



Query Data with DynamoDB



nikhil7_94@hotmail.com

```
[cloudshell-user@ip-10-140-2-228 ~]$ aws dynamodb get-item \
>   --table-name ContentCatalog \
>   --key '{"Id":{"N":"202"}}' \
>   --projection-expression "Title, ContentType, Services" \
>   --return-consumed-capacity TOTAL
{
    "Item": {
        "Title": {
            "S": "Don't miss out!"
        },
        "ContentType": {
            "S": "Video"
        }
    },
    "ConsumedCapacity": {
        "TableName": "ContentCatalog",
        "CapacityUnits": 0.5
    }
}
```

Introducing Today's Project!

What is Amazon DynamoDB?

DynamoDB is described as a NoSQL database or a key-value database. A NoSQL database is useful because it means you would not use SQL to query it while key-value is a specific way to store data that's flexible and efficient.

How I used Amazon DynamoDB in this project

Today's project I have learned to upload tables and data into DynamoDB using CloudShell, query DyanmoDB through the console and CloudShell and set up a transaction.

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

One thing I didnt expect in this project was that you can interact and make queries so easily with the Cloud powershell.

This project took me...

I took around 2 hours with this project.

Querying DynamoDB Tables

A partition key is a primary key attribute that determines how data is distributed across storage partitions for scalability and performance.

A sort key is a secondary key used to filter your query results again. Sort keys work after the partition key i.e. you still have to use the partition key to split up your data first and then the sort key partitions your data again.

▼ Scan or query items

Scan Query

Select a table or index
Table - Comment

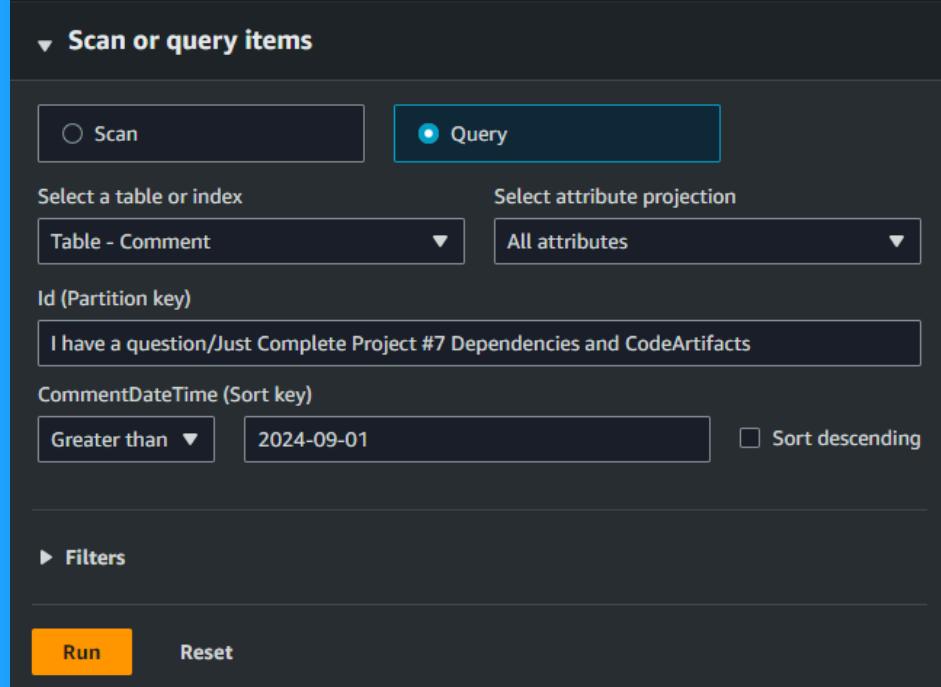
Select attribute projection
All attributes

Id (Partition key)
I have a question/Just Complete Project #7 Dependencies and CodeArtifacts

CommentDateTime (Sort key)
Greater than ▾ 2024-09-01 Sort descending

▶ Filters

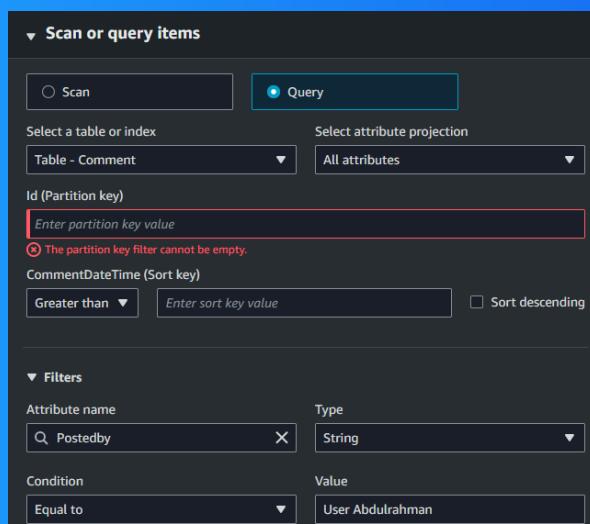
Run **Reset**



Limits of Using DynamoDB

I ran into an error when I queried for all posts by user called Abdulrahman. This was because you have to use the Id (Partition key) when I query items.

Insights we could extract from our Comment table includes partition key using filters.



Running Queries with CLI

A query I ran in CloudShell was in the ContentCatalog table with partition key as Id 202 that will give me the items of "Title, ContentType and Services".

Query options I could add to my query are filters, projection and key condition.

```
[cloudshell-user@ip-10-140-2-228 ~]$ aws dynamodb get-item \
>   --table-name ContentCatalog \
>   --key '{"Id":{"N":"202"}}' \
>   --projection-expression "Title, ContentType, Services" \
>   --return-consumed-capacity TOTAL
{
    "Item": {
        "Title": {
            "S": "Don't miss out!"
        },
        "ContentType": {
            "S": "Video"
        }
    },
    "ConsumedCapacity": {
        "TableName": "ContentCatalog",
        "CapacityUnits": 0.5
    }
}
```

Transactions

A transaction is a group of operations that all have to succeed. If any of the operations in the group fails, none of the changes get applied.

I ran a transaction using a command in the AWS CLI. This transaction did two things, first updated both the Comment and Forum table and then updated the number of comments in the Events table.

```
cloudshell-user@ip-10-140-110-101 ~]$ aws dynamodb transact-write-items --client-request-token TRANSACTION1 --transact-items '[  
    {  
        "Put": {  
            "TableName" : "Comment",  
            "Item" : {  
                "Id" : {"S": "Events/Do a Project Together - NextWork Study Session"},  
                "CommentDateTime" : {"S": "2024-9-27T17:47:30Z"},  
                "Comment" : {"S": "Excited to attend!"},  
                "PostedBy" : {"S": "User Connor"}  
            }  
        },  
        {  
            "Update": {  
                "TableName" : "Forum",  
                "Key" : {"Name" : {"S": "Events"}},  
                "UpdateExpression": "ADD Comments :inc",  
                "ExpressionAttributeValues" : { ":inc": {"N" : "1"} }  
            }  
        }  
    }'  
cloudshell-user@ip-10-140-110-101 ~]$ 
```



NextWork.org

Everyone should be in a job they love.

Check out nextwork.org for
more projects

