Advance AWS

AWS Project- 2 (Day -12)

Student:

Kishore Shinde

Teacher:

Mrs. Vinolin Jeremiah

Course:

Advance AWS Cloud Computing with DevOps Fundamentals

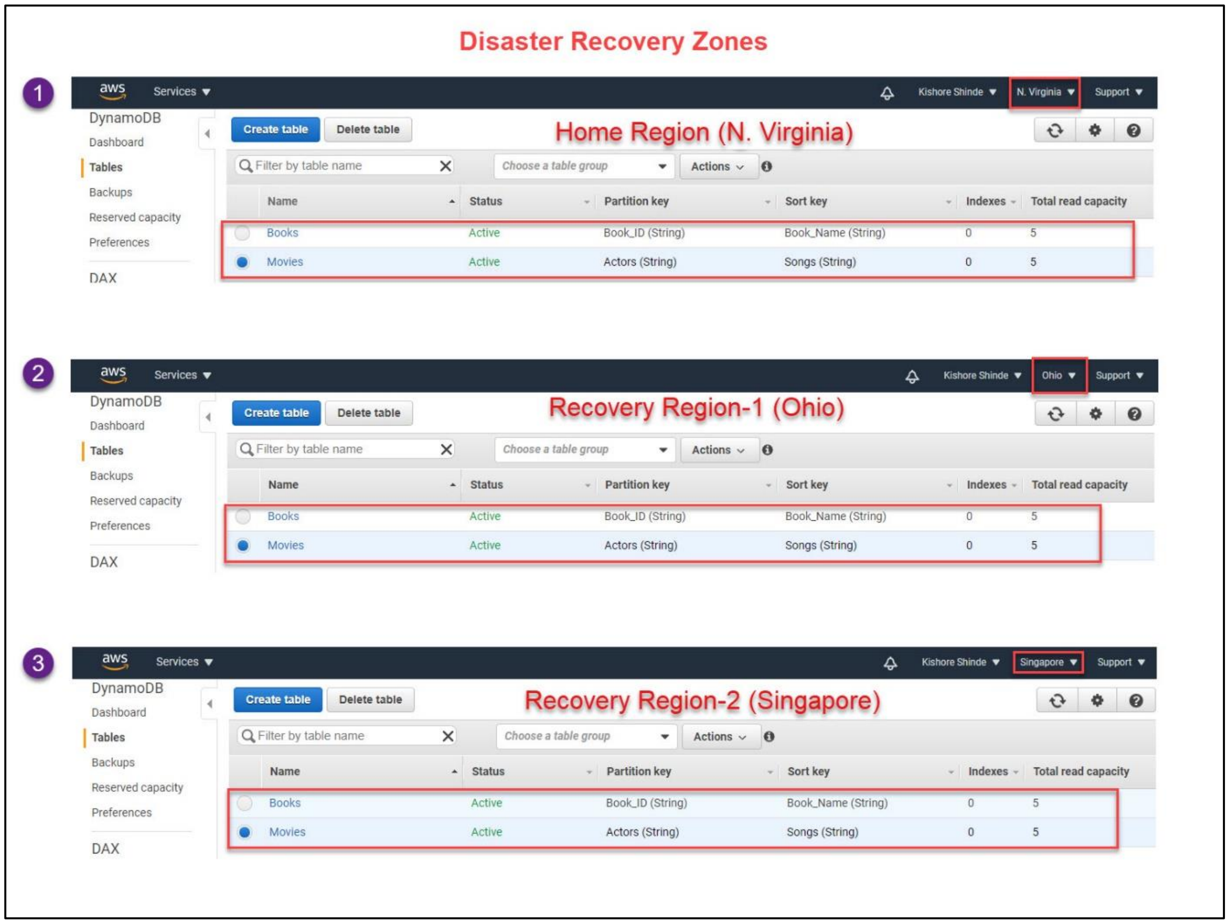
Institute:

Lets Upgrade

Project 2: Creating and Testing Dynamo DB table for Disaster recovery, fetching data using Global secondary indexes & deploying a phyton application in Elastic Beanstalk.

Task 1 : Create a Dynamo DB table with minimum two disaster recovery zones and verify replication.

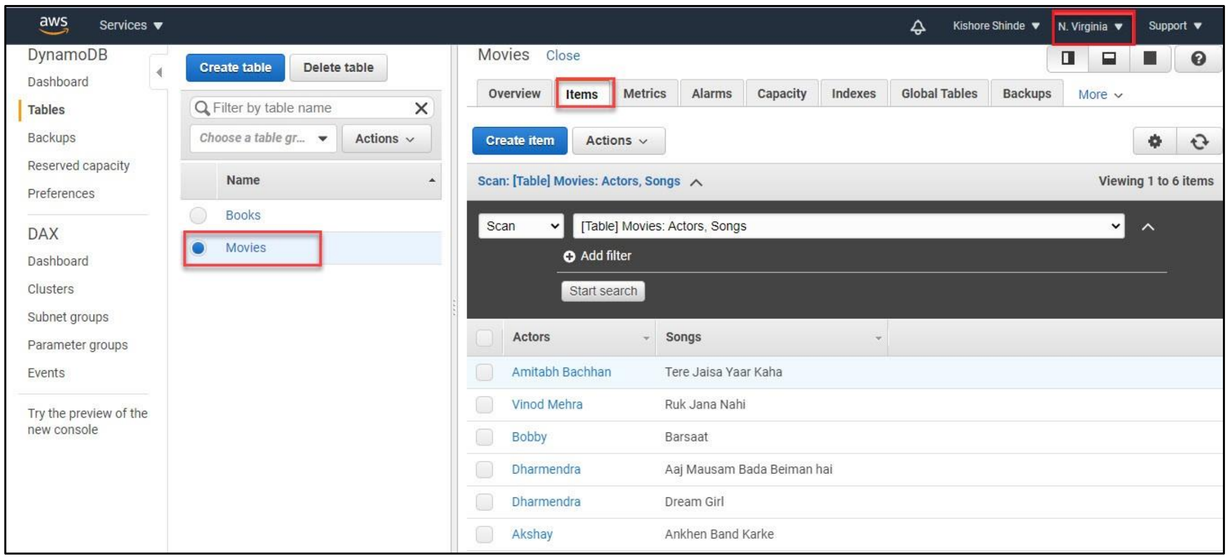
SS1 : Disaster recovery regions with the table


Disaster Recovery Zones
1
aws Services Kishore Shinde N. Virginia Support
DynamoDB Dashboard Tables Backups Reserved capacity Preferences DAX
Create table Delete table
Home Region (N. Virginia)
Filter by table name
Choose a table group
Actions
2
aws Services Kishore Shinde Ohio Support
Recovery Region-1 (Ohio)
3
aws Services Kishore Shinde Singapore Support
Recovery Region-2 (Singapore)

Name	Status	Partition key	Sort key	Indexes	Total read capacity
Books	Active	Book_ID (String)	Book_Name (String)	0	5
Movies	Active	Actors (String)	Songs (String)	0	5

S. No.	Name	Region	Table/s Replicated
1.	Home Region	N. Virginia	• Movies
2.	Recovery Region -1	Ohio	• Movies
3.	Recovery Region -2	Singapore	• Movies

SS2 : Home region with all items displayed



Movies

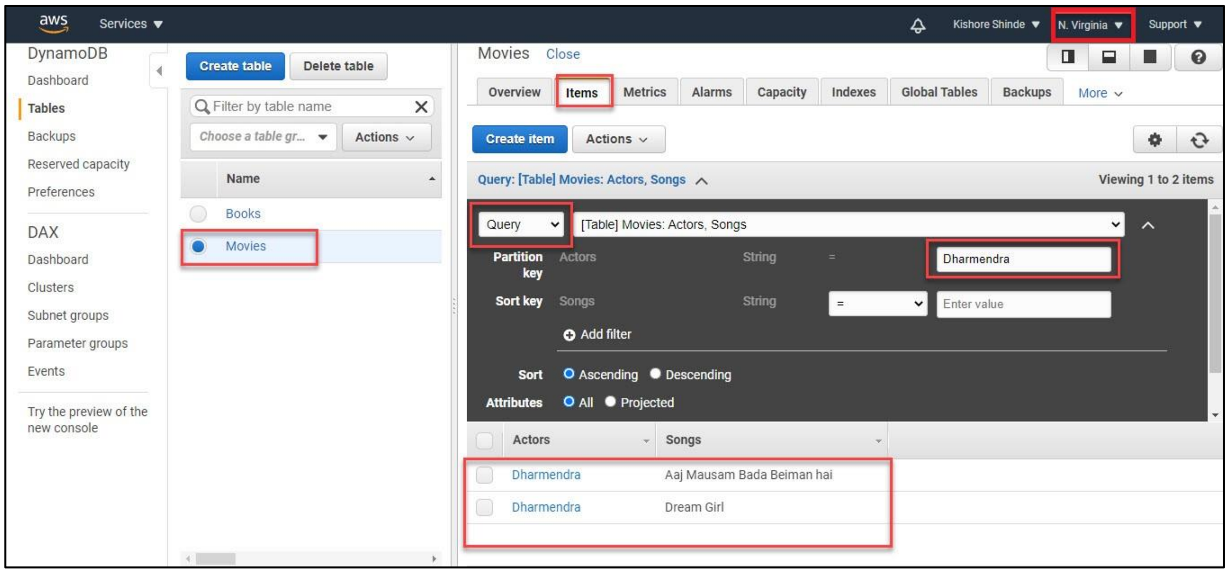
Scan: [Table] Movies: Actors, Songs — Viewing 1 to 6 items

Scan — [Table] Movies: Actors, Songs

Actors	Songs
Amitabh Bachhan	Tere Jaisa Yaar Kaha
Vinod Mehra	Ruk Jana Nahi
Bobby	Barsaat
Dharmendra	Aaj Mausam Bada Beiman hai
Dharmendra	Dream Girl
Akshay	Ankhen Band Karke

Sr. No.	Region	Table	Items
1	N. Virginia	Movies	• Actors • Songs

SS3 : Use query to fetch few items



Movies

Query: [Table] Movies: Actors, Songs — Viewing 1 to 2 items

Query — [Table] Movies: Actors, Songs

Partition key: Actors — String — = — Dharmendra

Sort key: Songs — String — =

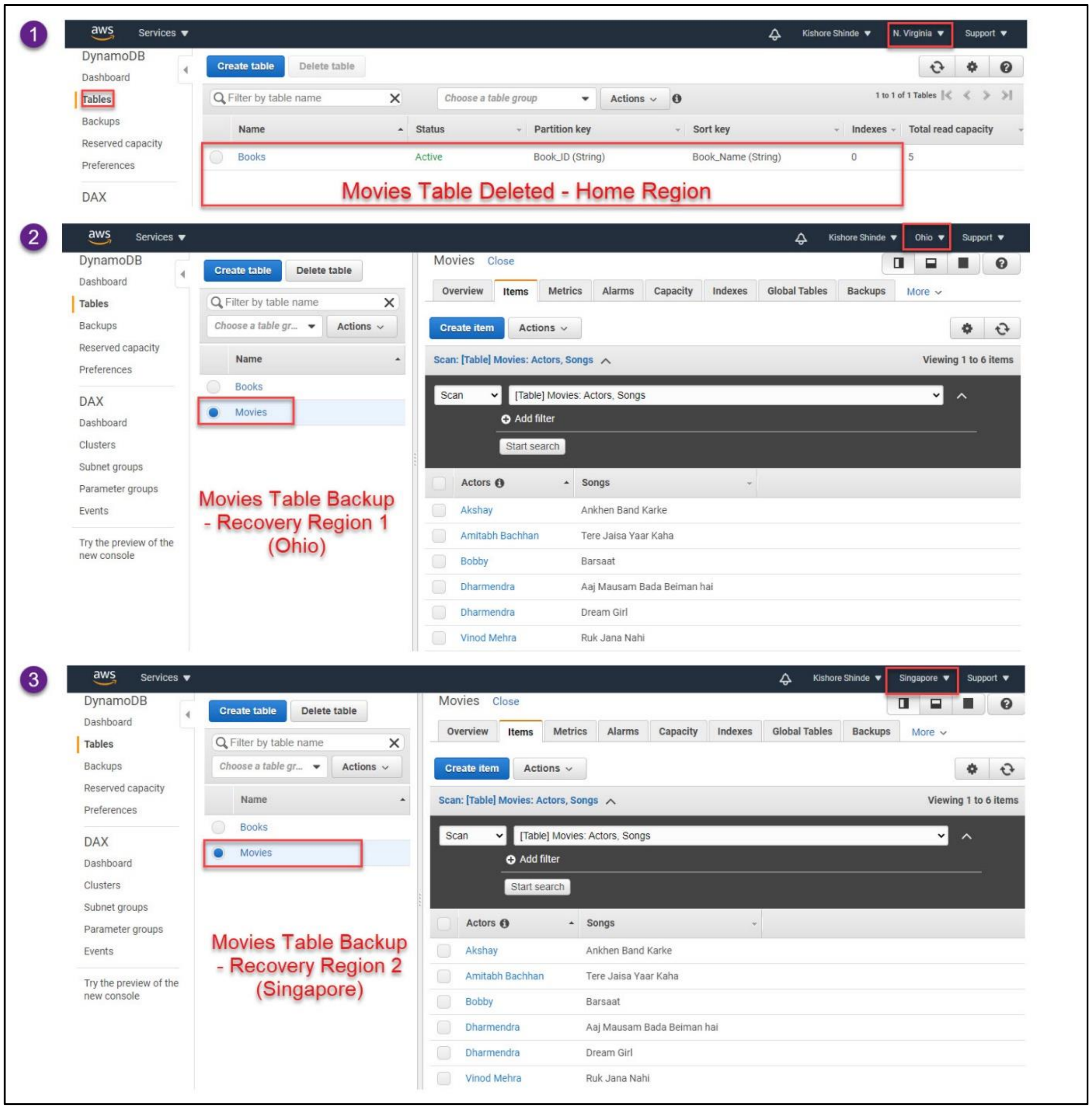
Sort: (● Ascending) Descending

Attributes: (● All) Projected

Actors	Songs
Dharmendra	Aaj Mausam Bada Beiman hai
Dharmendra	Dream Girl

Sr. No	Table Name	Query Field/Search String	Output
1	Movies	Field : Actors Search String **: "Dharmendra"**	Fetched **two** records for the specified search criteria.

SS4 : Deletion and Verification



	Name	Status	Partition key	Sort key	Indexes	Total read capacity
	Books	Active	Book_ID (String)	Book_Name (String)	0	5

Movies Table Deleted - Home Region

Movies Table Backup - Recovery Region 1 (Ohio)

Actors	Songs
Akshay	Ankhen Band Karke
Amitabh Bachhan	Tere Jaisa Yaar Kaha
Bobby	Barsaat
Dharmendra	Aaj Mausam Bada Beiman hai
Dharmendra	Dream Girl
Vinod Mehra	Ruk Jana Nahi

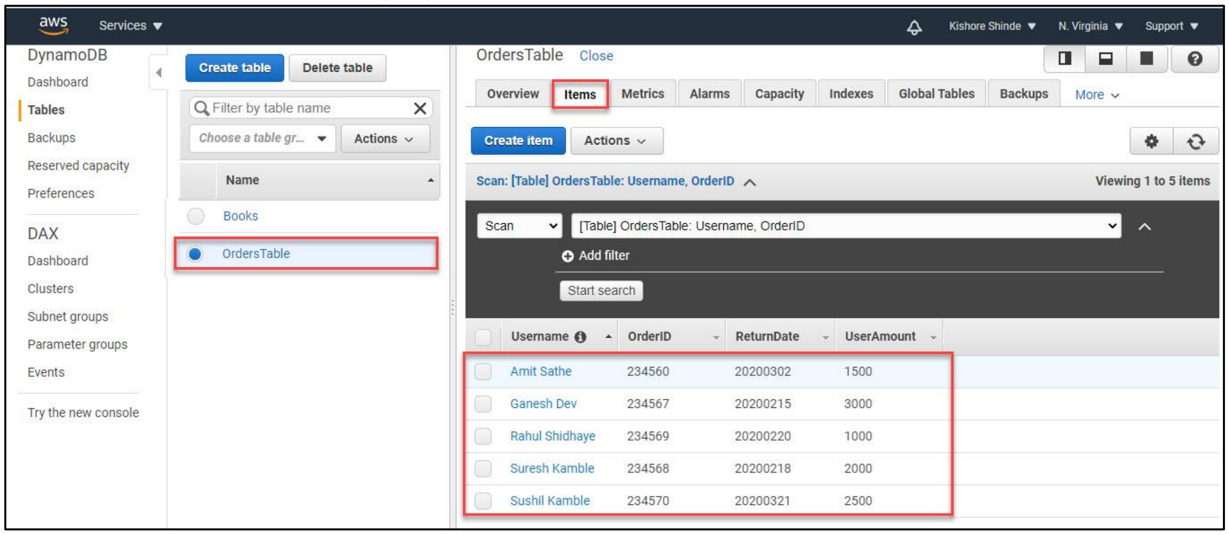
Movies Table Backup - Recovery Region 2 (Singapore)

Actors	Songs
Akshay	Ankhen Band Karke
Amitabh Bachhan	Tere Jaisa Yaar Kaha
Bobby	Barsaat
Dharmendra	Aaj Mausam Bada Beiman hai
Dharmendra	Dream Girl
Vinod Mehra	Ruk Jana Nahi

Sr. No.	Region	Region Details	Table	Status
1	N. Virginia	Home Region	Movies	Deleted
2	Ohio	Recovery Region -1	Movies	Present
3	Singapore	Recovery Region -2	Movies	Present

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

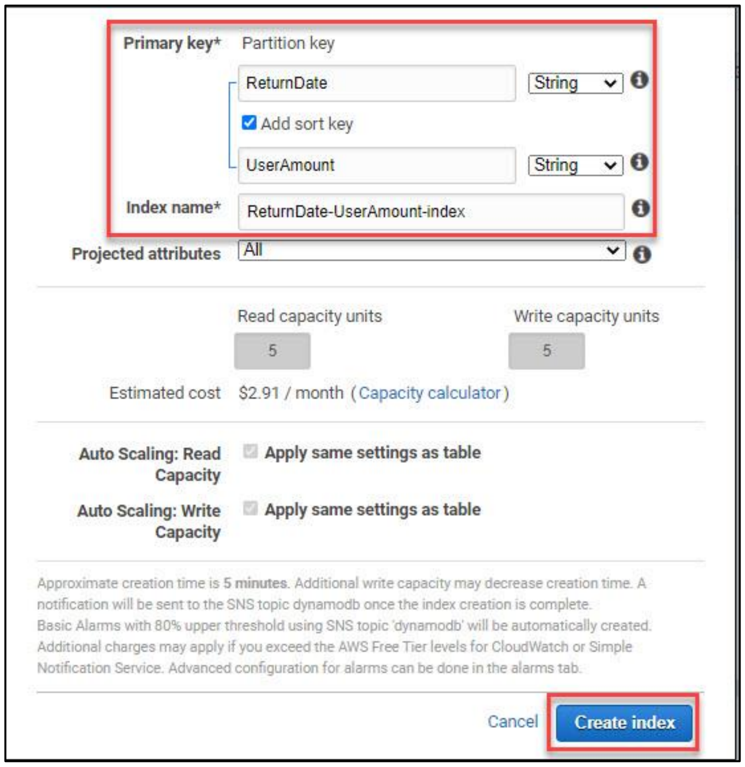
SS1 : Table with its item displayed



Username	OrderID	ReturnDate	UserAmount
Amit Sathe	234560	20200302	1500
Ganesh Dev	234567	20200215	3000
Rahul Shidhaye	234569	20200220	1000
Suresh Kamble	234568	20200218	2000
Sushil Kamble	234570	20200321	2500

Sr. No.	Table Name	Items
1.	OrdersTable	Username, OrderID, ReturnDate, UserAmount

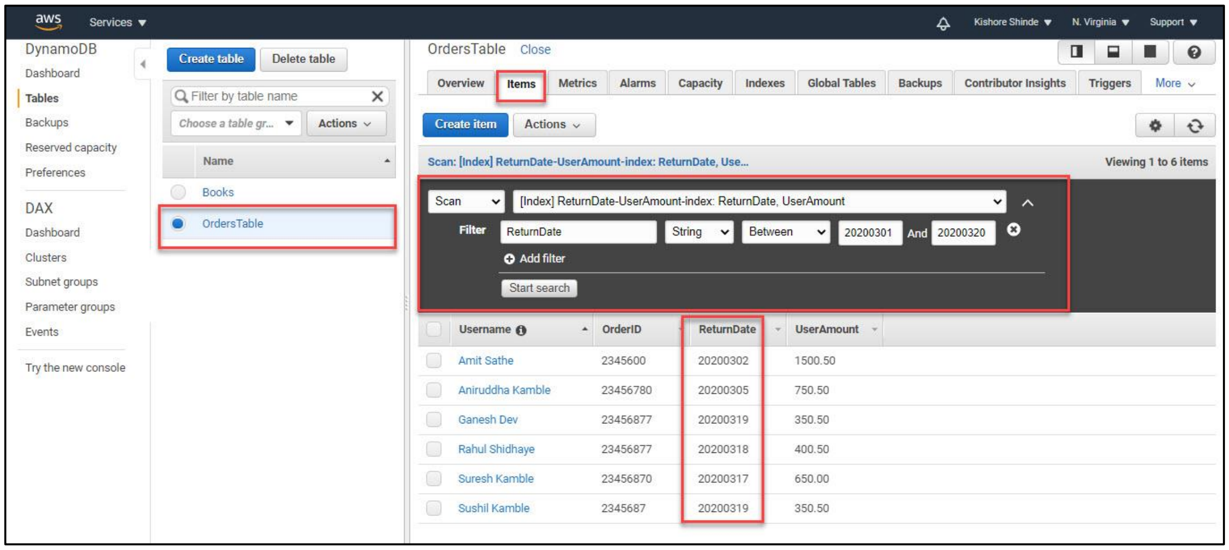
SS2 : Creating Secondary Global Index


Primary key* Partition key
ReturnDate String
[✓] Add sort key
UserAmount String
Index name* ReturnDate-UserAmount-index
Projected attributes All
Read capacity units 5
Write capacity units 5
Estimated cost $2.91 / month (Capacity calculator)
Auto Scaling: Read Capacity — Apply same settings as table
Auto Scaling: Write Capacity — Apply same settings as table

Approximate creation time is 5 minutes. Additional write capacity may decrease creation time. A notification will be sent to the SNS topic dynamodb once the index creation is complete.
Basic Alarms with 80% upper threshold using SNS topic 'dynamodb' will be automatically created. Additional charges may apply if you exceed the AWS Free Tier levels for CloudWatch or Simple Notification Service. Advanced configuration for alarms can be done in the alarms tab.

Sr. No.	Partition key	Sort key	Index name
1.	ReturnDate	UserAmount	ReturnDate-UserAmount-index

SS3 : Scan with Global Secondary Index



Scan: [Index] ReturnDate-UserAmount-index: ReturnDate, Use... Viewing 1 to 6 items

Scan [Index] ReturnDate-UserAmount-index: ReturnDate, UserAmount

Filter ReturnDate String Between 20200301 And 20200320

Add filter

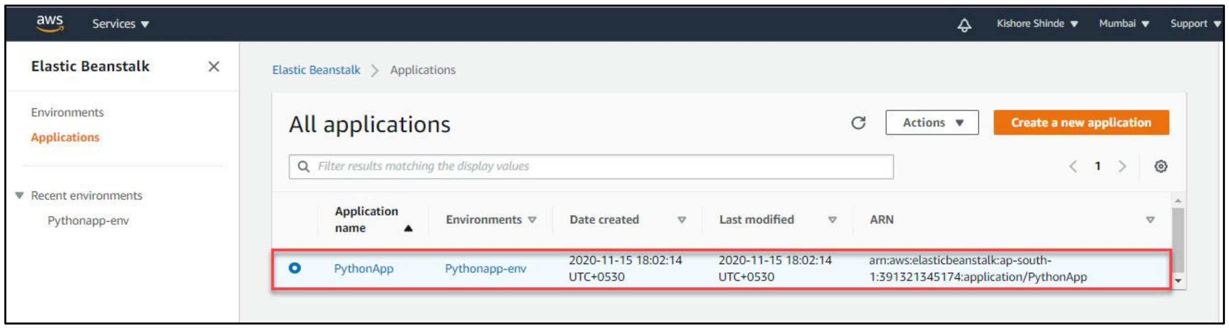
Start search

Username	OrderID	ReturnDate	UserAmount
Amit Sathe	2345600	20200302	1500.50
Aniruddha Kamble	23456780	20200305	750.50
Ganesh Dev	23456877	20200319	350.50
Rahul Shidhaye	23456877	20200318	400.50
Suresh Kamble	23456870	20200317	650.00
Sushil Kamble	2345687	20200319	350.50

Sr. No.	Table Name	Scan Index	Filter/Criteria	Output
1	OrdersTable	ReturnDate-UserAmount-index	ReturnDate Between '20200301' And '20200320'	Fetched **six** records for the specified search criteria.

Task 3 : Deploying a Python Application in Elastic Beanstalk

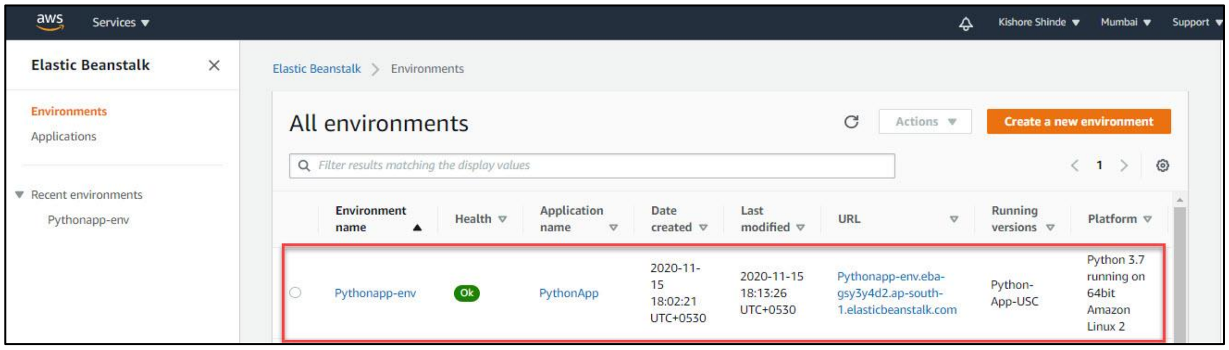
SS1 : Application Page



Application name	Environments	Date created	Last modified	ARN
PythonApp	Pythonapp-env	2020-11-15 18:02:14 UTC+0530	2020-11-15 18:02:14 UTC+0530	arn:aws:elasticbeanstalk:ap-south-1:391321345174:application/PythonApp

• Application Name	:	PythonApp
• Environments	:	Pythonapp-env

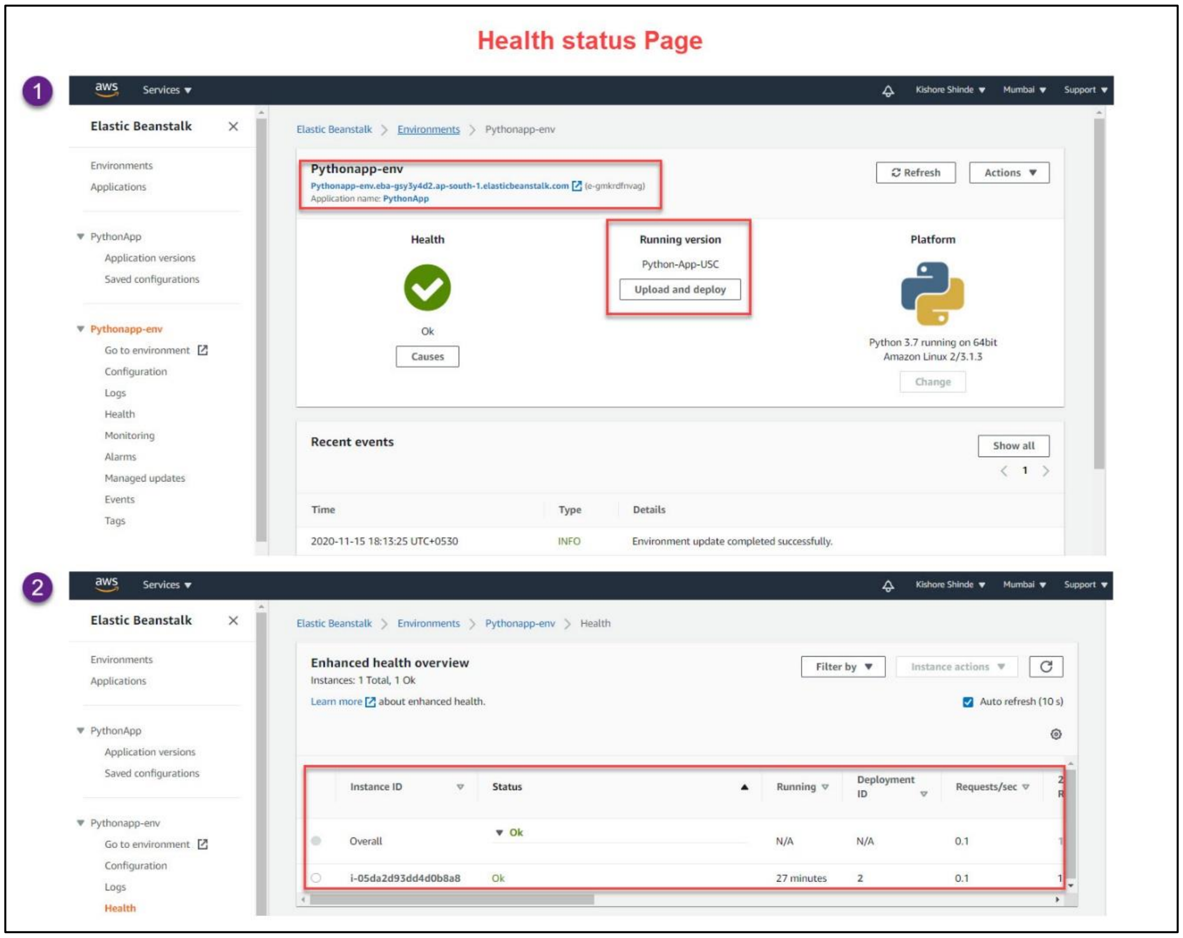
SS2 : Environment List Page



Environment name	Health	Application name	Date created	Last modified	URL	Running versions	Platform
Pythonapp-env	Ok	PythonApp	2020-11-15 18:02:21 UTC+0530	2020-11-15 18:13:26 UTC+0530	Pythonapp-env.eba-gsy3y4d2.ap-south-1.elasticbeanstalk.com	Python-App-USC	Python 3.7 running on 64bit Amazon Linux 2

- Environment name : Pythonapp-env
- Health : Ok
- Application name : PythonApp

SS3 : Environment Health Status Page



Pythonapp-env
Pythonapp-env.eba-gsy3y4d2.ap-south-1.elasticbeanstalk.com (e-gmkrdfnvag)
Application name: PythonApp

Time	Type	Details
2020-11-15 18:13:25 UTC+0530	INFO	Environment update completed successfully.

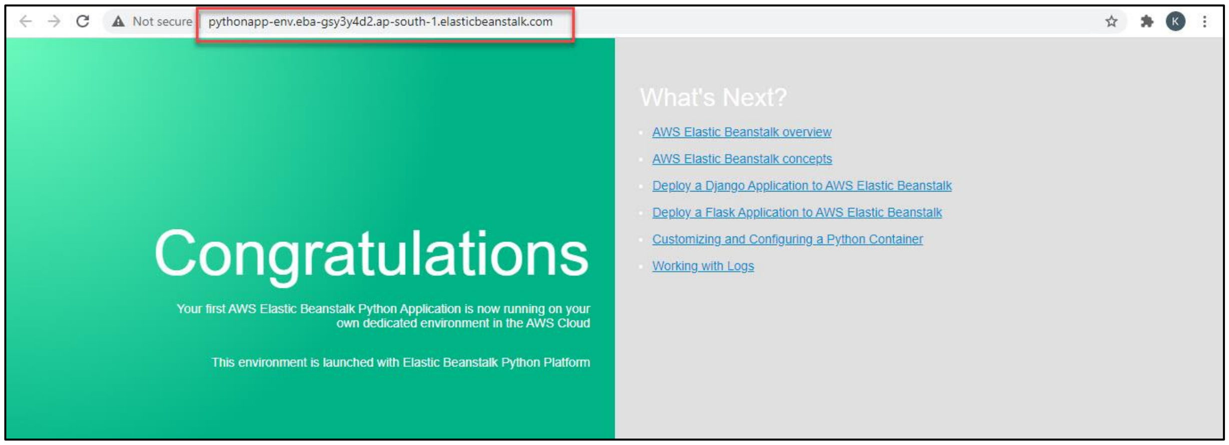
Enhanced health overview
Instances: 1 Total, 1 Ok
Learn more about enhanced health.

Instance ID	Status	Running	Deployment ID	Requests/sec
Overall	Ok	N/A	N/A	0.1
i-05da2d93dd4d0b8a8	Ok	27 minutes	2	0.1

Sr. No	Environment Name	Running Version	Platform	Health
1.	Pythonapp-env	Python-App-USC	Python 3.7	Ok

- DNS Name : Pythonapp-env-eba-gsy3y4d2.ap-south-1.elasticbeanstalk.com

SS4 : Web Page launched using Elastic Beanstalk


Not secure pythonapp-env.eba-gsy3y4d2.ap-south-1.elasticbeanstalk.com

Congratulations

Your first AWS Elastic Beanstalk Python Application is now running on your own dedicated environment in the AWS Cloud

This environment is launched with Elastic Beanstalk Python Platform

What's Next?

- 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
- Working with Logs

- DNS Name : Pythonapp-env-eba-gsy3y4d2.ap-south-1.elasticbeanstalk.com
- Message : The Environment is launched with **Elastic Beanstalk Python** Platform

xxx---Project 2 Ends Here--xxx