



nextwork.org

Load Data into a DynamoDB Table



Sanjana Tripathy

Table: ContentCatalog - Items returned (6)												
Scan started on August 01, 2025, 22:44:53												
	ID (Number)	Authors	ContentType	Difficulty	Price	ProjectCategory	Published	Services	Title	URL	VideoType	
1	1	["S","Nek..."]	Project	Easy peasy	0	AI/ML	true	Build a Ch... aws-ai-tex...	aws-ai-tex...			
2	2	["S","Nek..."]	Project	Easy peasy	0	Analytics	true	Visualize da... aws-analyti...				
	203		Video		0			["S","Am..."]	AWS x Data... https://you...		Live Project Demo	
	202		Video		0			["S","Am..."]	Don't miss... https://you...		Shorts	
	201		Video		0			["S","Am..."]	AWS Relati... https://you...		Educational	
	1	["S","Nat..."]	Project	Easy peasy	0	Storage	true	Host a Web... aws-host-a...				



Introducing Today's Project!

What is Amazon DynamoDB?

Amazon DynamoDB is a fully managed NoSQL database service that provides fast, predictable performance with seamless scalability. It's useful for applications that require low-latency access to large volumes of structured or semi-structured data.

How I used Amazon DynamoDB in this project

I used Amazon DynamoDB to first create tables from scratch and manually add items through the console. Then, I leveraged AWS CLI in CloudShell to automate table creation and batch-load data efficiently—speeding up the process.

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

One thing I didn't expect in this project was how powerful and efficient DynamoDB would be. After working with Aurora, I hadn't realized DynamoDB could so effectively solve challenges related to organizing unstructured data at scale.



Sanjana Tripathy
NextWork Student

nextwork.org

This project took me...

This project took me around 2 hours to complete, as it introduced several new concepts like NoSQL structures, table creation via CLI, and batch data loading, which required a bit of extra exploration and practice.



Sanjana Tripathy
NextWork Student

nextwork.org

Create a DynamoDB table

DynamoDB tables organize data using items and attributes, with each item uniquely identified by a primary key. This key can be a partition key alone or a combination of partition and sort keys, enabling fast and efficient data retrieval.

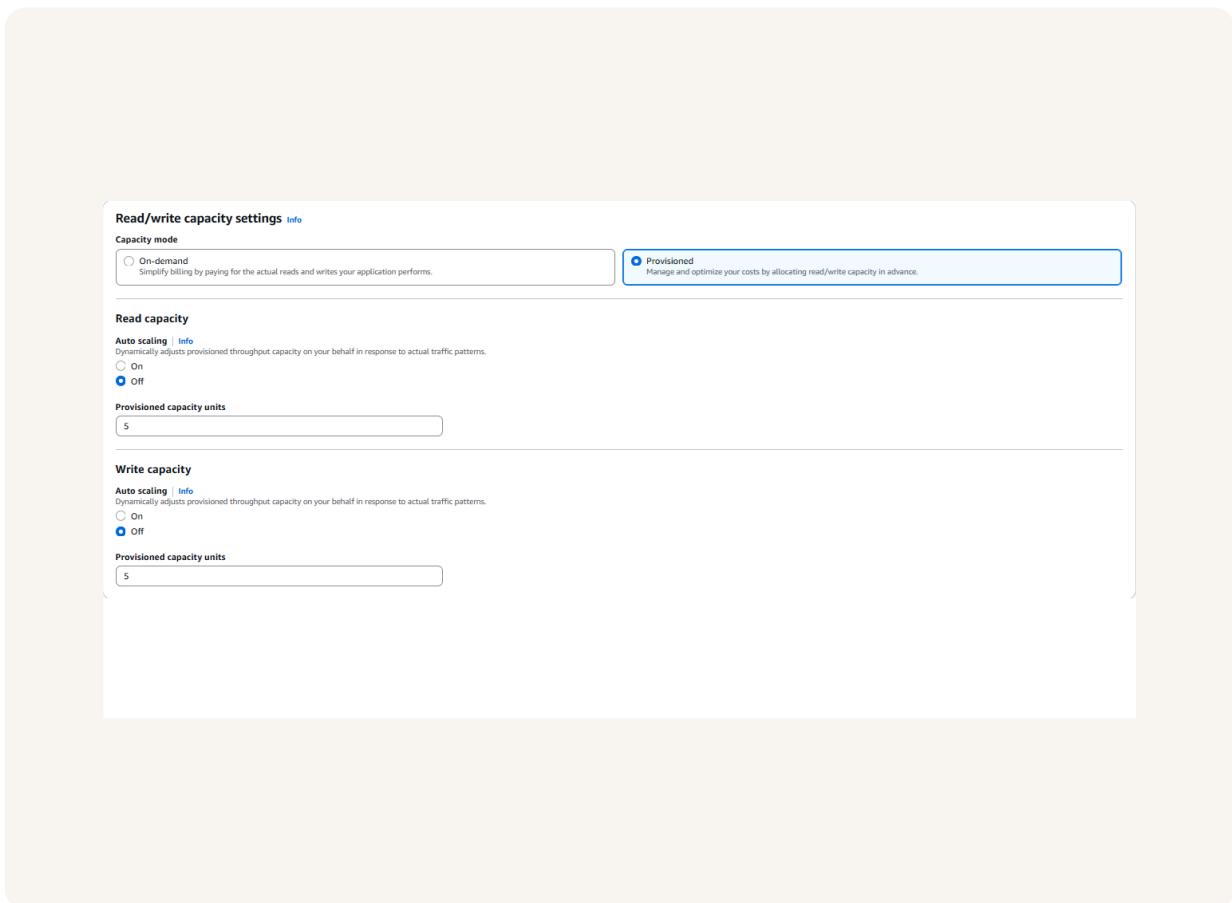
In DynamoDB, an attribute is a data element that defines a specific piece of information within an item, similar to a column in relational databases. Attributes can store various data types like strings, numbers, or binary values.

Table: NextWorkStudents - Items returned (1)	
Scan started on August 01, 2025, 21:36:34	
Actions	Create Item
< 1 >	⋮
StudentName (String)	ProjectName
Sanjana	4

Read and Write Capacity

Read Capacity Units (RCUs) and Write Capacity Units (WCUs) determine how many read and write operations your DynamoDB table can handle per second. They help manage performance and cost by allocating the necessary throughput for your application.

Amazon DynamoDB's Free Tier includes 25GB of data storage, along with 25 Write and 25 Read Capacity Units. I disabled auto-scaling to maintain control over my table's throughput and ensure it aligns with my project's resource usage and cost management



Using CLI and CloudShell

AWS CloudShell is a browser-based shell that provides secure command-line access to AWS resources. It comes pre-installed with essential tools like AWS CLI, eliminating local setup and enabling quick, scriptable access to manage cloud services.

The AWS Command Line Interface is a powerful tool that enables users to interact with AWS services through terminal commands. It supports automation, speeds up resource management, and is ideal for scripting and managing AWS environments efficiently.

I used AWS CLI in CloudShell to efficiently create four DynamoDB tables—ContentCatalog, Forum, Post, and Comment—by running structured commands that defined attributes, key schemas, and provisioned throughput settings.

```
# aws dynamodb create-table \
>   --table-name ContentCatalog \
>   --attribute-definitions AttributeName=Id,AttributeType=S \
>   --key-schema KeySchemaAttributeName=Id \
>   --provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1 \
>   --query "TableDescription.TableStatus"
CREATING
# aws dynamodb create-table \
>   --table-name Forum \
>   --attribute-definitions AttributeName=ForumName,AttributeType=S \
>   --key-schema KeySchemaAttributeName=ForumName \
>   --provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1 \
>   --query "TableDescription.TableStatus"
CREATING
# aws dynamodb create-table \
>   --table-name Post \
>   --attribute-definitions AttributeName=PostId,AttributeType=S \
>   --key-schema KeySchemaAttributeName=PostId \
>   --provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1 \
>   --query "TableDescription.TableStatus"
CREATING
# aws dynamodb create-table \
>   --table-name Comment \
>   --attribute-definitions AttributeName=CommentId,AttributeType=S \
>   --key-schema KeySchemaAttributeName=CommentId \
>   --provisioned-throughput ReadCapacityUnits=1,WriteCapacityUnits=1 \
>   --query "TableDescription.TableStatus"
- 1 |
```



Sanjana Tripathy
NextWork Student

nextwork.org

Loading Data with CLI

I ran CLI commands in AWS CloudShell using aws dynamodb batch-write-item to efficiently load data into four DynamoDB tables—ContentCatalog, Forum, Post, and Comment—by referencing structured JSON files for each.

```
'nextworksampled $ aws dynamodb batch-write-item --request-items file://ContentCatalog.json
{
    "UnprocessedItems": {}
}
nextworksampled $ aws dynamodb batch-write-item --request-items file://Forum.json
{
    "UnprocessedItems": {}
}
nextworksampled $ aws dynamodb batch-write-item --request-items file://Post.json
{
    "UnprocessedItems": {}
}
nextworksampled $ aws dynamodb batch-write-item --request-items file://Comment.json
{
    "UnprocessedItems": {}
}
nextworksampled $ █'
```

Observing Item Attributes

The screenshot shows the 'Edit item' interface for a ContentCatalog table. The interface is a form-based editor with the following details:

Attribute name	Type
Id - Partition key	Number
Authors	List
ContentType	String
Difficulty	String
Price	Number
ProjectCategory	String
Published	Boolean
Title	String
URL	String

Specific attribute values shown:

- Id - Partition key: 1
- Authors: Insert a field
- ContentType: Project
- Difficulty: Easy peasy
- Price: 0
- ProjectCategory: Storage
- Published: True
- Title: Host a Website on Amazon S3
- URL: aws-host-a-website-on-s3

Buttons at the bottom: Cancel, Save, Save and close.

The item from the ContentCatalog table included attributes such as Authors, ContentType, Difficulty, Price, ProjectCategory, Published, and Title. These attributes help categorize and describe each learning resource. Looks complete and well-structure

I checked another ContentCatalog item, which had a different set of attributes: ContentType, Price, Services, Title, URL and VideoType

Benefits of DynamoDB

A benefit of DynamoDB over relational databases is flexibility, because each item can have its own set of attributes. This is ideal when storing varied data types—like different products without needing every item to follow the same fixed structure.

Another benefit over relational databases is speed, because DynamoDB uses partition keys to efficiently locate and retrieve items. Unlike relational databases that may scan entire tables, DynamoDB's structure allows for faster access to specific data

Table: ContentCatalog - Items returned (6)											
Scan started on August 01, 2025, 22:44:33											
	ID (Number)	Authors	Content Type	Difficulty	Price	Project Category	Published	Services	Title	URL	Video Type
	1	{ "S": "Nex..." }	Project	Easy peasy	0	AI/ML	true	Build a Cha...	aws-ai-lex1		
	2	{ "S": "Nex..." }	Project	Easy peasy	0	Analytics	true	Visualize da...	aws-analyti...		
	203		Video		0			{ "S": "Am..." }	AWS x Data...	https://you...	Live Project Demo
	202		Video		0				Don't miss ...	https://you...	Shorts
	201		Video		0			{ "S": "Am..." }	AWS Relati...	https://you...	Educational
	1	{ "S": "Nat..." }	Project	Easy peasy	0	Storage	true	Host a Web...	aws-host-a...		



nextwork.org

The place to learn & showcase your skills

Check out nextwork.org for more projects

