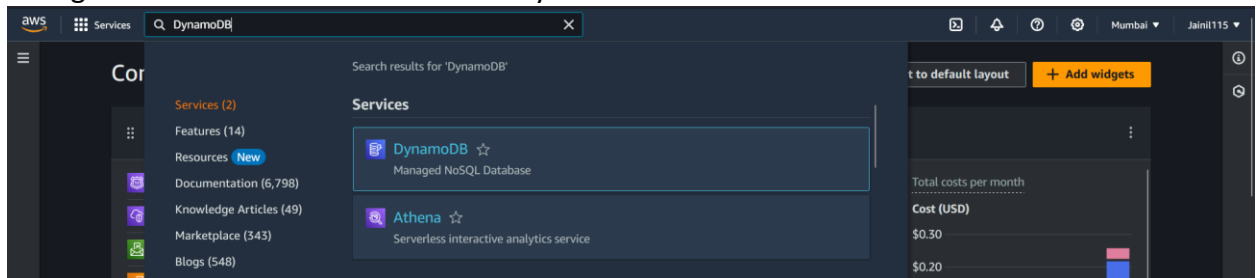


## TASK 2: Create a DynamoDB Table:

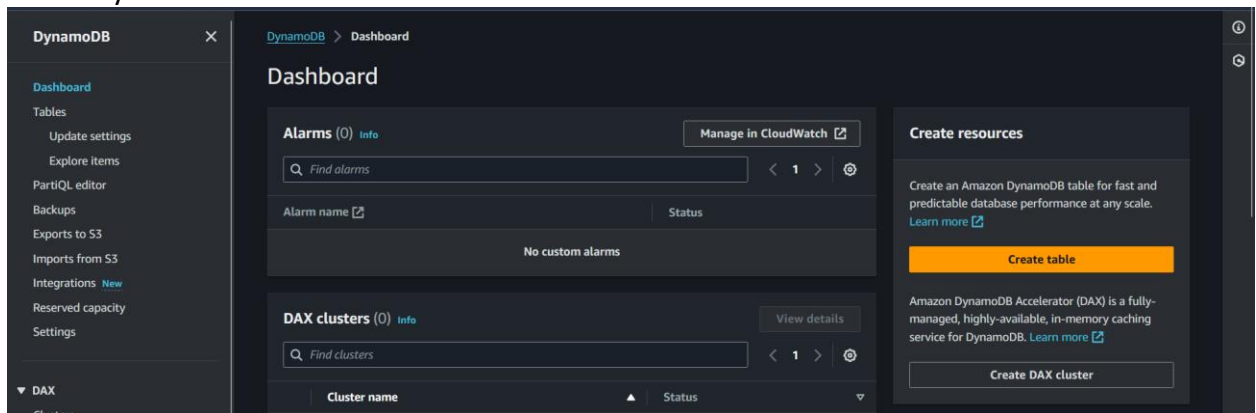
1. Create a new DynamoDB table with a primary key of your choice.
2. Define the provisioned throughput for the table.
3. Run CRUD operation in the DB table using Queries.

Steps to create a new DynamoDB Table with a primary key of your choice:

1. First go to AWS console and search for DynamoDB.



2. Inside DynamoDB dashboard click on "create table".



3. Inside create table form, Enter the following details.
  - i. Table Name: Books
  - ii. Partition Key: BookID, Type: String
  - iii. Sort Key: Author, Type: String

**Create table**

**Table details** [info](#)  
DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

**Table name**  
This will be used to identify your table.  
  
Between 3 and 255 characters, containing only letters, numbers, underscores (\_), hyphens (-), and periods (.).

**Partition key**  
The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.  
   
1 to 255 characters and case sensitive.

**Sort key - optional**  
You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.  
   
1 to 255 characters and case sensitive.

- Then select Default settings under Table settings and click on Create Table

**Table settings**

☒ **Default settings**  
The fastest way to create your table. You can modify these settings now or after your table has been created.

☐ **Customize settings**  
Use these advanced features to make DynamoDB work better for your needs.

**Default table settings**  
These are the default settings for your new table. You can change some of these settings after creating the table.

Setting	Value	Editable after creation
Table class	DynamoDB Standard	Yes
Capacity mode	Provisioned	Yes
Provisioned read capacity	5 RCU	Yes
Provisioned write capacity	5 WCU	Yes
Auto scaling	On	Yes
Local secondary indexes	-	No
Global secondary indexes	-	Yes
Encryption key management	Owned by Amazon DynamoDB	Yes
Deletion protection	Off	Yes

- Now in DynamoDB Table Dashboard you will be able to see that Books Table has been created.

**DynamoDB**

**The Books table was created successfully.**

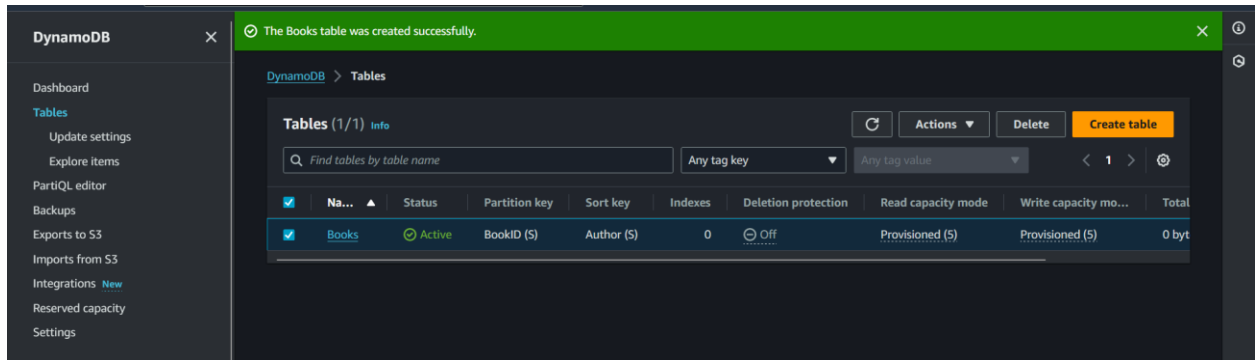
**DynamoDB > Tables**

**Tables (1)** [info](#)

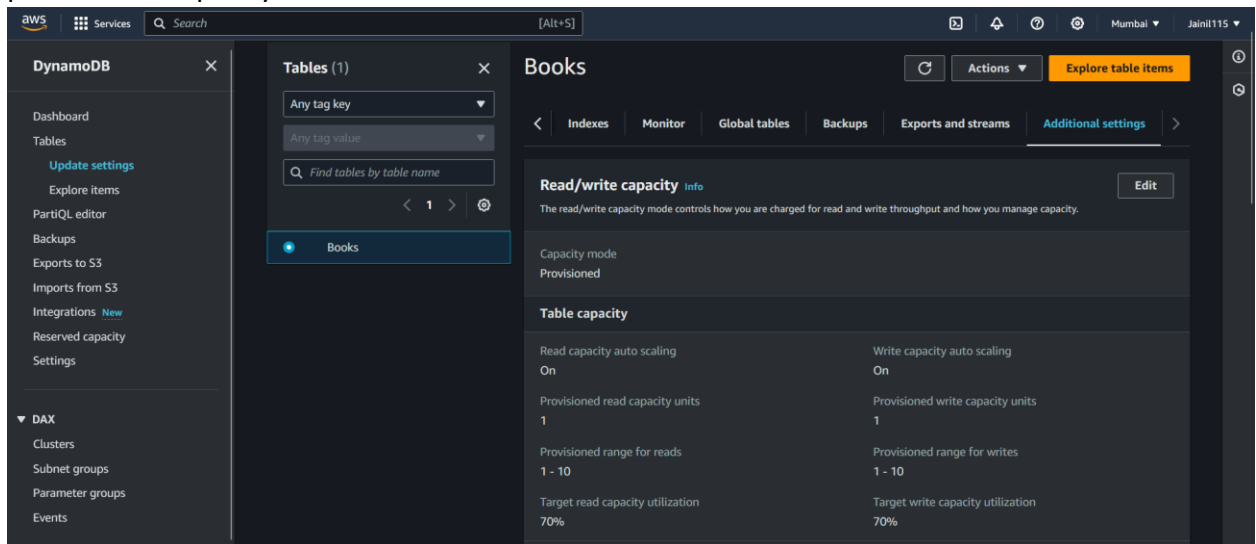
	Na...	Status	Partition key	Sort key	Indexes	Deletion protection	Read capacity mode	Write capacity mo...	Total
<input type="checkbox"/>	Books	Active	BookID (S)	Author (S)	0	Off	Provisioned (S)	Provisioned (S)	0 byt

## Steps to define the provisioned throughput for the table:

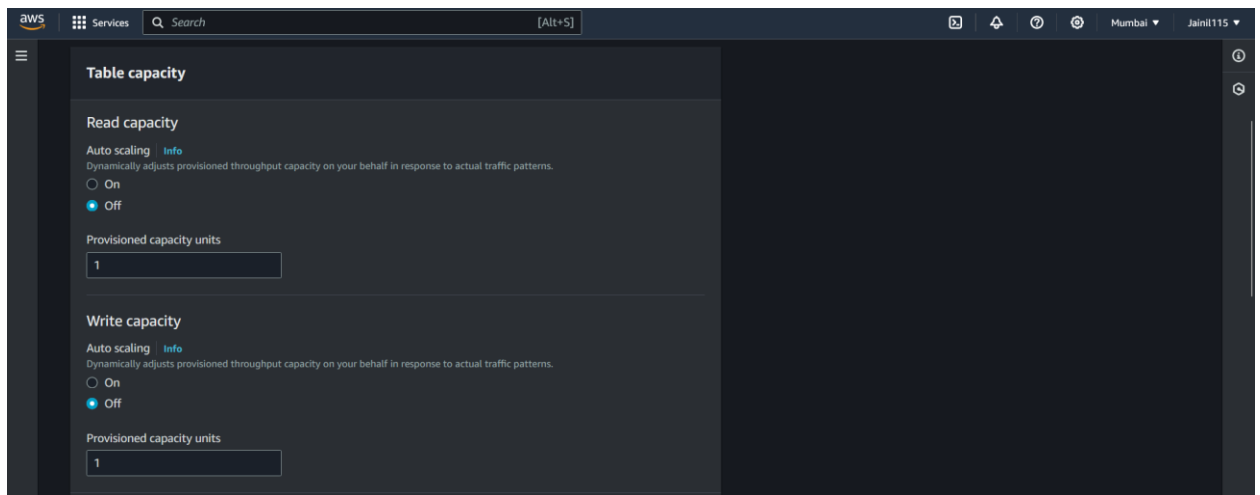
1. Go to Tables under DynaomODB dashboard and then clickN Books table.



2. Now inside DynamoDB > Table > Books go to additional settings. Where we can see the provisioned capacity for the table.

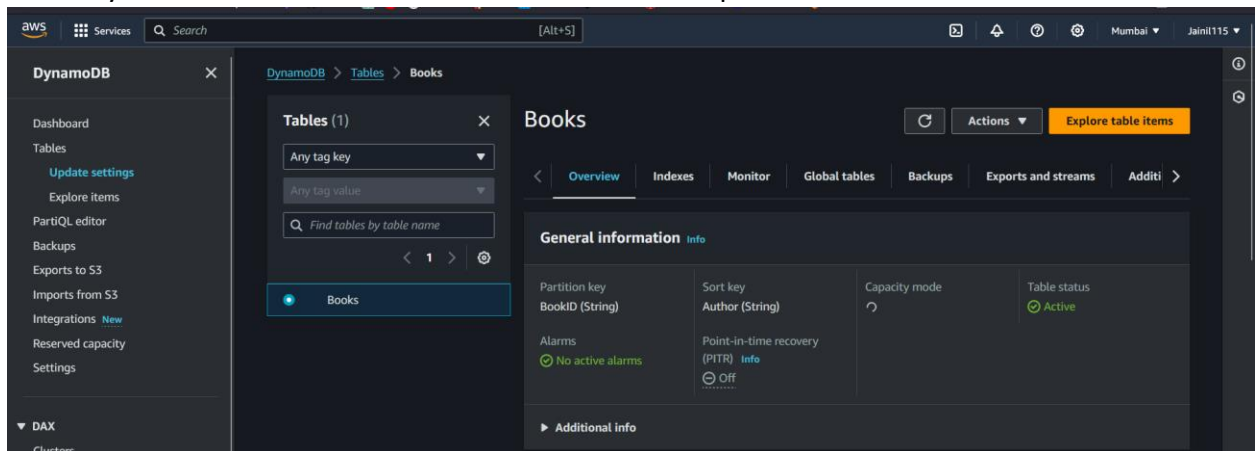


3. Click on edit read/write capacity, Now, we can change table provisioned read write capacity. After that turn off Auto Scaling. And then enter the provisioned capacity units. In my case I will keep it as 1 (because I don't want to incur any additional charges). And then click on save changes.

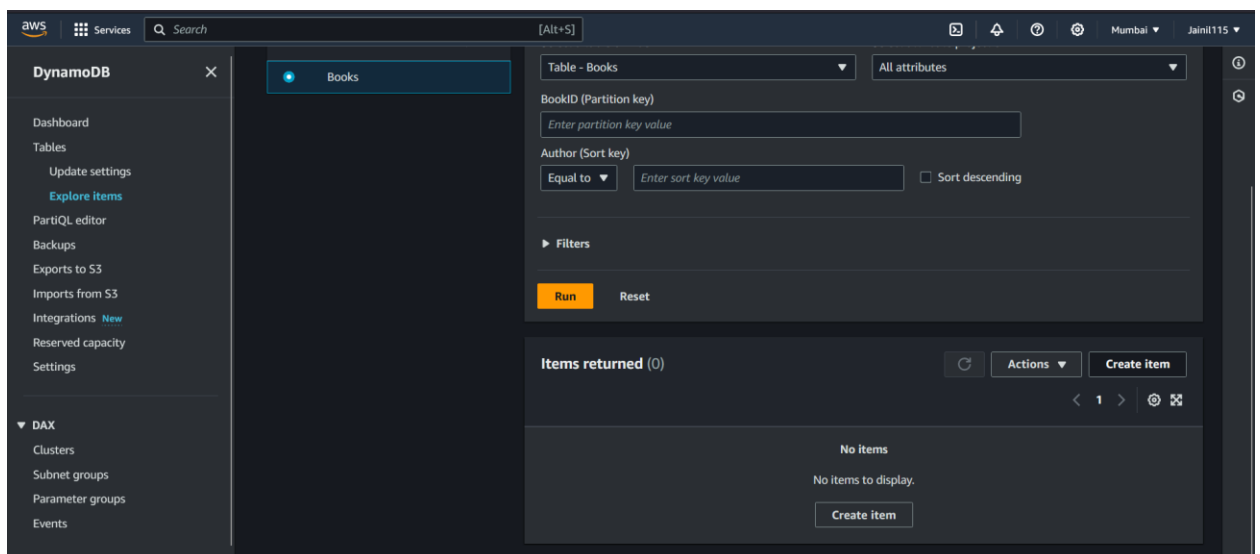


## Steps to run CRUD operation in the DB table using Queries:

1. Go to DynamoDB > Table > Books and then click on "Explore table items".

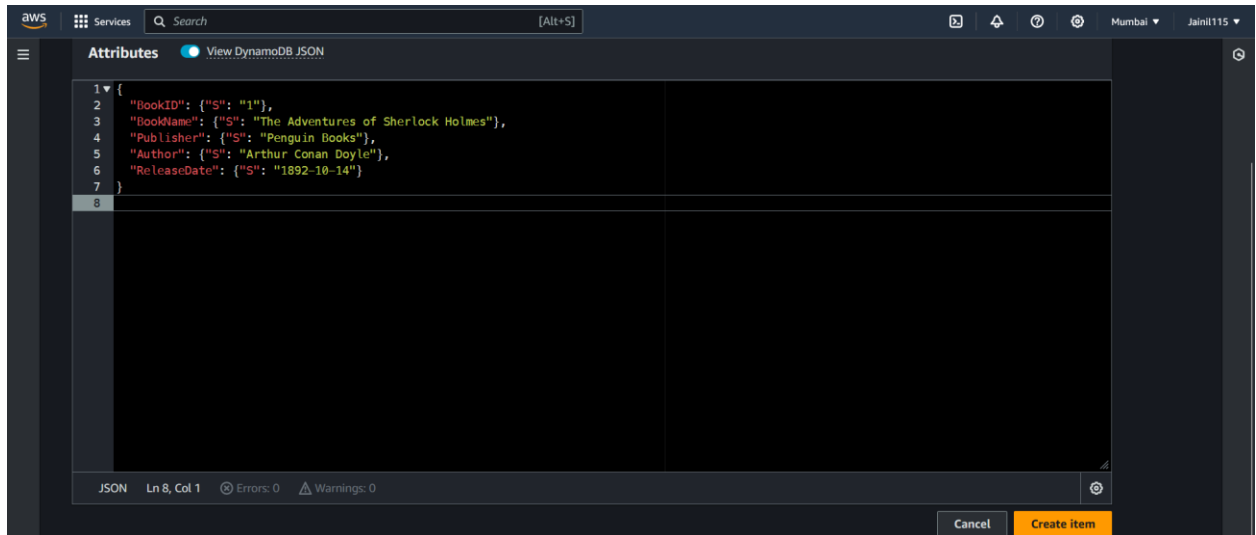


2. Then scroll down and click on create item.



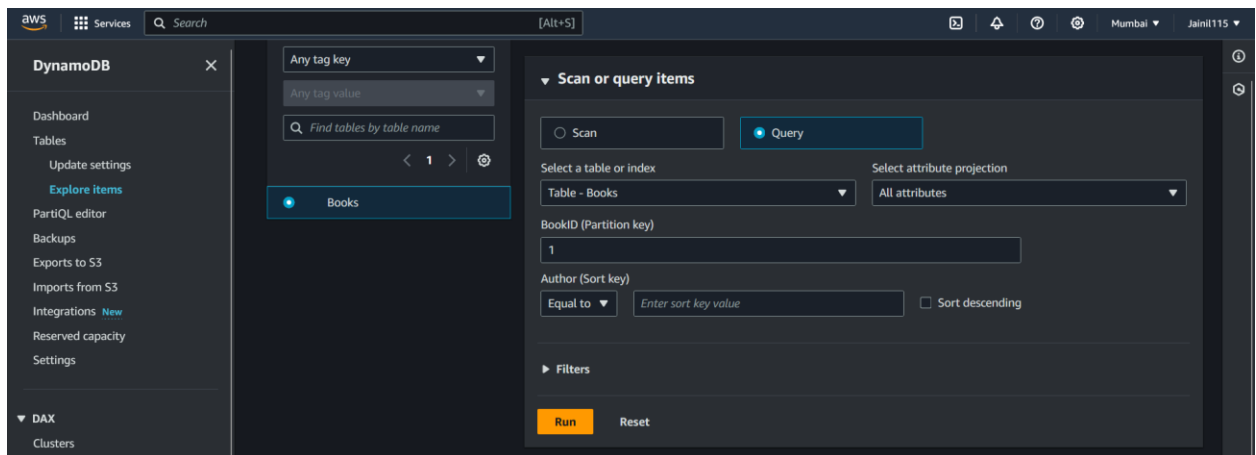
3. Now select JSON View, and then enter the item details. Then click on create item.

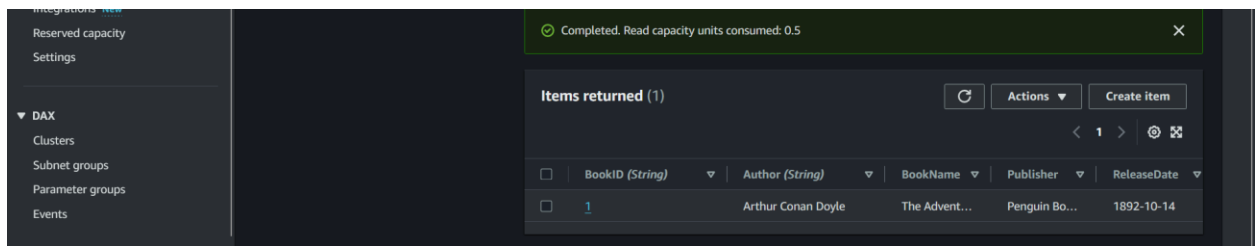
```
{
  "BookID": {"S": "1"},
  "BookName": {"S": "The Adventures of Sherlock Holmes"},
  "Publisher": {"S": "Penguin Books"},
  "Author": {"S": "Arthur Conan Doyle"},
  "ReleaseDate": {"S": "1892-10-14"}
}
```



## To Read Item using AWS console:

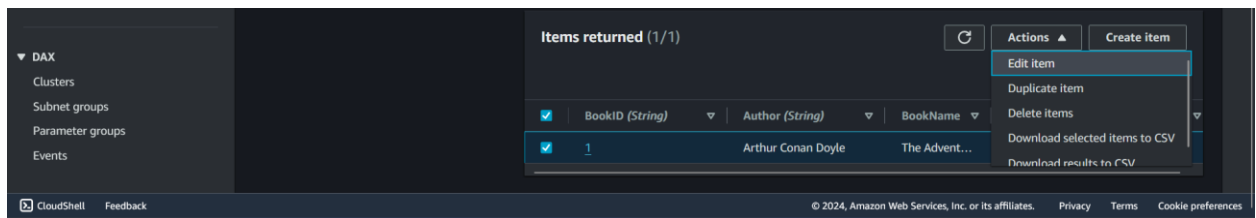
1. Inside DynamoDB > Explore Items > Books. Click on query and enter the BookID = 1. You will be able to see book details with BookID = 1.



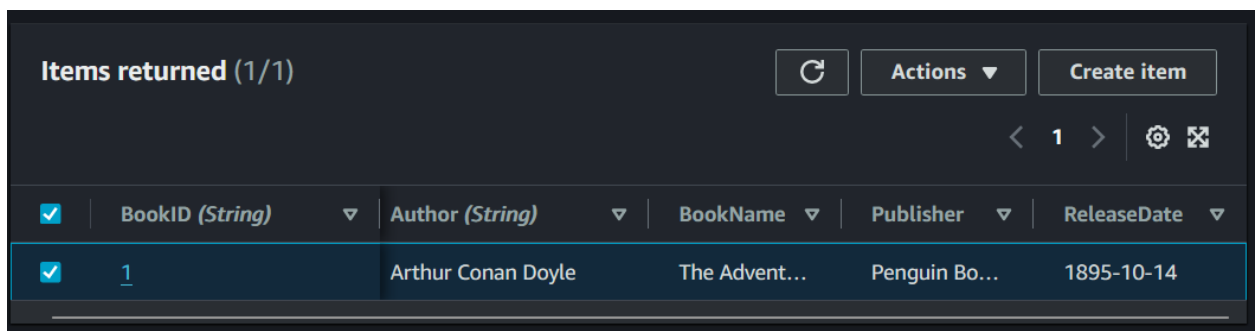
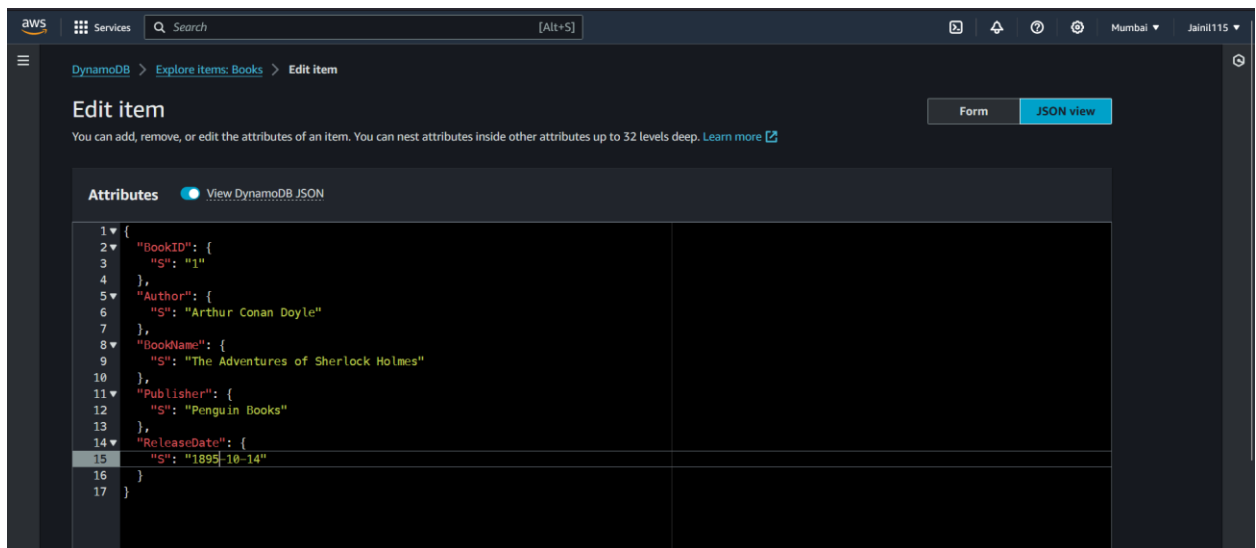


## Update Items using AWS Console:

1. Select the item you want to update and then click on action, under action select edit item.

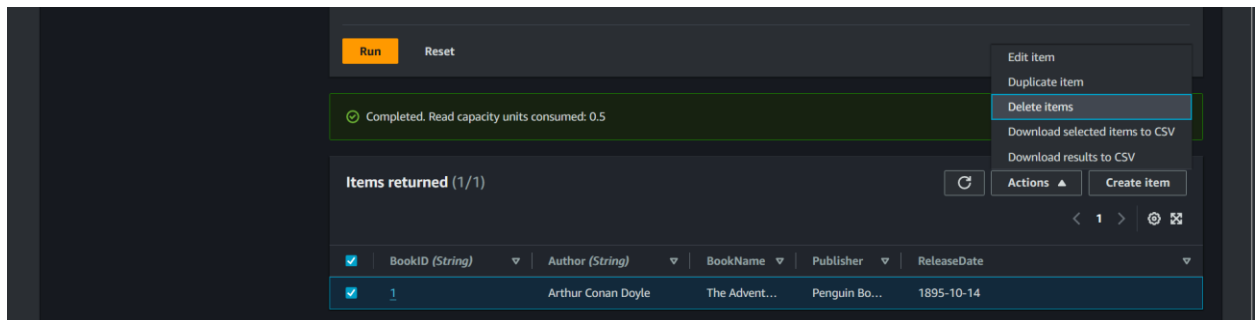


2. Now select JSON View, and change the details that needs to be updated, In my case I change the releaseDate from 1892 to 1895. After that click on save and close to update the item.

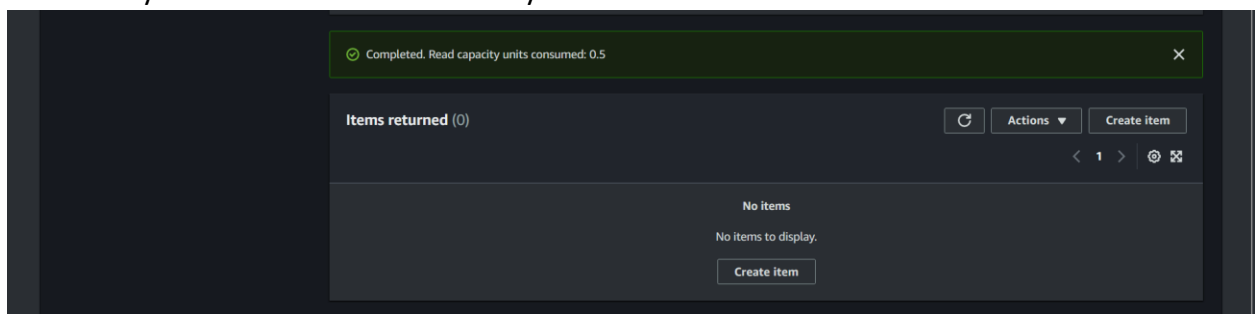


## Delete item using AWS Console:

1. Select the item you want to delete, and then click on action and then select delete items to delete selected item.



2. After that you will be able to see that my item with BookID=1 is deleted.

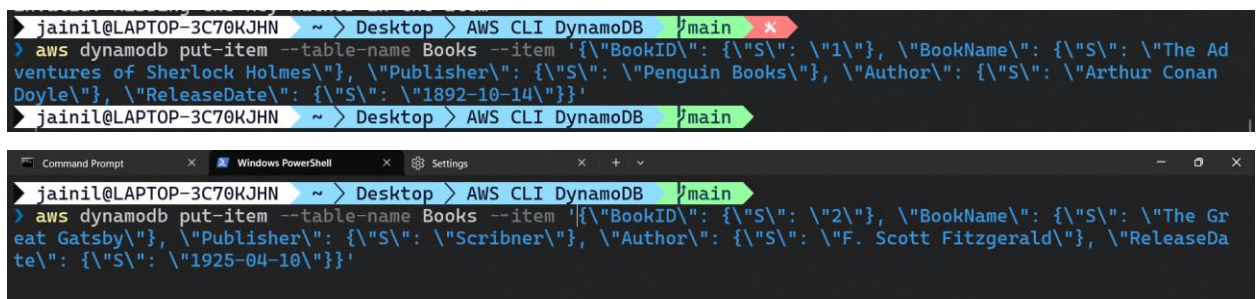


## Steps for CRUD Operations using AWS CL:

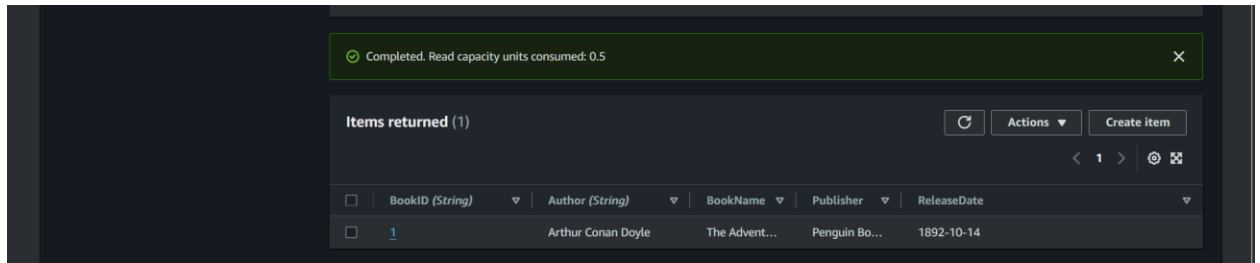
1. Open Powershell/Command Prompt on windows and enter the following for creating items in Books table.

```
aws dynamodb put-item --table-name Books --item '{"BookID": {"S": "1"}, "BookName": {"S": "The Adventures of Sherlock Holmes"}, "Publisher": {"S": "Penguin Books"}, "Author": {"S": "Arthur Conan Doyle"}, "ReleaseDate": {"S": "1892-10-14"}}'
```

```
aws dynamodb put-item --table-name Books --item '{"BookID": {"S": "2"}, "BookName": {"S": "The Great Gatsby"}, "Publisher": {"S": "Scribner"}, "Author": {"S": "F. Scott Fitzgerald"}, "ReleaseDate": {"S": "1925-04-10"}}'
```



2. Check that the item is created.



### Steps to create READ operation:

1. Enter the following code in the command line

```
aws dynamodb get-item --table-name Books --key '{"BookID": {"S": "1"}, "Author": {"S": "Arthur Conan Doyle"}}'
```

```
jainil@LAPTOP-3C70KJHN ~ > Desktop > AWS CLI DynamoDB /main *
> aws dynamodb get-item --table-name Books --key '{"BookID": {"S": "1"}, "Author": {"S": "Arthur Conan Doyle"}}'
{
  "Item": {
    "Publisher": {
      "S": "Penguin Books"
    },
    "BookID": {
      "S": "1"
    },
    "Author": {
      "S": "Arthur Conan Doyle"
    },
    "BookName": {
      "S": "The Adventures of Sherlock Holmes"
    },
    "ReleaseDate": {
      "S": "1892-10-14"
    }
  }
}
```

### Steps to update items in Books table:

1. Enter the following command to update BookName.

```
aws dynamodb update-item --table-name Books --key '{"BookID": {"S": "1"}, "Author": {"S": "Sir Arthur Conan Doyle"}}' --update-expression "SET BookName = :value" --expression-attribute-values '{":value": {"S": "The Shinning"}}' --return-values UPDATED_NEW
```



```
jainil@LAPTOP-3C70KJHN ~ > Desktop > AWS CLI DynamoDB /main
> aws dynamodb update-item --table-name Books --key '{"BookID": {"S": "1"}, "Author": {"S": "Sir Arthur Conan Doyle"}}' --update-expression "SET BookName = :value" --expression-attribute-values '{"S": "The Shinning"}' --return-values UPDATED_NEW
{
  "Attributes": {
    "BookName": {
      "S": "The Shinning"
    }
  }
}
```

```
jainil@LAPTOP-3C70KJHN ~ > Desktop > AWS CLI DynamoDB /main
>
```

<input type="checkbox"/>	BookID (String)	Author (String)	BookName
<input type="checkbox"/>	<u>1</u>	Sir Arthur Conan Doyle	The Shinning

### Steps to delete item in DynamoDB:

1. Enter the following command to delete item from the table:

```
aws dynamodb delete-item --table-name Books --key '{"BookID": {"S": "1"}, "Author": {"S": "Sir Arthur Conan Doyle"}}'
```

```
jainil@LAPTOP-3C70KJHN ~ > Desktop > AWS CLI DynamoDB /main
> aws dynamodb delete-item --table-name Books --key '{"BookID": {"S": "1"}, "Author": {"S": "Sir Arthur Conan Doyle"}}'
jainil@LAPTOP-3C70KJHN ~ > Desktop > AWS CLI DynamoDB /main
>
```

<input type="checkbox"/>	BookID (String)	Author (String)	BookName	Publisher	ReleaseDate
<input type="checkbox"/>	<u>2</u>	F. Scott Fitzgerald	The Great G...	Scribner	1925-04-10