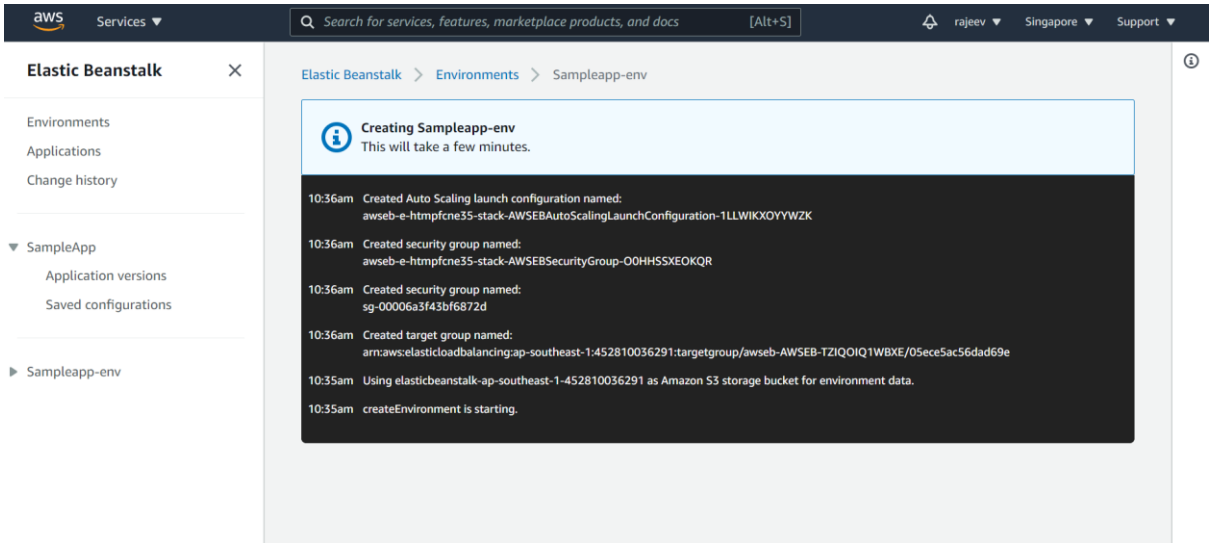
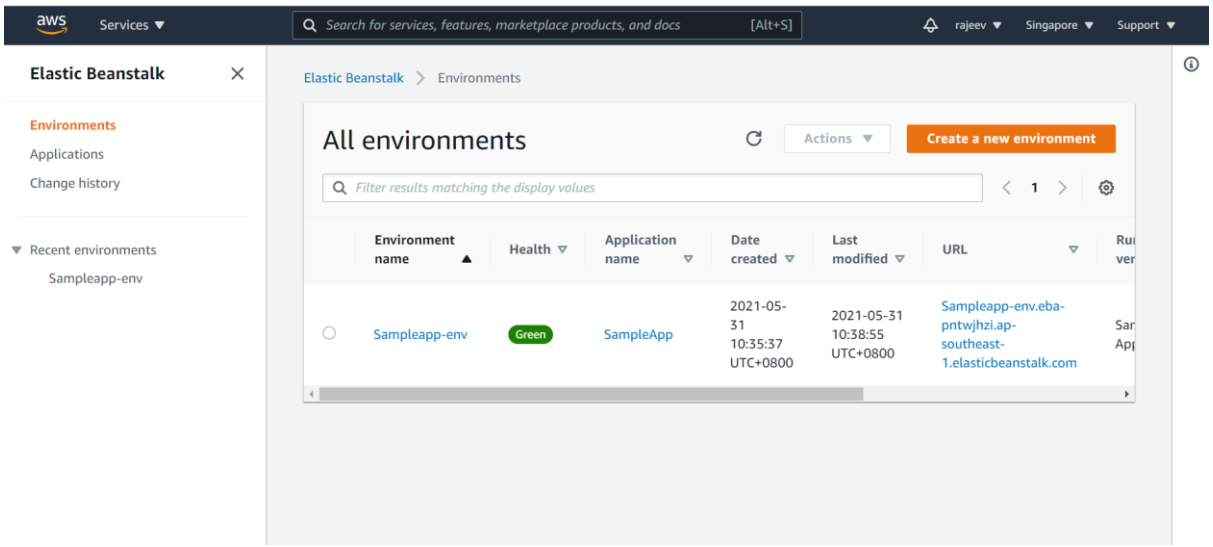
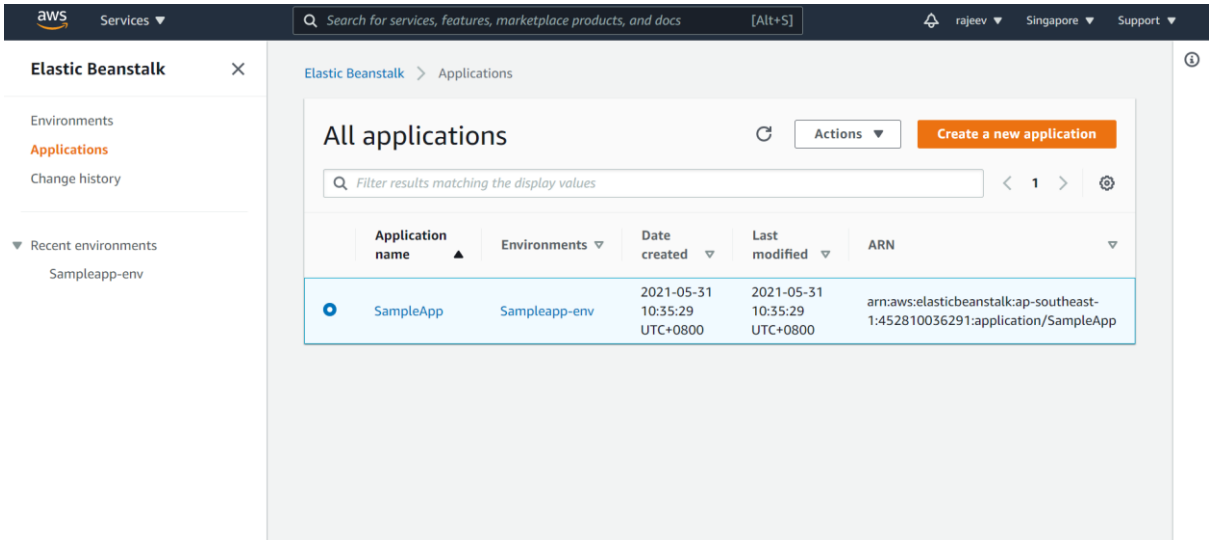
Assignment 5: ElasticBeanStalk

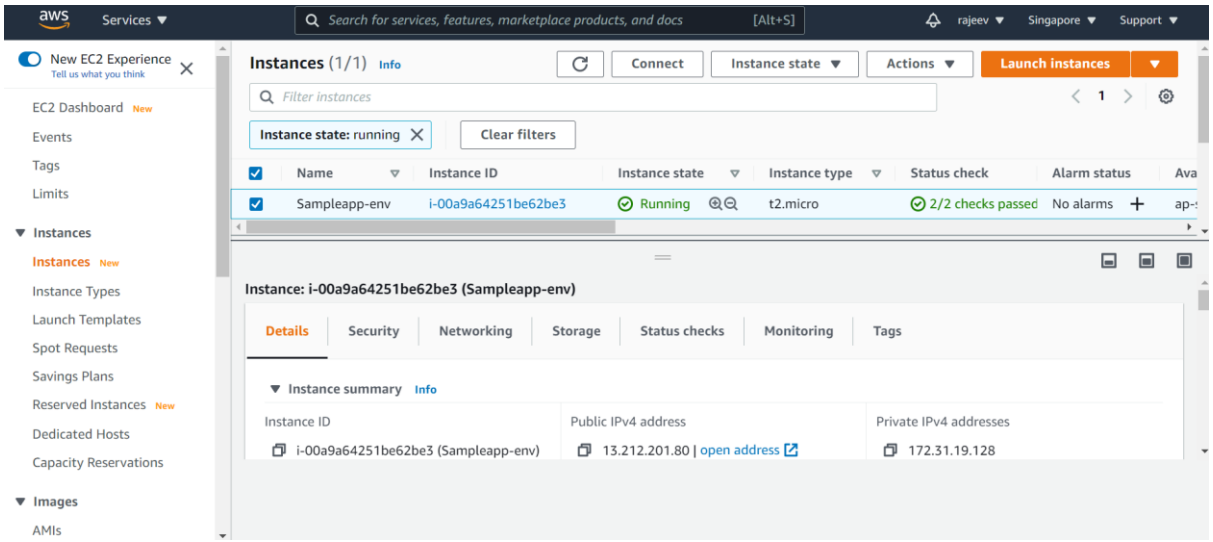
Create ElasticBeanStalk application using Java and edit the Auto Scaling desired capacity as 4 and Min instances as 4 Max as 6.



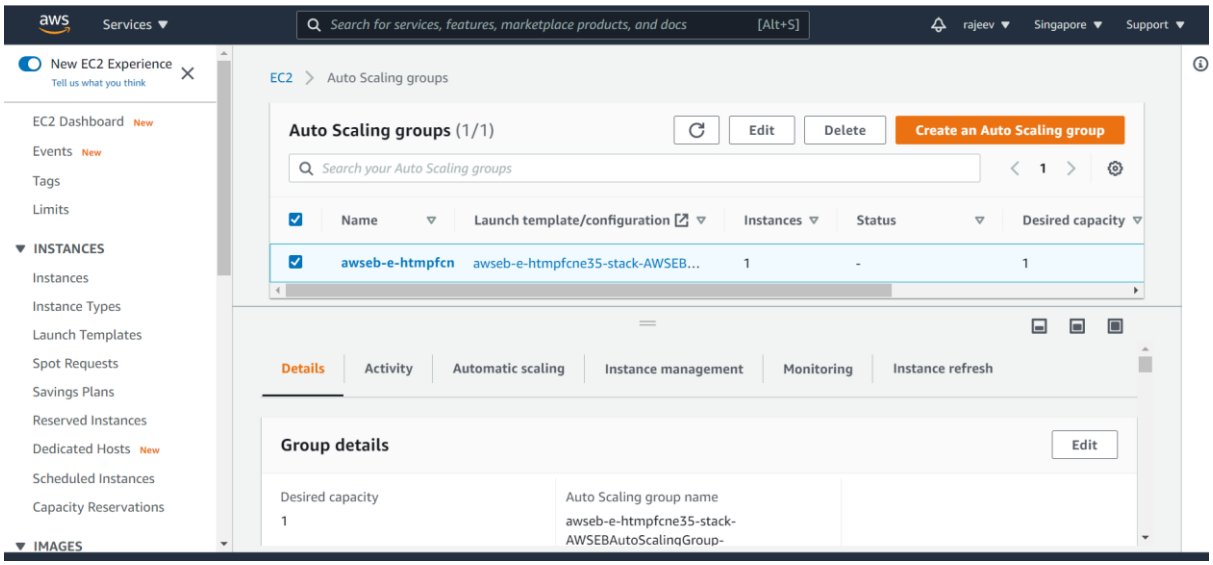


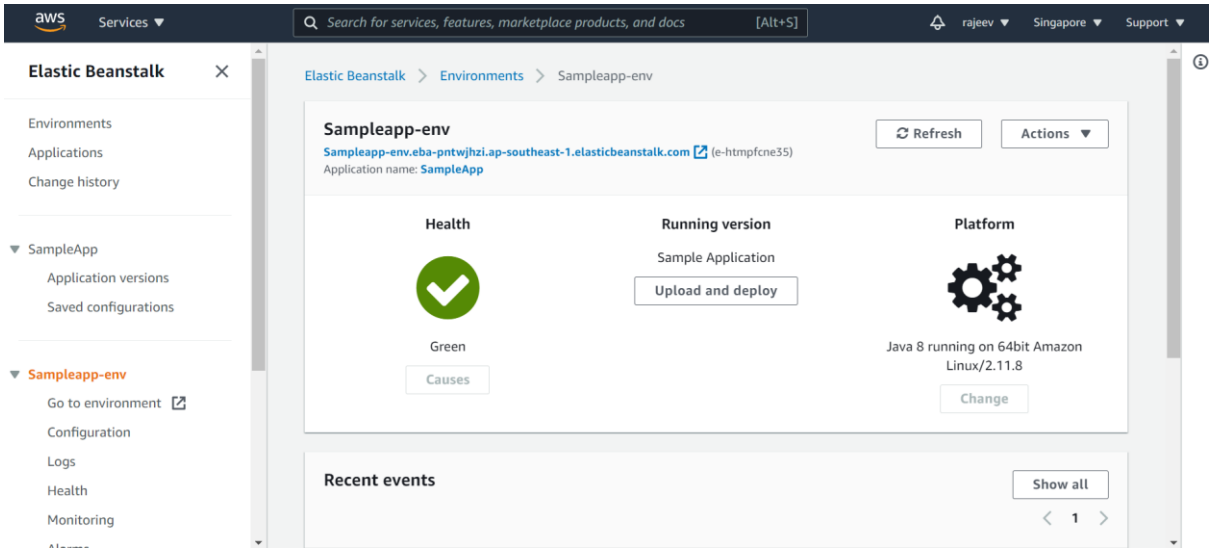


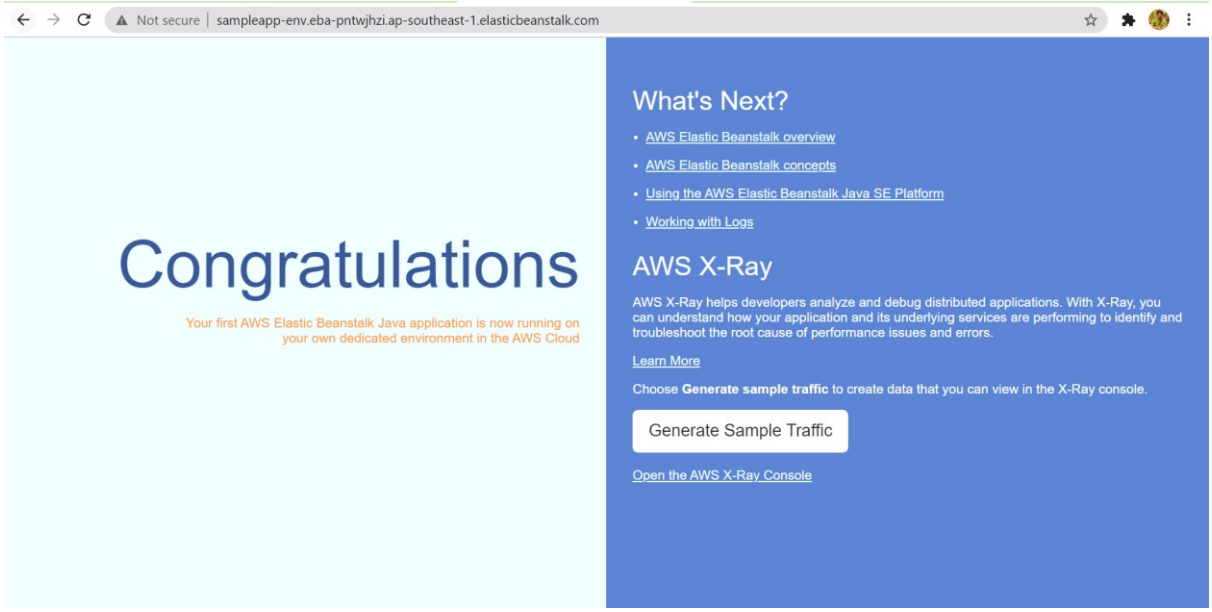
EC2 instance created



Autoscaling created




aws
Services
Search for services, features, marketplace products, and docs [Alt+S]
rajeev
Singapore
Support
Elastic Beanstalk
Environments
Applications
Change history
SampleApp
Application versions
Saved configurations
Sampleapp-env
Go to environment
Configuration
Logs
Health
Monitoring
Elastic Beanstalk > Environments > Sampleapp-env
Sampleapp-env
Sampleapp-env.eba-pntwjhzi.ap-southeast-1.elasticbeanstalk.com (e-htmpfcne35)
Application name: SampleApp
Refresh
Actions
Health
Green
Causes
Running version
Sample Application
Upload and deploy
Platform
Java 8 running on 64bit Amazon Linux/2.11.8
Change
Recent events
Show all
1


Not secure | sampleapp-env.eba-pntwjhzi.ap-southeast-1.elasticbeanstalk.com
Congratulations
Your first AWS Elastic Beanstalk Java application is now running on your own dedicated environment in the AWS Cloud
What's Next?
AWS Elastic Beanstalk overview
AWS Elastic Beanstalk concepts
Using the AWS Elastic Beanstalk Java SE Platform
Working with Logs
AWS X-Ray
AWS X-Ray helps developers analyze and debug distributed applications. With X-Ray, you can understand how your application and its underlying services are performing to identify and troubleshoot the root cause of performance issues and errors.
Learn More
Choose Generate sample traffic to create data that you can view in the X-Ray console.
Generate Sample Traffic
Open the AWS X-Ray Console

editing the Auto Scaling desired capacity as 4 and Min instances as 4 Max as 6.