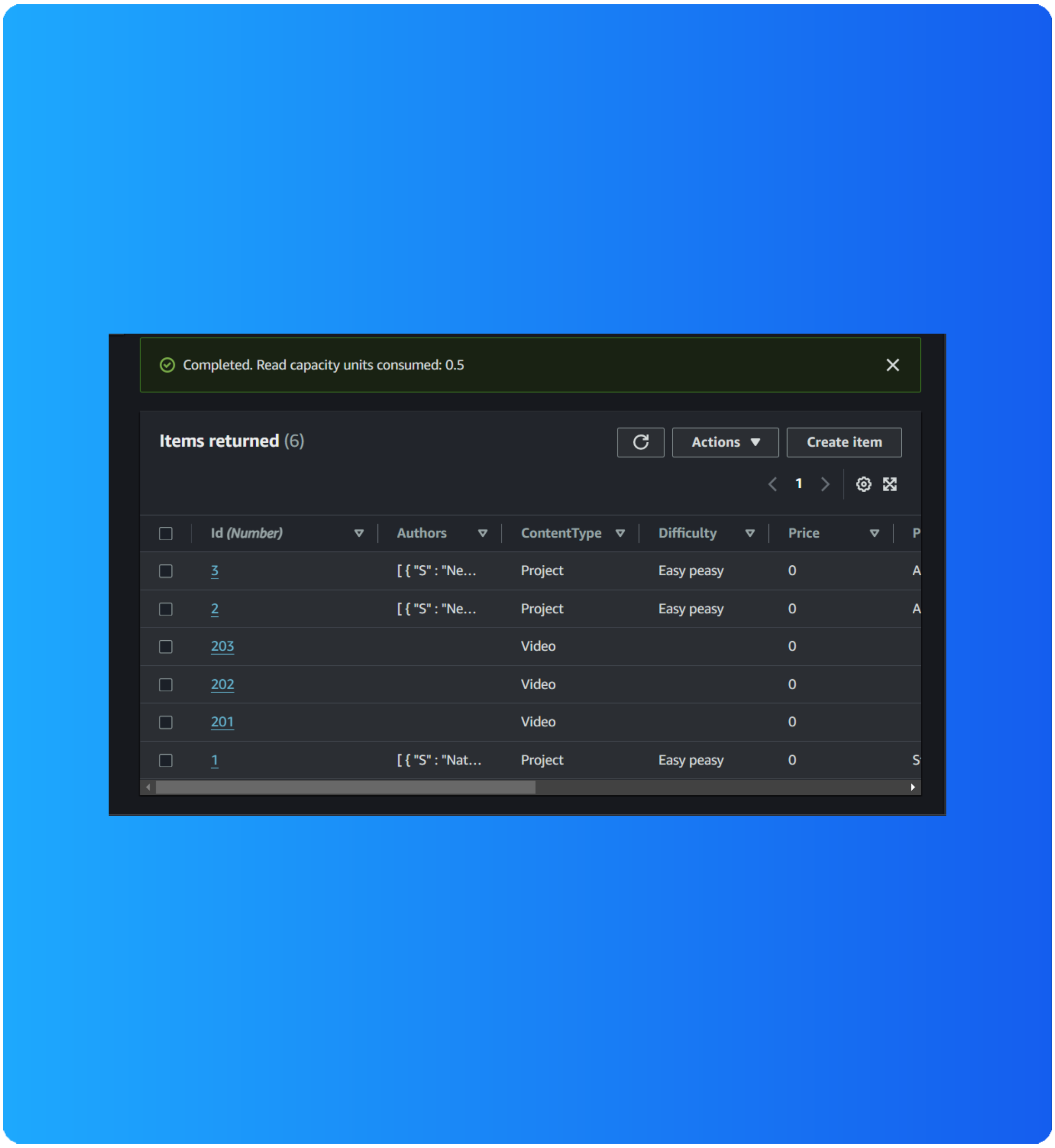
NextWork.org

Load Data Into DynamoDB

Praneeth Bhandwalkar



Completed. Read capacity units consumed: 0.5

Items returned (6)

☐	Id (Number)	Authors	ContentType	Difficulty	Price	P
☐	3	[{ "S" : "Ne...	Project	Easy peasy	0	A
☐	2	[{ "S" : "Ne...	Project	Easy peasy	0	A
☐	203		Video		0	
☐	202		Video		0	
☐	201		Video		0	
☐	1	[{ "S" : "Nat...	Project	Easy peasy	0	S

Introducing Today's Project!

What is Amazon DynamoDB?

Amazon DynamoDB is a fully managed NoSQL database service designed for fast, predictable performance with single-digit millisecond response times. It supports both key-value and document data structures, making it versatile for various applications.

How I used Amazon DynamoDB in this project

In this project, I used DynamoDB to create tables. And load data quickly & efficiently into it by using the AWS CloudShell and CLI.

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

One thing we didn't expect in this project we could use to create a table in DynamoDB with AWS CloudShell CLI which is very helpful for CRUD operations for creation of tables, and it's very handy to use.

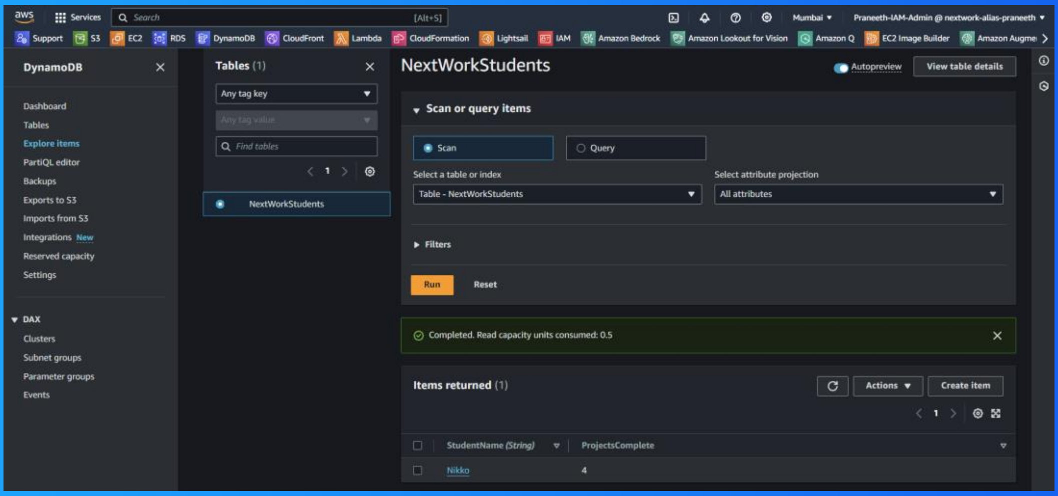
This project took me...

It took me 60 minutes to complete this project.

Create a DynamoDB table

DynamoDB tables organize data using items and attributes as the list of items Student, each with its list of attributes ProjectsComplete by Student.

An attribute is like a piece of data about an item. For example, our item is Student and the attribute is the number of projects Student completed.



● NextWorkStudents

NextWorkStudents

● Scan ○ Query

Select a table or index: Table - NextWorkStudents

Select attribute projection: All attributes

Completed. Read capacity units consumed: 0.5

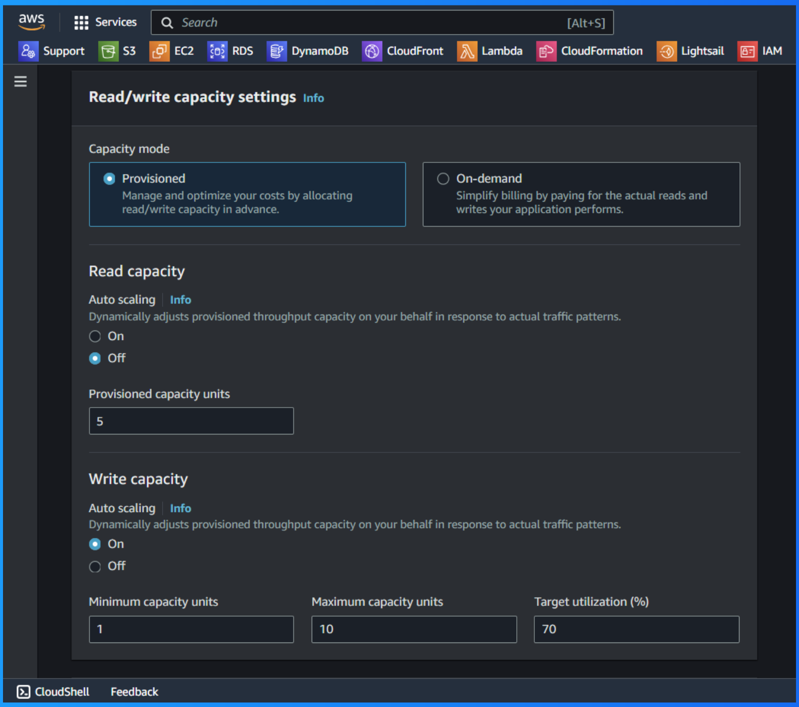
Items returned (1)

□	StudentName (String)	ProjectsComplete
□	Nikko	4

Read and Write Capacity

Read capacity units (RCUs) and write capacity units (WCUs) are units that measure the DynamoDB table's performance. and DynamoDB pricing is based on these units where RCU has 1 read capacity unit (RCU) and WCU has 1 item write/second.

Amazon DynamoDB's Free Tier covers RCU's and WCU's monthly. I turned off auto-scaling because it can help us reduce our costs without getting higher charges.



Read/write capacity settings Info

Capacity mode

● Provisioned
Manage and optimize your costs by allocating read/write capacity in advance.

○ On-demand
Simplify billing by paying for the actual reads and writes your application performs.

Read capacity

Auto scaling | Info
Dynamically adjusts provisioned throughput capacity on your behalf in response to actual traffic patterns.

○ On
● Off

Provisioned capacity units
5

Write capacity

Auto scaling | Info
Dynamically adjusts provisioned throughput capacity on your behalf in response to actual traffic patterns.

● On
○ Off

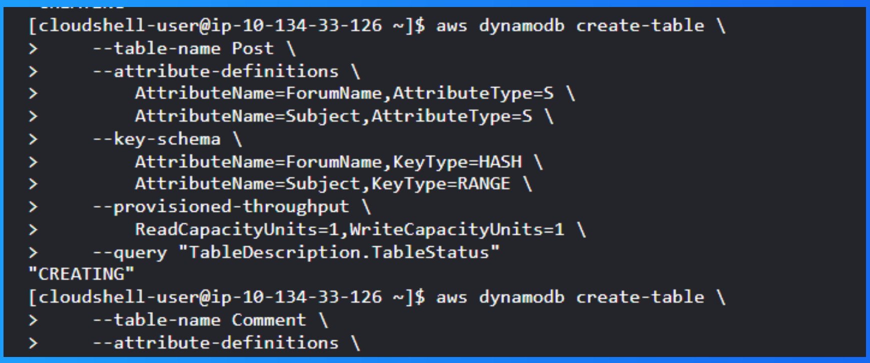
Minimum capacity units	Maximum capacity units	Target utilization (%)
1	10	70

Using CLI and CloudShell

AWS CloudShell is a shell in your AWS Management Console, which means it's a space for you to run code! The awesome thing about AWS CloudShell is that it already has AWS CLI pre-installed.

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

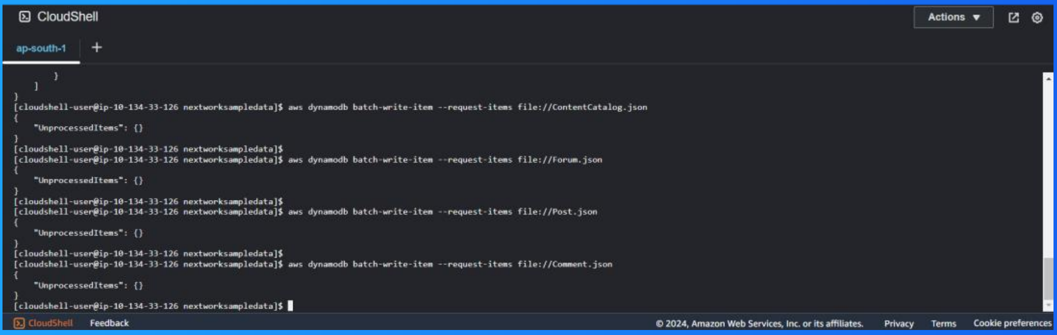
I ran a CLI command in AWS CloudShell that created four new DynamoDB tables called ContentCatalog Table, Forum Table, Post Table, and Comment Table. The command AWS DynamoDB creates-table and its attribute within the command.



```
[cloudshell-user@ip-10-134-33-126 ~]$ aws dynamodb create-table \
>     --table-name Post \
>     --attribute-definitions \
>         AttributeName=ForumName,AttributeType=S \
>         AttributeName=Subject,AttributeType=S \
>     --key-schema \
>         AttributeName=ForumName,KeyType=HASH \
>         AttributeName=Subject,KeyType=RANGE \
>     --provisioned-throughput \
>         ReadCapacityUnits=1,WriteCapacityUnits=1 \
>     --query "TableDescription.TableStatus"
"CREATING"
[cloudshell-user@ip-10-134-33-126 ~]$ aws dynamodb create-table \
>     --table-name Comment \
>     --attribute-definitions \
```

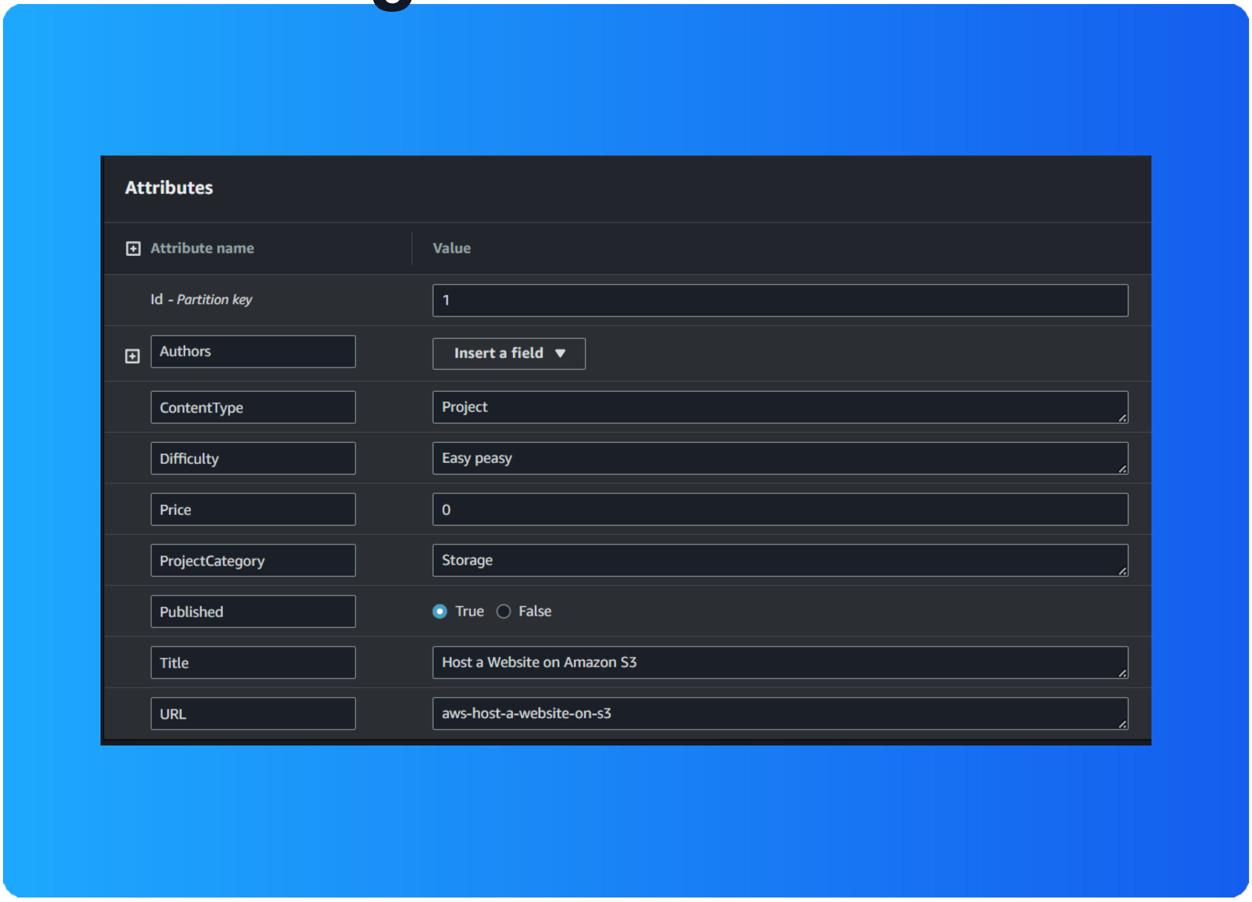
Loading Data with CLI

I ran a CLI command in AWS CloudShell that loads multiple items into the DynamoDB table. and Loaded the data of all four files into DynamoDB using AWS CLI's "batch-write-item" commands.



```
        }
    ]
}
[cloudshell-user@ip-10-134-33-126 nextworksampledata]$ aws dynamodb batch-write-item --request-items file://ContentCatalog.json
{
    "UnprocessedItems": {}
}
[cloudshell-user@ip-10-134-33-126 nextworksampledata]$
[cloudshell-user@ip-10-134-33-126 nextworksampledata]$ aws dynamodb batch-write-item --request-items file://Forum.json
{
    "UnprocessedItems": {}
}
[cloudshell-user@ip-10-134-33-126 nextworksampledata]$
[cloudshell-user@ip-10-134-33-126 nextworksampledata]$ aws dynamodb batch-write-item --request-items file://Post.json
{
    "UnprocessedItems": {}
}
[cloudshell-user@ip-10-134-33-126 nextworksampledata]$
[cloudshell-user@ip-10-134-33-126 nextworksampledata]$ aws dynamodb batch-write-item --request-items file://Comment.json
{
    "UnprocessedItems": {}
}
[cloudshell-user@ip-10-134-33-126 nextworksampledata]$ 
```

Observing Item Attributes



Attributes

Attribute name	Value
Id - *Partition key*	1
Authors	Insert a field ▼
ContentType	Project
Difficulty	Easy peasy
Price	0
ProjectCategory	Storage
Published	● True ○ False
Title	Host a Website on Amazon S3
URL	aws-host-a-website-on-s3

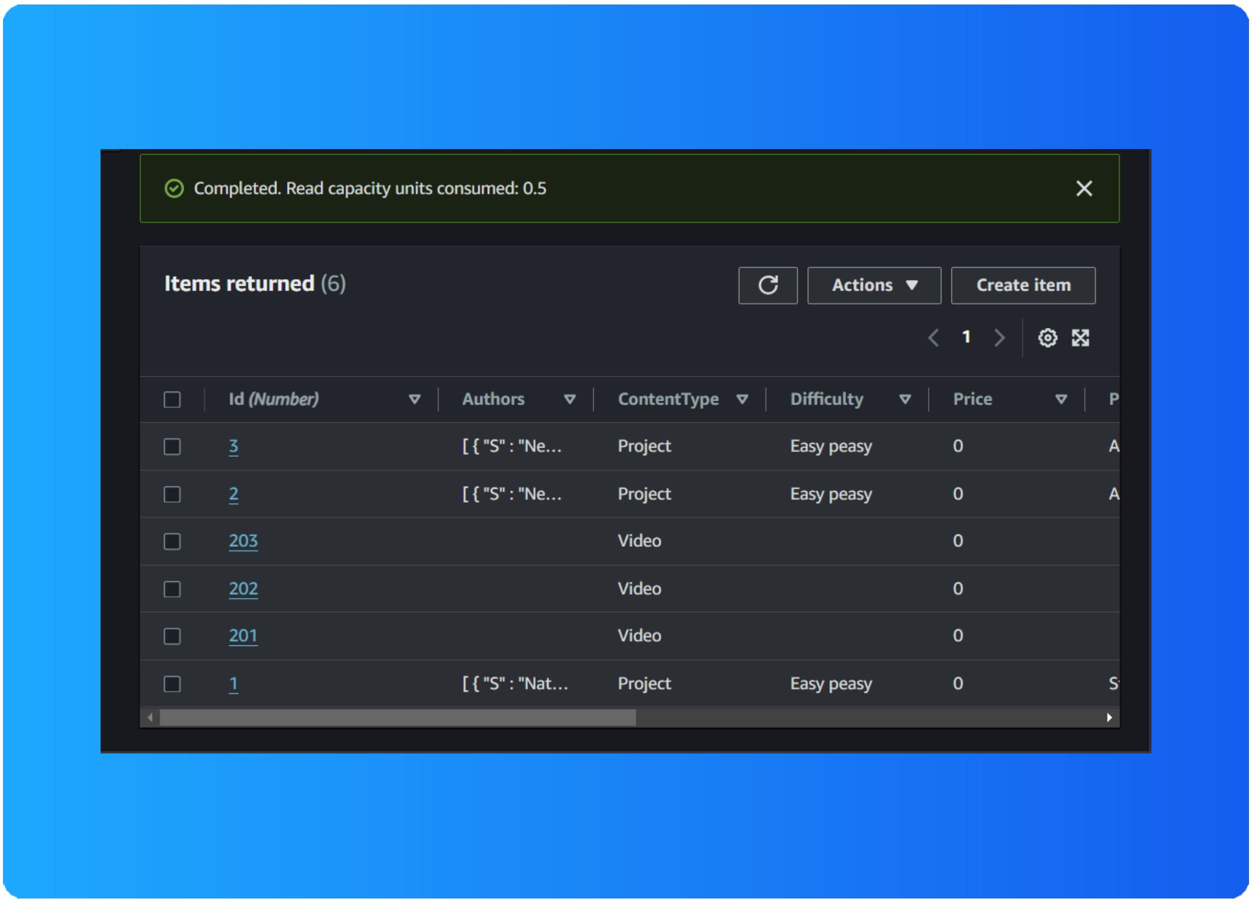
I checked a ContentCatalog item, which had the following attributes: ContentType, Difficulty, ProjectCategory, VideoType, Title and URL.

I checked another ContentCatalog item, which had a different set of attributes: Services, Title, VideoType, etc.

Benefits of DynamoDB

A benefit of DynamoDB over relational databases is flexibility because items can have their own set of attributes. This is great for scenarios where a table has different types of data and some attributes don't apply.

Another benefit over relational databases is speed because DynamoDB finds quickly using partition keys which is faster than relational databases which scan entire tables to find specific pieces of data quickly.



Completed. Read capacity units consumed: 0.5

Items returned (6)

□	Id *(Number)*	Authors	ContentType	Difficulty	Price	P
□	3	[{ "S" : "Ne...	Project	Easy peasy	0	A
□	2	[{ "S" : "Ne...	Project	Easy peasy	0	A
□	203		Video		0	
□	202		Video		0	
□	201		Video		0	
□	1	[{ "S" : "Nat...	Project	Easy peasy	0	S

Everyone should be in a job they love.

Check out nextwork.org for more projects