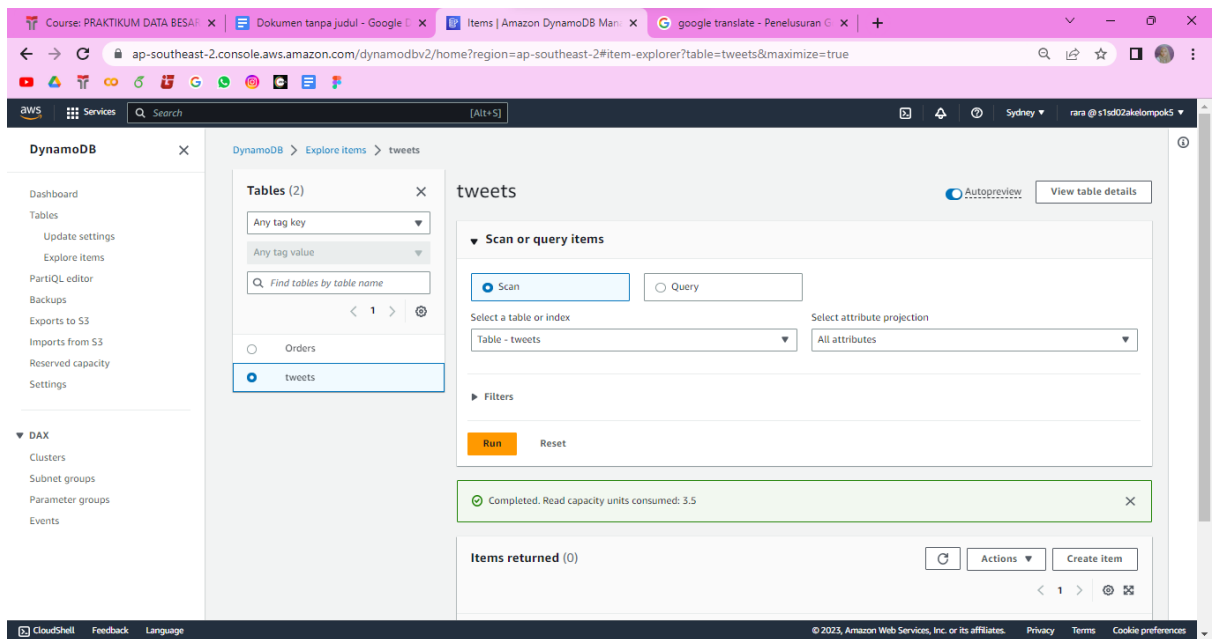
9. In the Optional Table Configuration section, select "Default settings" and click Next.



10. Review the Import and Review section carefully, then click on Import to proceed.



11. After the CSV file has been successfully imported, go back to the DynamoDB menu > Tables > select the uploaded CSV file > Explore Table Items. Here is the content of the imported CSV file.



Step 3: Querying and Scanning Data with Filters:

1. SCAN

Orders
Autopreview
View table details

▼ Scan or query items

☒ Scan
☐ Query

Select a table or index

Table - Orders

Select attribute projection

Specific attributes

Specific attributes to project

Enter attribute name

Add attribute

CreationDate X

▶ Filters

Run

Reset

Completed. Read capacity units consumed: 0.5

Items returned (1)

↺

Actions ▼

Create item

< 1 >

<input type="checkbox"/>	CreationDate
<input type="checkbox"/>	04-05-2022

2. Query

Orders
Autopreview
View table details

▼ Scan or query items

☐ Scan
☒ Query

Select a table or index

Table - Orders

Select attribute projection

Specific attributes

Specific attributes to project

Enter attribute name

Add attribute

CreationDate X

Ordersid (Partition key)

Ordersid

CreationDate (Sort key)

Equal to

Enter sort key value

☐ Sort descending

▶ Filters

Run

Reset

Completed. Read capacity units consumed: 0.5

3. Filter

Orders Autopreview View table details

▼ Scan or query items

☒ Scan ☐ Query

Select a table or index: Table - Orders

Select attribute projection: Specific attributes

Specific attributes to project: Add attribute

×

▼ Filters

Attribute name	Type	Condition	Value	
<input type="text" value="aabb"/>	<input type="text" value="String"/>	<input type="text" value="Equal to"/>	<input type="text" value="Enter attribute value"/>	<input type="button" value="Remove"/>

Reset

Completed. Read capacity units consumed: 0.5 ×

Step 4: Exporting Results:

1. Search for the "DynamoDB" menu and navigate to "Export to S3". Select "Orders" as the table, fill in the Destination S3 Bucket with the bucket you created earlier, and click "Export". Wait for the export process to complete until the Status shows "Completed".

Course: PRAKTIKUM DATA BESA | Dokumen tanpa judul - Google | Exports to S3 | Amazon Dynamoi | google translate - Penelusuran | +

ap-southeast-2.console.aws.amazon.com/dynamodbv2/home?region=ap-southeast-2#exports

RWS Services Search [Alt+S] Sydney rra @ s1d02zalaiompok5

DynamoDB x Exporting the Orders table to the s3://s1d02z-ara S3 bucket

DynamoDB > Export to S3

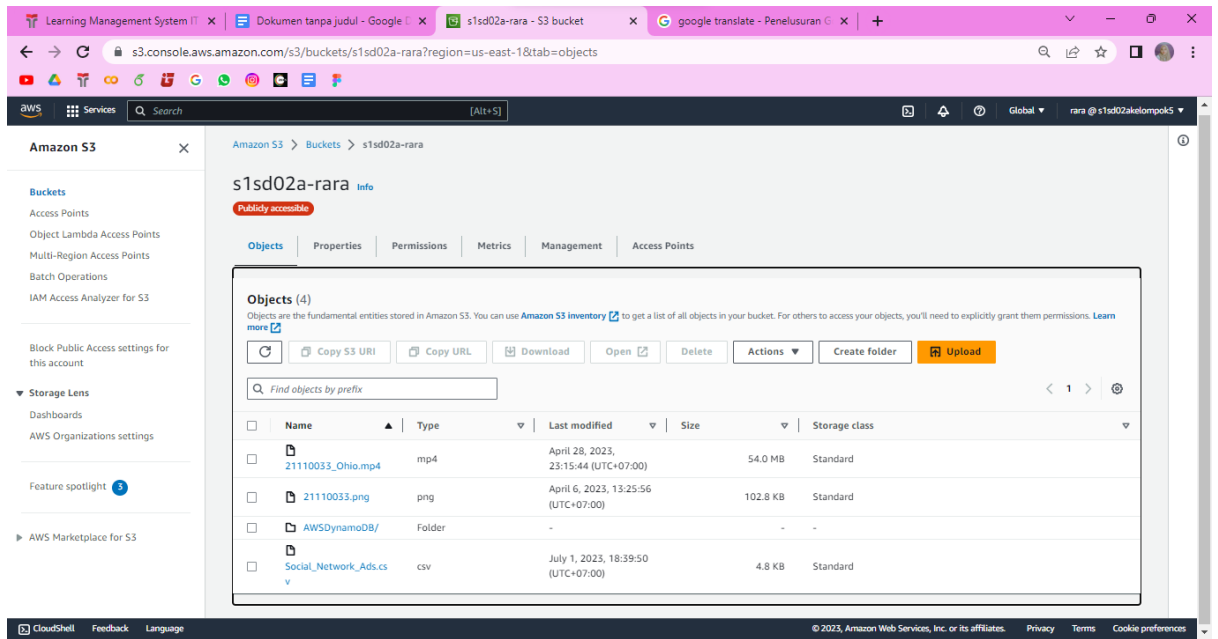
Exports to S3 (1) info View details Export to S3

Showing all export jobs from the last 90 days.

Export ARN	Table name	Destination S3 bucket	Status	Export job start time (UTC+07:00)
arn:aws:dynamodb:ap-southeast-2:854359526315:table/Orders/export/0168821...	Orders	s3://s1d02z-ara	Exporting	Jul 1, 2023, 19:22:05

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

2. Verify if the table that was created has been successfully exported by navigating to Amazon S3 > Bucket > select the created bucket. You should see a folder named "AWS DynamoDB/". This folder represents the exported data from the created table.



Conclusion:

In this tutorial, we have covered the essential steps to create a DynamoDB table, import data from a CSV file, perform queries and scans with filters, and export the results. Amazon DynamoDB offers a highly scalable and fully managed NoSQL database solution, empowering developers to build robust and high-performance applications. With the skills acquired from this tutorial, you can leverage the power of DynamoDB for various use cases in your projects.

Remember to clean up any resources (such as tables) you have created to avoid unnecessary costs in your AWS account.