

Task1: 1:Create a dynamo db table with minimum two disaster recovery zones and verify replication.

N.California and Sydney are the disaster recovery regions.

The screenshot shows the AWS DynamoDB console for the 'movies' table. The 'Global Tables' tab is selected, displaying the configuration for a multi-region, multi-master database. The table is configured with three regions: Asia Pacific (Sydney), US East (N. Virginia), and US West (N. California). All regions are active and have read and write capacity units of 5. The IAM role is 'AWSServiceRoleForDynamoDBReplication'.

Region Name	Status	Read capacity units	Write capacity units	Auto Scaling	Endpoint
Asia Pacific (Sydney)	Active	5	5	READ_AND_WRITE	dynamodb.ap-southeast-2.amazonaws.com
US East (N. Virginia)	Active	5	5	READ_AND_WRITE	dynamodb.us-east-1.amazonaws.com
US West (N. California)	Active	5	5	READ_AND_WRITE	dynamodb.us-west-1.amazonaws.com

Items replicated in the N.California from the original region north virginia

The screenshot shows the AWS DynamoDB console for the 'movies' table, specifically the 'Items' tab. The table is configured with three regions: Asia Pacific (Sydney), US East (N. Virginia), and US West (N. California). The 'Items' tab shows a list of items, including 'Actors' and 'songs'.

Actors	songs
Ak	mama
Mo	puchayakaya
NTR	Tring Tring
SDT	pilla nuvvu leni
pawan kalyan	hey pilla

Items replicated in the sydney region from the original region north virginia

The screenshot shows the AWS DynamoDB console for the 'movies' table in the Sydney region. The 'Items' tab is active, showing a list of items. The table has a primary key 'Actors' and a secondary index 'songs'. The items are as follows:

Actors	songs
Ak	mama
Mb	puchayakaya
NTR	Tring Tring
SDT	pilla nuvvu leni
pawan kalyan	hey pilla

using the query to fetch the items.

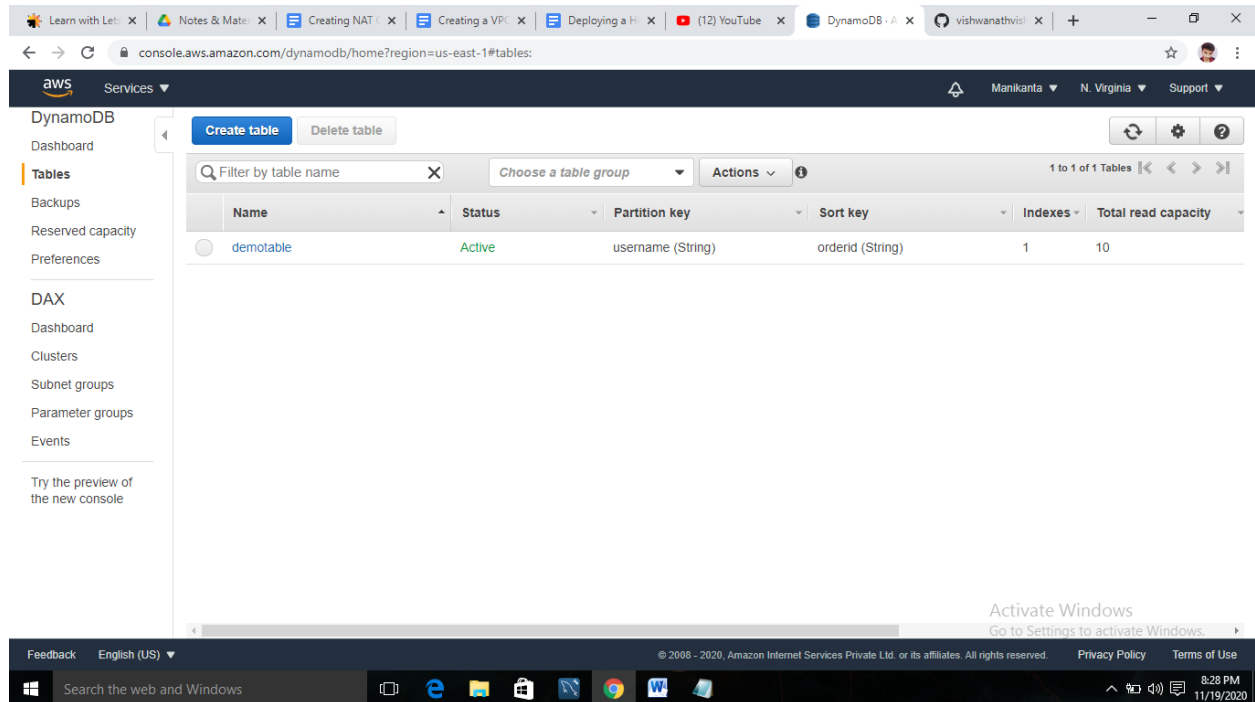
The screenshot shows the AWS DynamoDB console for the 'movies' table in the N. Virginia region. The 'Query' tab is active, showing a query result for the 'songs' index. The query is configured with the following parameters:

- Partition key: Actors (String) = Mb
- Sort key: songs (String) = puchayakaya
- Sort: Ascending
- Attributes: All

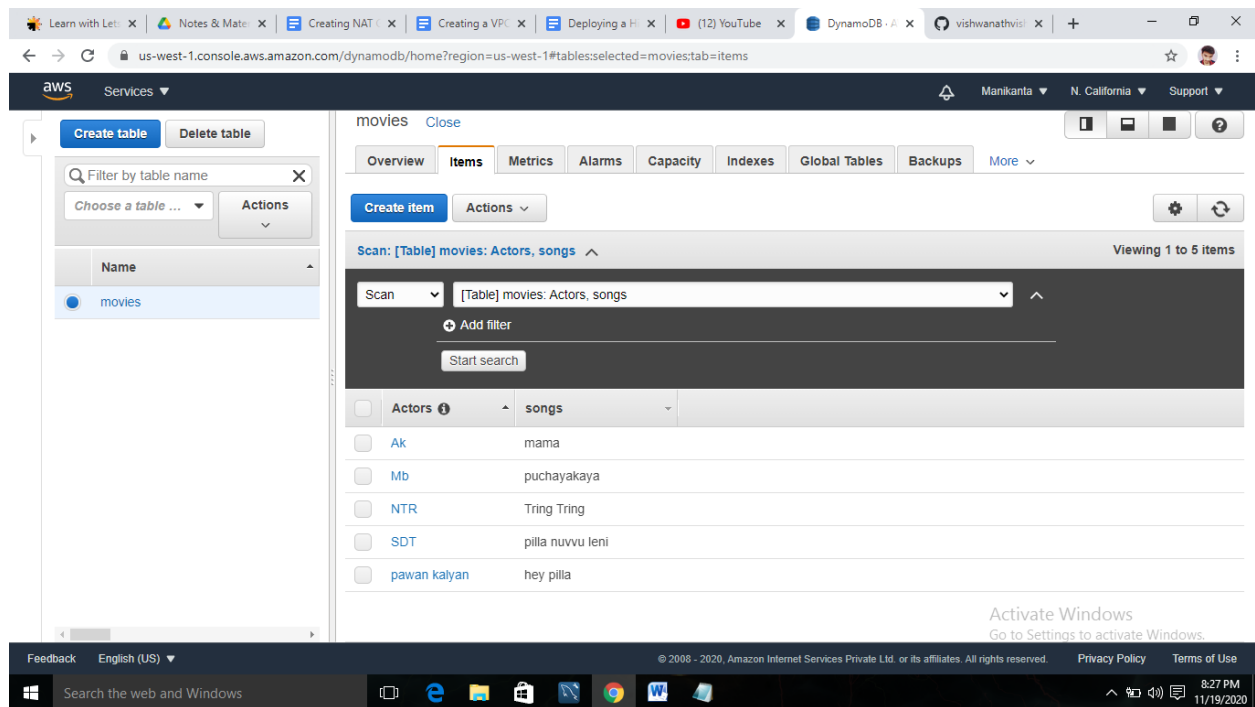
The query result shows one item:

Actors	songs
Mb	puchayakaya

Deleted the table movies from the Main region N.Virginia :



But it is still available in Recovery region:



Task 2: Creating a dynamo DB table with global secondary indexes and fetching data using global secondary indexes.

1: table and the items

The screenshot shows the AWS DynamoDB console interface. On the left, there's a sidebar with a search bar and a list of tables: 'demotable' and 'movies'. The main panel displays the 'Items' tab for the 'demotable'. It shows a scan of the table with the following items:

username	orderid	returndate	useramount
Captain	8877	2592017	55556
Ironman	6699	23072018	5425
groot	3322	15071999	66666
hulk	7744	22182019	333.2

The bottom of the screen shows the Windows taskbar with the time 9:42 PM on 11/18/2020.

2: creating an global secondary index:

The screenshot shows the AWS DynamoDB console interface. On the left, there's a sidebar with a search bar and a list of tables: 'demotable' and 'movies'. The main panel displays the 'Indexes' tab for the 'demotable'. It shows a list of indexes with the following details:

Name	Status	Type	Partition key	Sort key	Attributes	Read capacity	Write capacity
returndate-useramount-lr	Active	GSI	returndate (String)	useramount (String)	ALL	5	5

The bottom of the screen shows the Windows taskbar with the time 9:41 PM on 11/18/2020.

3: Scanning the items with GSI

The screenshot shows the AWS DynamoDB console interface. On the left, a sidebar lists tables: 'demotable' and 'movies'. The main panel displays the 'Items' tab for 'demotable'. A filter is applied to the 'returndate' attribute with the value '2592017'. The search results show one item with the following details:

username	orderid	returndate	useramount
Captain	8877	2592017	55556

Task3: Deploying a python application in elastic beanstalk

Environments list:

The screenshot shows the AWS Elastic Beanstalk console interface. The main panel displays the 'All environments' list. Two environments are listed:

Environment name	Health	Application name	Date created	Last modified	URL	Running versions
Myapplication-env	Ok	myapplication	2020-11-19 14:49:47 UTC+0530	2020-11-19 14:53:29 UTC+0530	Myapplication-env.eba-mks4x45s.us-east-1.elasticbeanstalk.com	Sample Application
Myapplication1-env	Severe	myapplication1	2020-11-19 15:09:46 UTC+0530	2020-11-19 15:15:10 UTC+0530	reddomsin.us-east-1.elasticbeanstalk.com	myapplication1 source

Applications list:

The screenshot shows the AWS Elastic Beanstalk console. The left sidebar has a menu with 'Elastic Beanstalk' selected, and sub-items for 'Environments' and 'Applications'. The 'Applications' section is active, showing a list of applications. The main content area is titled 'All applications' and contains a table with the following data:

Application name	Environments	Date created	Last modified	ARN
myapplication	Myapplication-env	2020-11-19 14:49:23 UTC+0530	2020-11-19 14:49:23 UTC+0530	arn:aws:elasticbeanstalk:us-east-1:055184232800:application/myapplication
myapplication1	Myapplication1-env	2020-11-19 15:09:37 UTC+0530	2020-11-19 15:09:37 UTC+0530	arn:aws:elasticbeanstalk:us-east-1:055184232800:application/myapplication1

The bottom of the screenshot shows a Windows taskbar with the time 3:16 PM on 11/19/2020.

Health check page:

The screenshot shows the AWS Elastic Beanstalk console for the 'Myapplication-env' environment. The left sidebar has a menu with 'Elastic Beanstalk' selected, and sub-items for 'Environments' and 'Applications'. The 'Environments' section is active, showing a list of environments. The main content area is titled 'Myapplication-env' and contains a health check section with a green checkmark and the text 'Ok'. Below this is a 'Recent events' section with a table of events.

Time	Type	Details

The bottom of the screenshot shows a Windows taskbar with the time 3:16 PM on 11/19/2020.

Web page launched using the elastic beanstalk environment:

