



# Load Data into a DynamoDB Table



Nchindo Boris

The screenshot shows the AWS Management Console interface for the Amazon DynamoDB service. The left sidebar navigation includes 'Dashboard', 'Tables', 'Explore items' (which is selected), 'PartiQL editor', 'Backups', 'Exports to S3', 'Imports from S3', 'Integrations', 'Reserved capacity', and 'Settings'. Below this is a 'DAX' section with 'Clusters', 'Subnet groups', 'Parameter groups', and 'Events'. The main content area displays a table titled 'Table: ContentCatalog - Items returned (6)'. The table has columns: Id (Number), Authors, ContentType, Difficulty, Price, and ProjectC. The data rows are:

Id (Number)	Authors	ContentType	Difficulty	Price	ProjectC
3	[{"\$": "Ne...}	Project	Easy peasy	0	AI/ML
2	[{"\$": "Ne...}	Project	Easy peasy	0	Analytics
203		Video		0	
202		Video		0	
201		Video		0	
1	[{"\$": "Nat...}	Project	Easy peasy	0	Storage



# Introducing Today's Project!

## What is Amazon DynamoDB?

Amazon DynamoDB is a fully managed NoSQL database service offered by AWS. It is ideal for applications that require high availability, low latency, and seamless scaling.

## How I used Amazon DynamoDB in this project

In this project, I used Amazon DynamoDB to Load data into a NoSQL database

## One thing I didn't expect in this project was...

One thing I didn't expect in this project is that data would be loaded using Shell

## This project took me...

This project took me 40 minutes to complete



# Create a DynamoDB table

DynamoDB tables organises data using lists of items (i.e. StudentNames, like Nikko), each with their own list of attributes

An attribute is a piece of data about an item.

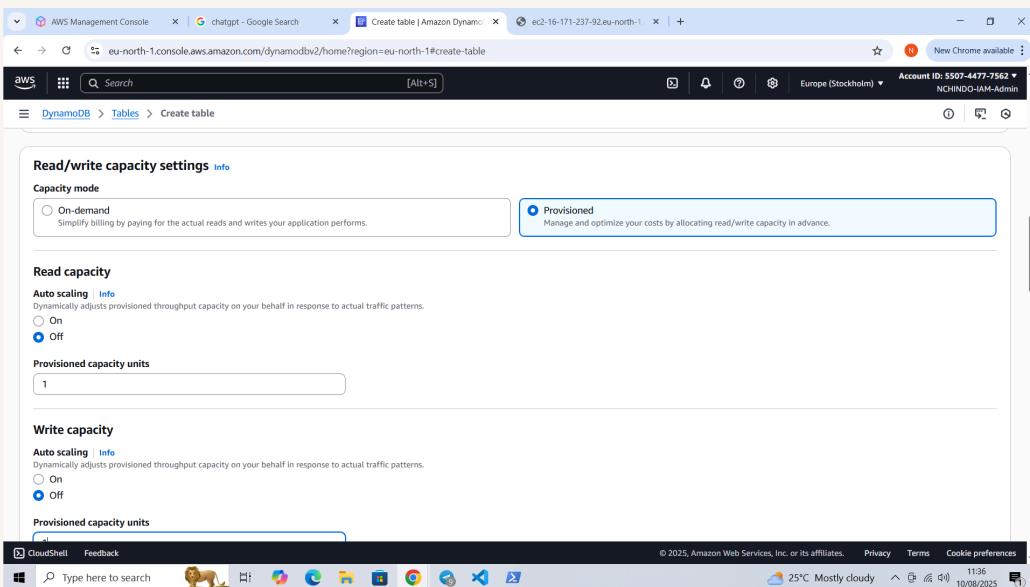
The screenshot shows the AWS Management Console interface for the DynamoDB service. The left sidebar has a navigation tree with 'DynamoDB' selected, under which 'Tables' is expanded, showing 'NextWorkStudents' as the active table. The main content area is titled 'Explore items' for 'NextWorkStudents'. It features a search bar at the top and a 'Scan' button. Below the search bar, there are dropdown menus for 'Select a table or index' (set to 'Table - NextWorkStudents') and 'Select attribute projection' (set to 'All attributes'). A section for 'Filters - optional' includes a 'Run' button. A green status bar at the bottom indicates 'Completed - Items returned: 0 - Items scanned: 0 - Efficiency: 100% - RCU consumed: 0.5'. Below this, a table titled 'Table: NextWorkStudents - Items returned (1)' shows one item: 'Nikko'. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.



# Read and Write Capacity

Read capacity units (RCUs) and write capacity units (WCUs) are measurements of how many engines DynamoDB is using to operate.

Amazon DynamoDB's Free Tier covers 25GB of data storage, plus 25 Write and 25 Read Capacity Units. I turned off auto scaling because boost your table's processing power which can push my table's settings to go over Free Tier limits.

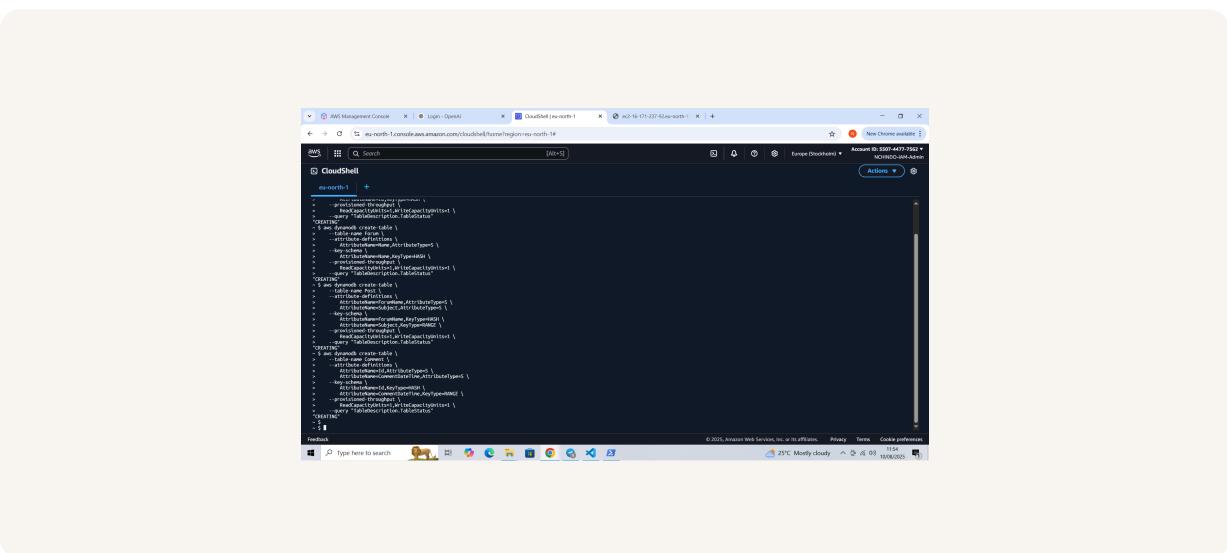


# Using CLI and CloudShell

AWS CloudShell is shell in your AWS Management Console, which means it's a space for you to run code

AWS CLI is a software that lets you create, delete and update AWS resources with commands instead of clicking through your console

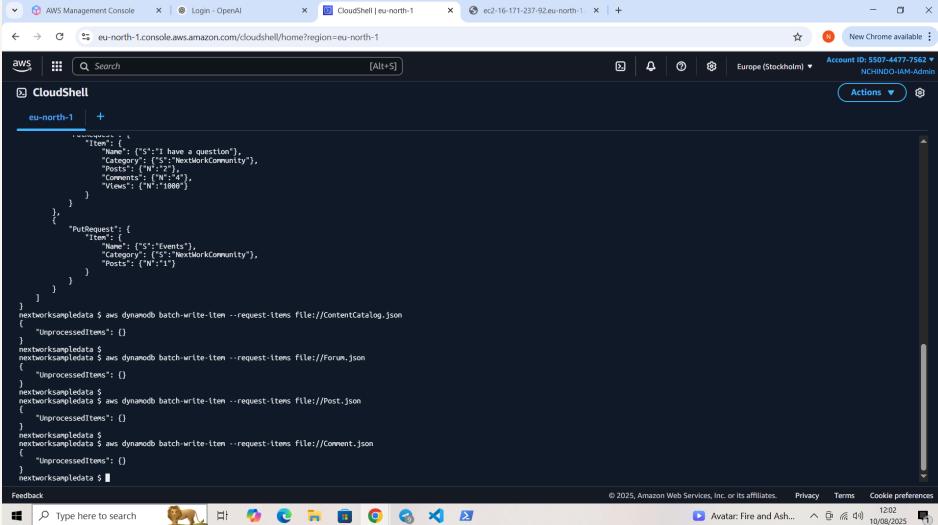
I ran a CLI command in AWS CloudShell that created four new tables in AWS DynamoDB, each with specific attributes and settings.



# Loading Data with CLI

I ran a CLI command in AWS CloudShell to;

- Download and unzip the zip file with data
- Load the data of all four files into DynamoDB using AWS CLI's batch-write-item command



The screenshot shows a terminal window in the AWS CloudShell interface. The user has run a command to download and extract a zip file, then iterated through four JSON files (ContentCatalog.json, Forum.json, Post.json, Comment.json) using the AWS CLI's batch-write-item command to load their contents into a DynamoDB table.

```
aws dynamodb batch-write-item --request-items file://ContentCatalog.json
aws dynamodb batch-write-item --request-items file://Forum.json
aws dynamodb batch-write-item --request-items file://Post.json
aws dynamodb batch-write-item --request-items file://Comment.json
```

# Observing Item Attributes

Attributes		
Attribute name	Value	Type
Id - Partition key	1	Number
Authors	Insert a field	List
ContentType	Project	String
Difficulty	Easy peasy	String
Price	0	Number
ProjectCategory	Storage	String
Published	<input checked="" type="radio"/> True <input type="radio"/> False	Boolean
Title	Host a Website on Amazon S3	String
URL	aws-host-a-website-on-s3	String

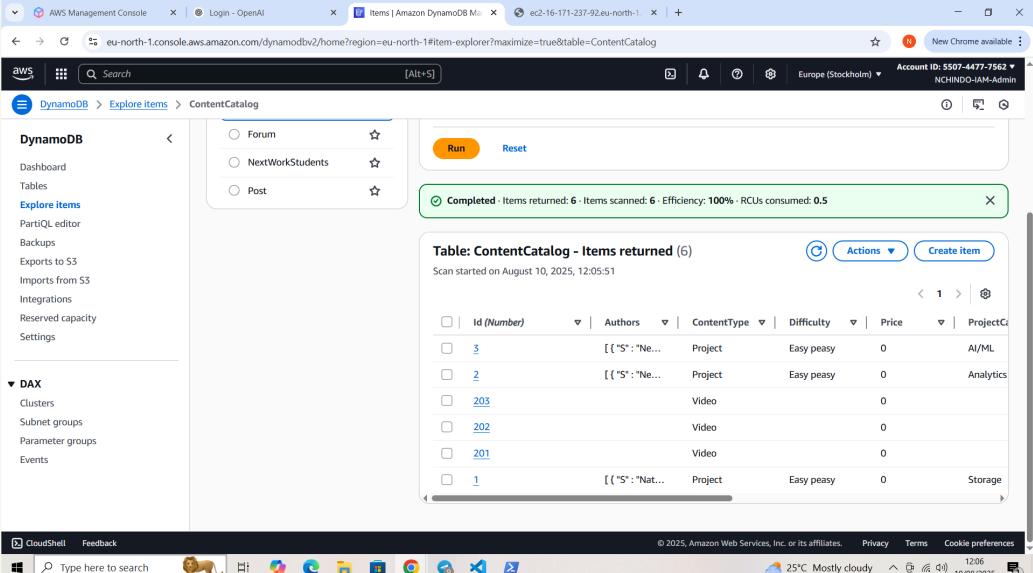
I checked a ContentCatalog item, which had the following attributes: - ContentType - Difficulty - Price - Project - Category - Published - Title - URL

I checked another ContentCatalog item, which had a different set of attributes; - Content - Type - Price - Services - Title - URL -Video - Type

# Benefits of DynamoDB

A benefit of DynamoDB over relational databases is flexibility, because every item having their own unique set of attributes is a huge advantage when items in a table could look different from each other.

Another benefit over relational databases is speed because DynamoDB tables can use partition keys to split up a table and quickly find the items they're looking for. Relational databases have to scan through the entire table to find data which is slow.



The screenshot shows the AWS Management Console interface for exploring items in a DynamoDB table named 'ContentCatalog'. The left sidebar has a 'DynamoDB' section with 'Explore items' selected, showing tables like Forum, NextWorkStudents, and Post. The main area displays a table titled 'Table: ContentCatalog - Items returned (6)'. The table has columns: Id (Number), Authors, ContentType, Difficulty, Price, and ProjectCategory. The data is as follows:

Id (Number)	Authors	ContentType	Difficulty	Price	ProjectCategory
3	[{"\$": "Ne..."}]	Project	Easy peasy	0	AI/ML
2	[{"\$": "Ne..."}]	Project	Easy peasy	0	Analytics
203		Video		0	
202		Video		0	
201		Video		0	
1	[{"\$": "Nat..."}]	Project	Easy peasy	0	Storage



[nextwork.org](https://nextwork.org)

# The place to learn & showcase your skills

Check out [nextwork.org](https://nextwork.org) for more projects

