

Assignment Day- 12 | AWS Advance

Question1:

Task-1: Create a Dynamodb table with minimum two disaster recovery Zones and verify replication.

ss1:Disaster recovery regions with the table

The screenshot shows the AWS DynamoDB console interface. On the left, a sidebar lists the services and a search bar. The main content area displays the configuration for 'MyFirstTable'. The 'Global Tables' tab is selected, showing the 'IAM role' as 'AWSServiceRoleForDynamoDBReplication'. Below this, the 'Global Table regions' section shows the 'Global table version' as '2019.11.21'. A table lists the configured regions:

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

The bottom of the screenshot shows the Windows taskbar with the search bar and system tray icons.

ss2:Home region with all items displayed

The screenshot shows the AWS DynamoDB console interface. On the left, there's a sidebar with 'Create table' and 'Delete table' buttons, and a search bar for filtering by table name. The main area displays 'MyFirstTable' with tabs for Overview, Items, Metrics, Alarms, Capacity, Indexes, Global Tables, and Backups. The 'Items' tab is selected, showing a list of items. The table has two columns: 'Movies' and 'Heros'. The items are:

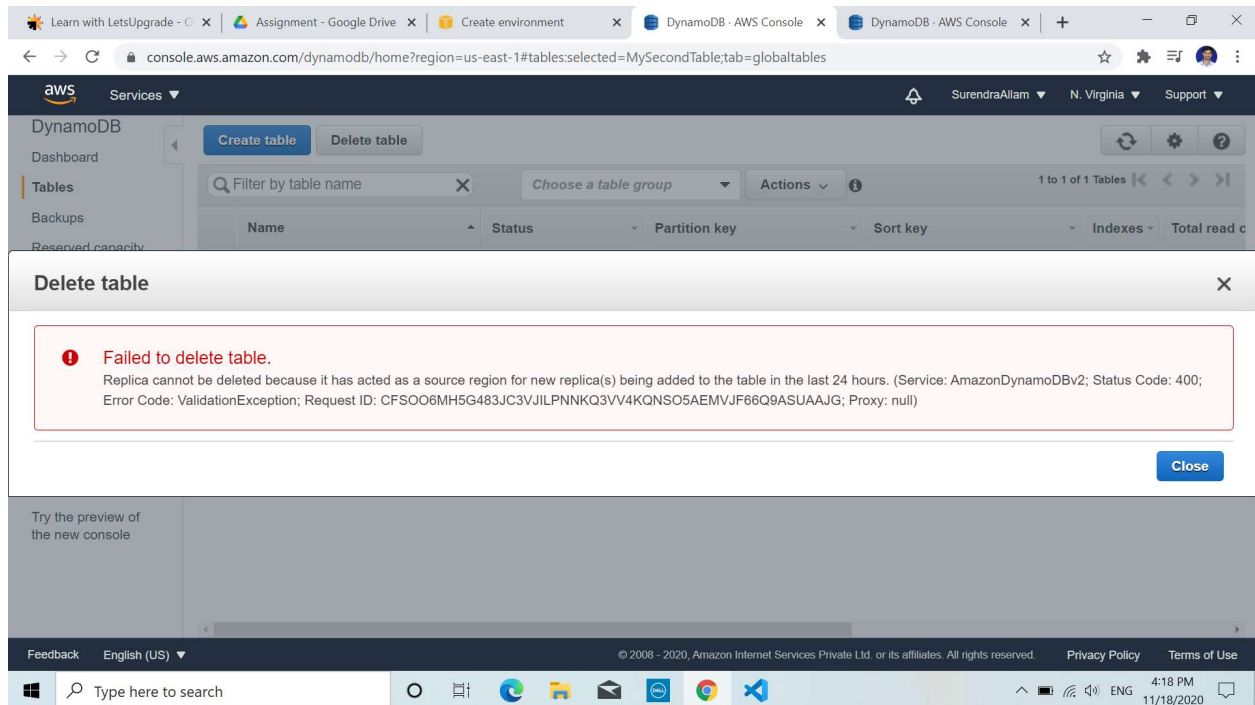
Movies	Heros
Bahubali	Prabhas
KGF	Yash
Mission Mangal	Akshay Kumar
RadheShyam	Prabhas

ss3:Use query to fetch few items

The screenshot shows the AWS DynamoDB console interface. On the left, there's a sidebar with 'Create table' and 'Delete table' buttons, and a search bar for filtering by table name. The main area displays 'MyFirstTable' with tabs for Overview, Items, Metrics, Alarms, Capacity, Indexes, Global Tables, and Backups. The 'Items' tab is selected, showing a list of items. The table has two columns: 'Movies' and 'Heros'. The items are:

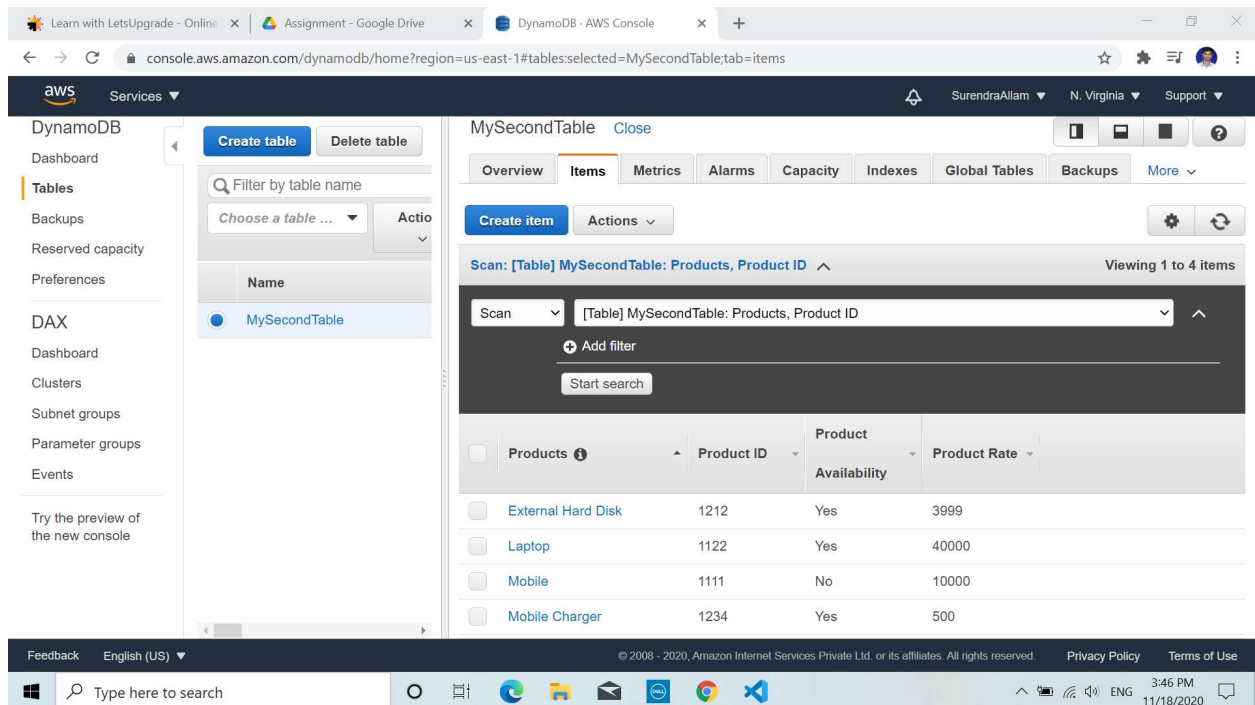
Movies	Heros
KGF	Yash

ss4:deletion and verification



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

ss1:Table with its items displayed



ss2:creating global secondary index

The screenshot shows the AWS DynamoDB console interface. On the left, the 'DynamoDB' sidebar is visible with options like 'Tables', 'Backups', 'Reserved capacity', 'Preferences', and 'DAX'. The main panel displays 'MySecondTable' with tabs for 'Overview', 'Items', 'Metrics', 'Alarms', 'Capacity', 'Indexes', 'Global Tables', 'Backups', and 'More'. The 'Indexes' tab is selected, showing a table with one index: 'Product-Rate-Product-Av' with status 'Active', type 'GSI', partition key 'Product Rate (String)', sort key 'Product Availability (\$)', attributes 'ALL', and read capacity '5'. The bottom of the screen shows a Windows taskbar with various application icons and the system clock indicating 3:45 PM on 11/18/2020.

ss3:scan with global secondary index

The screenshot shows the AWS DynamoDB console interface with the 'Items' tab selected for 'MySecondTable'. A scan operation is in progress, displaying a filter: 'Product Rate' is 'String' and equals '500'. The results show one item: 'Mobile Charger' with 'Product ID' 1234, 'Product Availability' 'Yes', and 'Product Rate' 500. The bottom of the screen shows a Windows taskbar with various application icons and the system clock indicating 3:44 PM on 11/18/2020.

Task 3: Deploying a python application in elastic beanstalk

ss1: Application page

The screenshot shows the AWS Elastic Beanstalk console's 'Applications' page. The left sidebar has 'Elastic Beanstalk' selected, with 'Applications' highlighted under the 'Environments' section. The main content area is titled 'All applications' and features a search bar, a refresh button, and a 'Create a new application' button. A table lists the applications:

Application name	Environments	Date created	Last modified	ARN
Python-web-application1	PythonWebApplication1-env	2020-11-18 16:15:27 UTC+0530	2020-11-18 16:15:27 UTC+0530	arn:aws:elasticbeanstalk:us-east-1:573310877256:application/Python-web-application1

The bottom of the image shows the Windows taskbar with the search bar and various application icons.

ss2: env list page

The screenshot shows the AWS Elastic Beanstalk console's 'Environments' page. The left sidebar has 'Elastic Beanstalk' selected, with 'Environments' highlighted under the 'Applications' section. The main content area is titled 'All environments' and features a search bar, a refresh button, and a 'Create a new environment' button. A table lists the environments:

Environment name	Health	Application name	Date created	Last modified	URL
PythonWebApplication-env (terminated)	-	Python-web-application	2020-11-18 15:52:22 UTC+0530	2020-11-18 16:12:26 UTC+0530	PythonWebApplication-env.eba-i66gqt68.us-east-1.elasticbeanstalk.com
PythonWebApplication1-env	Ok	Python-web-application1	2020-11-18 16:16:06 UTC+0530	2020-11-18 16:19:27 UTC+0530	PythonWebApplication-env.eba-35ipfzqb.us-east-1.elasticbeanstalk.com

The bottom of the image shows the Windows taskbar with the search bar and various application icons.

ss3:env health status page

The screenshot shows the AWS Elastic Beanstalk console. The left sidebar lists navigation options: Elastic Beanstalk, Environments, Applications, Python-web-application1, Application versions, Saved configurations, PythonWebApplication1-env, Go to environment, Configuration, Logs, Health, Monitoring, Alarms, and Managed updates. The main content area displays the details for the environment **PythonWebApplication1-env**. It includes a **Health** status of **Ok** with a green checkmark icon and a **Causes** button. The **Running version** section shows **Sample Application** and an **Upload and deploy** button. The **Platform** section shows the Python logo and text: **Python 3.7 running on 64bit Amazon Linux 2/3.1.3**, with a **Change** button. Below these sections is a **Recent events** section with a **Show all** button. The top of the console shows the AWS logo, **Services** dropdown, and user information: **SurendraAllam**, **N. Virginia**, and **Support**. The bottom of the console shows the Windows taskbar with the search bar and various application icons.

ss4:Web page launched using the elastic beanstalk env

The screenshot shows a web browser displaying a 'Congratulations' message. The main content area has a green background with the text **Congratulations** and a sub-header **What's Next?**. The text reads: **Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud**. Below this, it says: **This environment is launched with Elastic Beanstalk Python Platform**. The **What's Next?** section lists several links: [AWS Elastic Beanstalk overview](#), [AWS Elastic Beanstalk concepts](#), [Deploy a Django Application to AWS Elastic Beanstalk](#), [Deploy a Flask Application to AWS Elastic Beanstalk](#), [Customizing and Configuring a Python Container](#), and [Working with Logs](#). The top of the browser shows the address bar with the URL pythonwebapplication1-env.eba-35ipfzqb.us-east-1.elasticbeanstalk.com. The bottom of the browser shows the Windows taskbar with the search bar and various application icons.