CI/CD Deployment for Springboot Application.

Project 1

DESCRIPTION

Project Objective:

As a Full Stack Developer, you have to build a CI/CD pipeline to demonstrate continuous deployment and host the application on AWS EC2 instance.

Background of the problem statement:

As the project is in the final stage, management has asked you to automate the integration and deployment of the web application. You are required to set up an environment where the application will be hosted and accessed by users. The source code is supposed to be fetched from a GitHub repository.

You must use the following:

- Eclipse
- GitHub
- S3 bucket
- AWS EC2/ Virtual machine

This section will guide you to:

- Launch and connect to an EC2 instance

This lab has three subsections, namely:

1. Write spring boot program.

2. Launching an EC2 instance

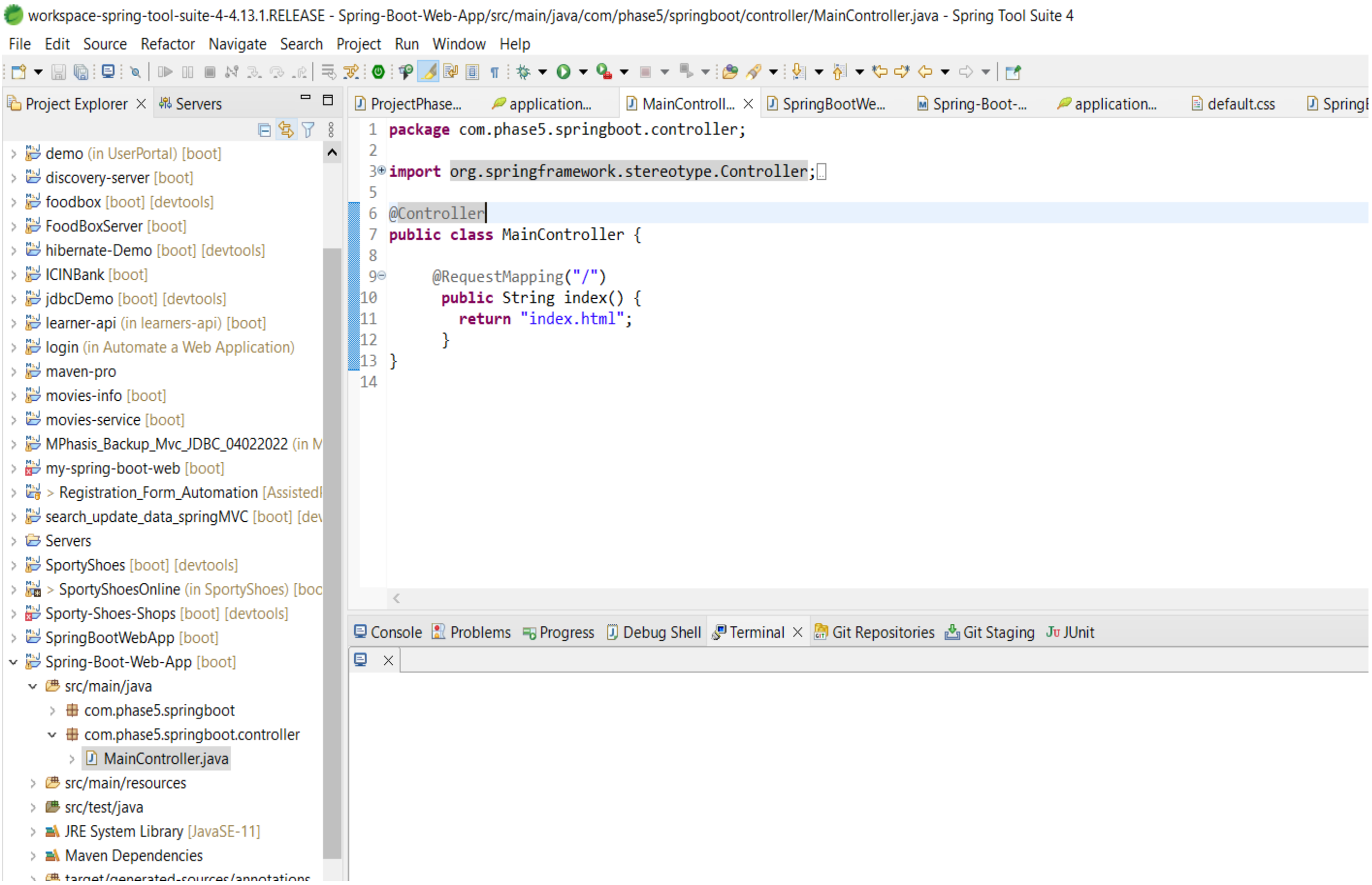
3. Connecting to the EC2 instance

4. Creating S3 Bucket.

5.Added jar file in bucket.

4. Pushing the files to GitHub repositories

Step 1: Write spring boot program:

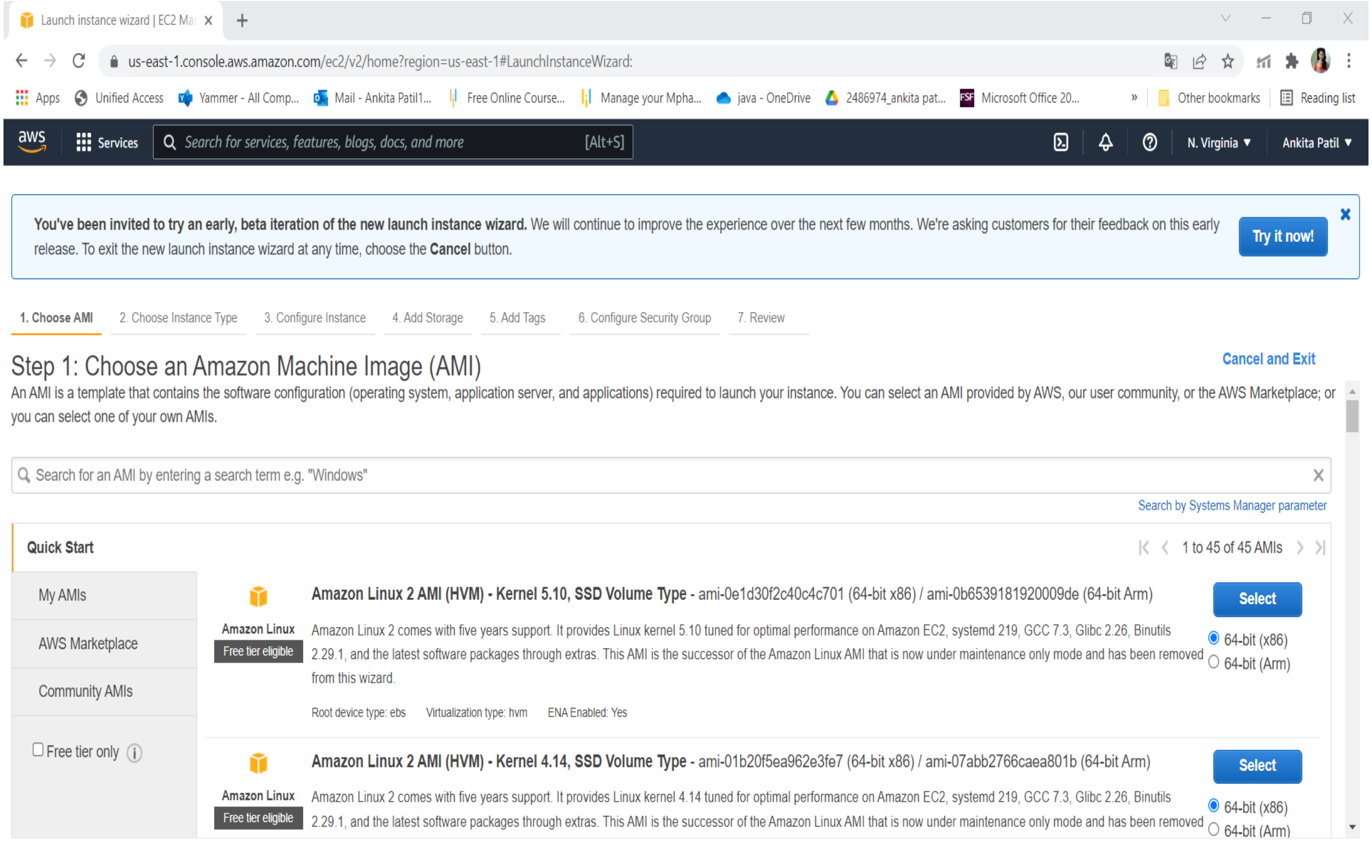


workspace-spring-tool-suite-4-4.13.1.RELEASE - Spring-Boot-Web-App/src/main/java/com/phase5/springboot/controller/MainController.java - Spring Tool Suite 4

```
package com.phase5.springboot.controller;

import org.springframework.stereotype.Controller;

@Controller
public class MainController {

     @RequestMapping("/")
      public String index() {
        return "index.html";
      }
}
```

File
Edit
Source
Refactor
Navigate
Search
Project
Run
Window
Help
Project Explorer
Servers
ProjectPhase...
application...
MainControll...
SpringBootWe...
Spring-Boot-...
application...
default.css
Spring
demo (in UserPortal) [boot]
discovery-server [boot]
foodbox [boot] [devtools]
FoodBoxServer [boot]
hibernate-Demo [boot] [devtools]
ICINBank [boot]
jdbcDemo [boot] [devtools]
learner-api (in learners-api) [boot]
login (in Automate a Web Application)
maven-pro
movies-info [boot]
movies-service [boot]
MPhasis_Backup_Mvc_JDBC_04022022 (in M
my-spring-boot-web [boot]
> Registration_Form_Automation [Assisted
search_update_data_springMVC [boot] [dev
Servers
SportyShoes [boot] [devtools]
> SportyShoesOnline (in SportyShoes) [boo
Sporty-Shoes-Shops [boot] [devtools]
SpringBootWebApp [boot]
Spring-Boot-Web-App [boot]
src/main/java
com.phase5.springboot
com.phase5.springboot.controller
MainController.java
src/main/resources
src/test/java
JRE System Library [JavaSE-11]
Maven Dependencies
target/generated-sources/annotations
Console
Problems
Progress
Debug Shell
Terminal
Git Repositories
Git Staging
JUnit

- **Step 2: Launching EC2 instance:**
  1. Click on launch instance to run any instance
  2. Select the AMI



You've been invited to try an early, beta iteration of the new launch instance wizard. We will continue to improve the experience over the next few months. We're asking customers for their feedback on this early release. To exit the new launch instance wizard at any time, choose the **Cancel** button.

Step 1: Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.

Search for an AMI by entering a search term e.g. "Windows"
Search by Systems Manager parameter
Quick Start
1 to 45 of 45 AMIs
My AMIs
AWS Marketplace
Community AMIs
□ Free tier only

Amazon Linux
Free tier eligible

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type - ami-0e1d30f2c40c4c701 (64-bit x86) / ami-0b6539181920009de (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 5.10 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is now under maintenance only mode and has been removed from this wizard.

Root device type: ebs   Virtualization type: hvm   ENA Enabled: Yes

Select
● 64-bit (x86)
○ 64-bit (Arm)

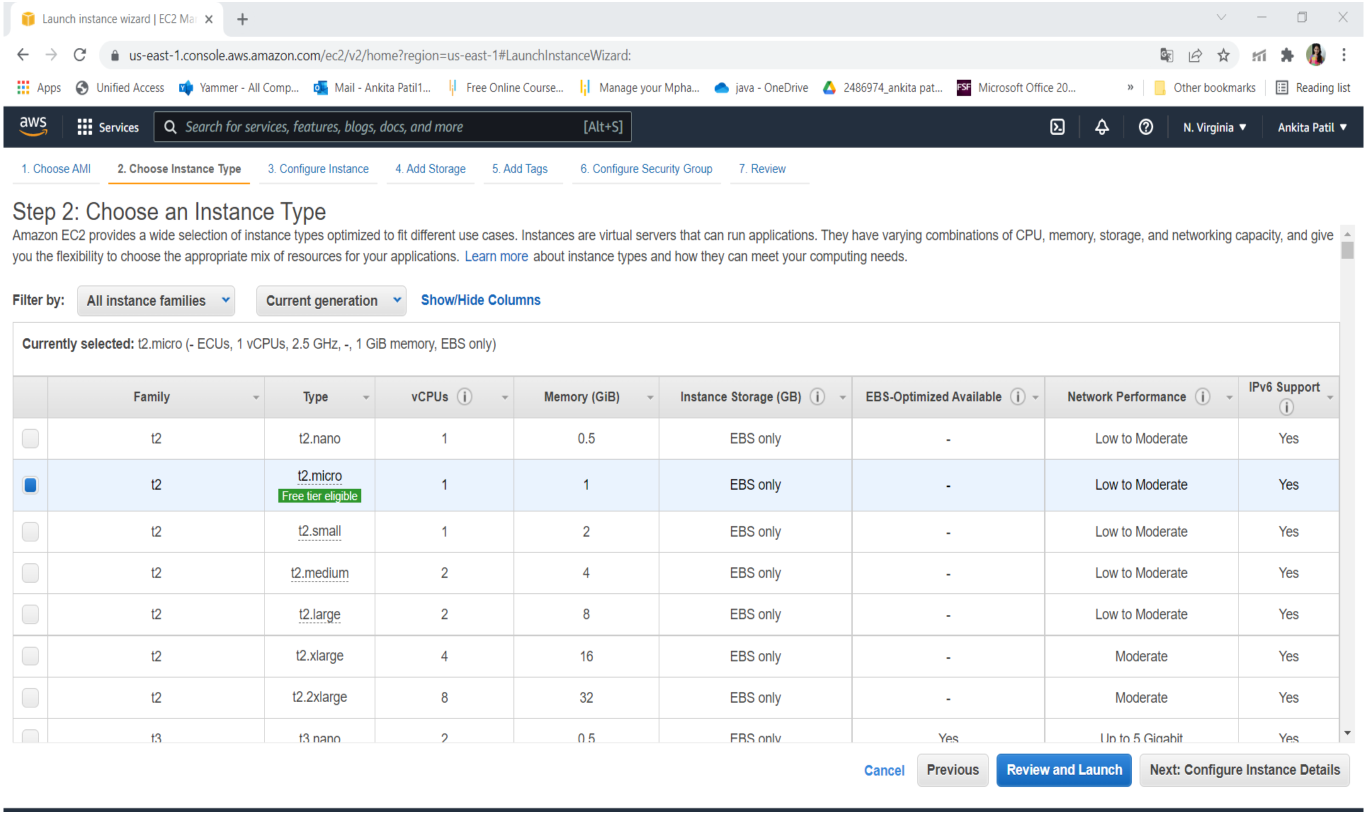
Amazon Linux
Free tier eligible

Amazon Linux 2 AMI (HVM) - Kernel 4.14, SSD Volume Type - ami-01b20f5ea962e3fe7 (64-bit x86) / ami-07abb2766caea801b (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is now under maintenance only mode and has been removed

Select
● 64-bit (x86)
○ 64-bit (Arm)

3. Select t2.micro as the instance type



Step 2: Choose an Instance Type

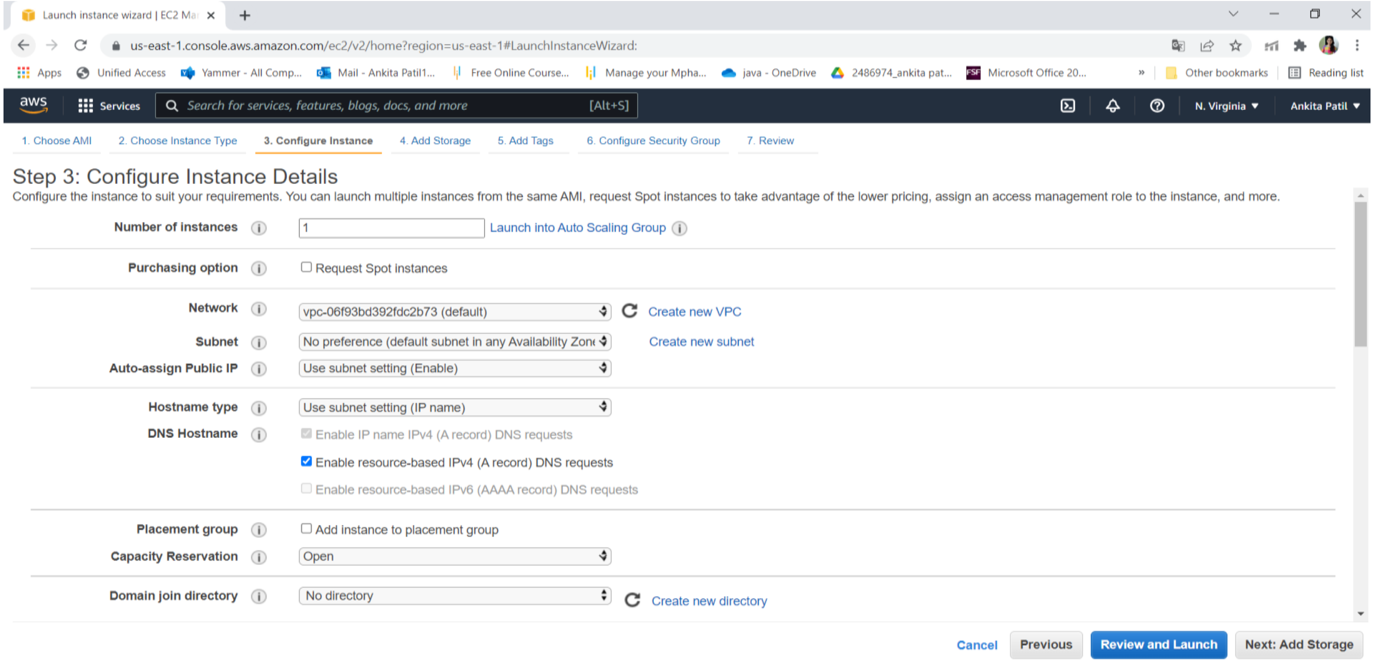
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. Learn more about instance types and how they can meet your computing needs.

Filter by: All instance families ▾   Current generation ▾   Show/Hide Columns

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
□	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
☑	t2	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
□	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
□	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
□	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
□	t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
□	t2	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
□	t3	t3.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes

4. Specify the number of instances, networks, placement groups, and IAM roles and click Next

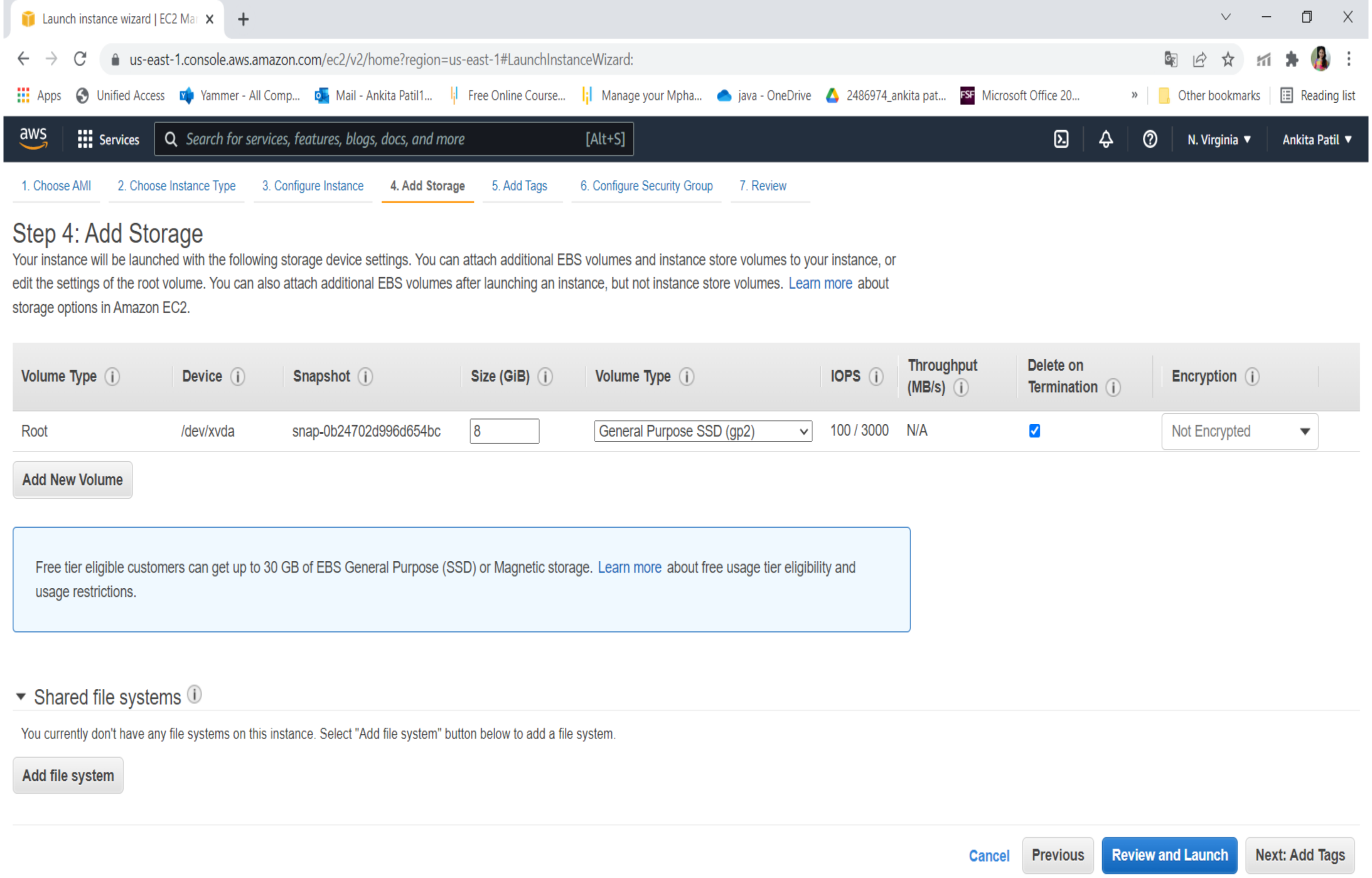


Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances	1 Launch into Auto Scaling Group
Purchasing option	☐ Request Spot instances
Network	vpc-06f93bd392fdc2b73 (default) Create new VPC
Subnet	No preference (default subnet in any Availability Zone Create new subnet
Auto-assign Public IP	Use subnet setting (Enable)
Hostname type	Use subnet setting (IP name)
DNS Hostname	☑ Enable IP name IPv4 (A record) DNS requests ☑ Enable resource-based IPv4 (A record) DNS requests ☐ Enable resource-based IPv6 (AAAA record) DNS requests
Placement group	☐ Add instance to placement group
Capacity Reservation	Open
Domain join directory	No directory Create new directory

5. Add storage



Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Learn more about storage options in Amazon EC2.

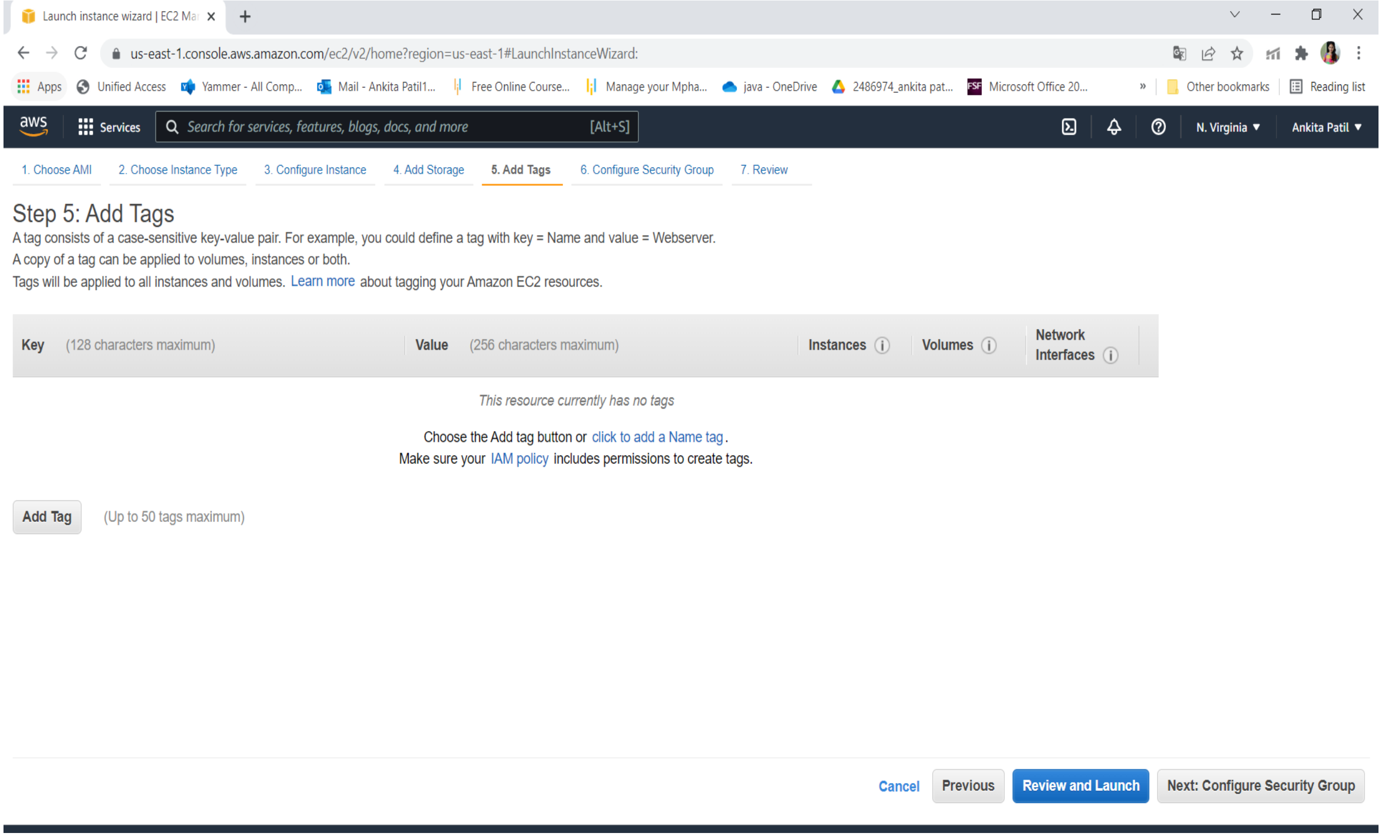
Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encryption
Root	/dev/xvda	snap-0b24702d996d654bc	8	General Purpose SSD (gp2)	100 / 3000	N/A	☑	Not Encrypted

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. Learn more about free usage tier eligibility and usage restrictions.

Shared file systems

You currently don't have any file systems on this instance. Select "Add file system" button below to add a file system.

6. You can add a key-value pair to the instance



Step 5: Add Tags

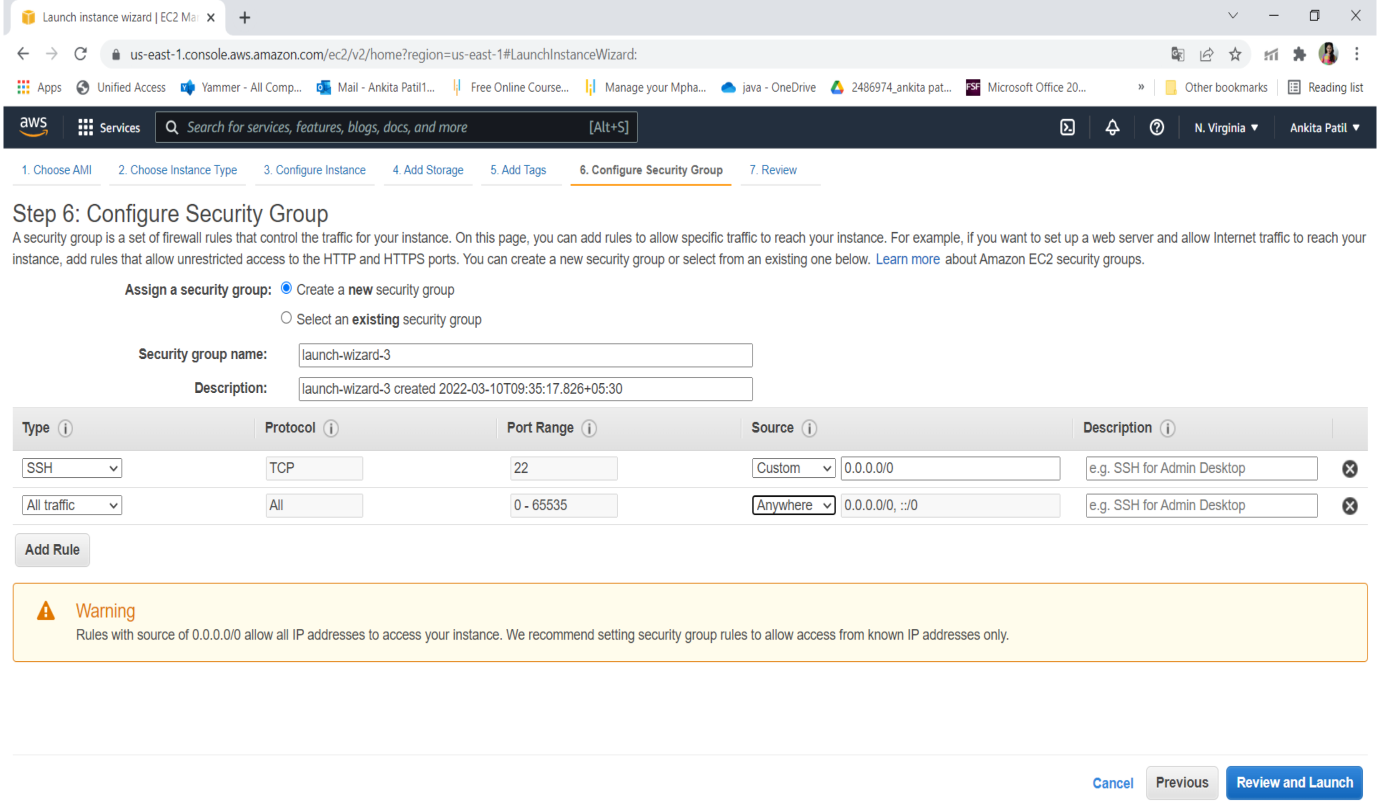
A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.

Key (128 characters maximum)	Value (256 characters maximum)	Instances	Volumes	Network Interfaces

This resource currently has no tags

Choose the Add tag button or click to add a Name tag.
Make sure your IAM policy includes permissions to create tags.

Add Tag (Up to 50 tags maximum)



Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. Learn more about Amazon EC2 security groups.

Assign a security group: ● Create a **new** security group
○ Select an **existing** security group

Security group name: launch-wizard-3

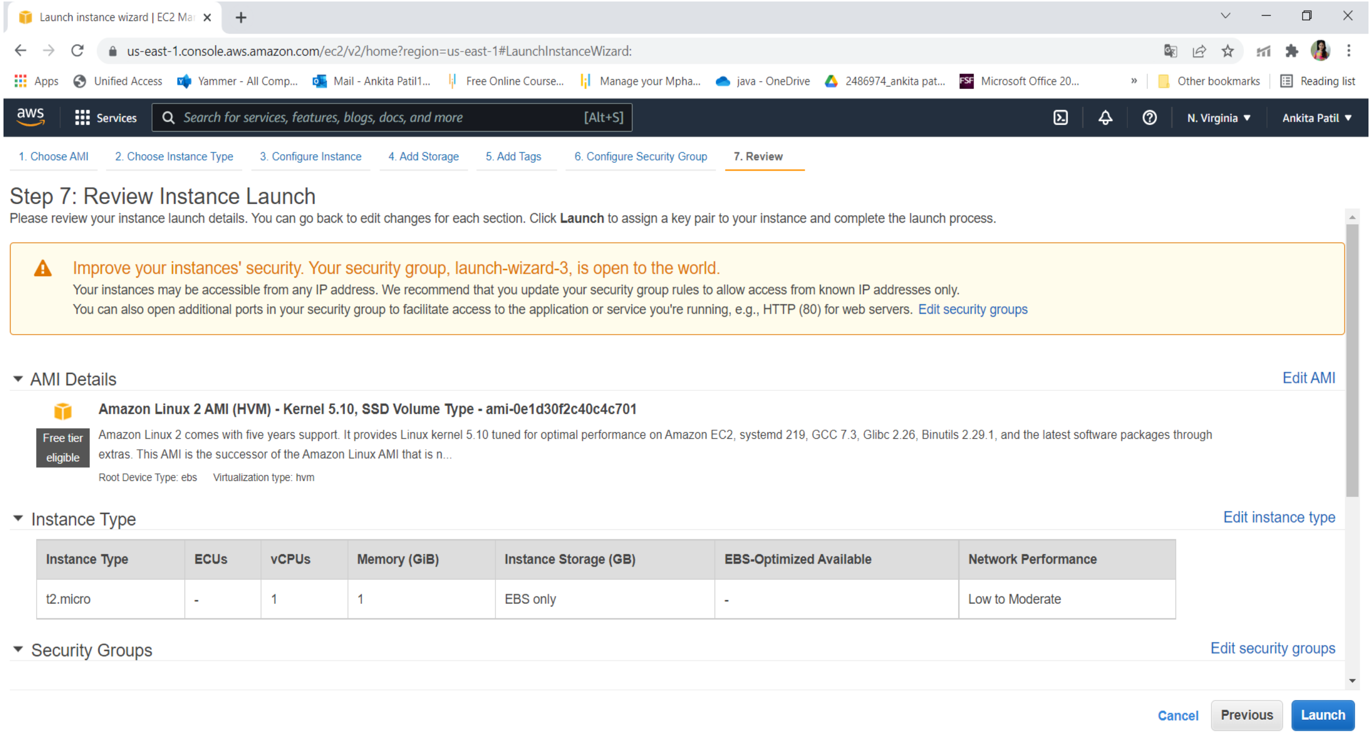
Description: launch-wizard-3 created 2022-03-10T09:35:17.826+05:30

Type	Protocol	Port Range	Source		Description
SSH	TCP	22	Custom	0.0.0.0/0	e.g. SSH for Admin Desktop
All traffic	All	0 - 65535	Anywhere	0.0.0.0/0, ::/0	e.g. SSH for Admin Desktop

Add Rule

Warning
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

7. Click on launch



Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

Improve your instances' security. Your security group, launch-wizard-3, is open to the world.

Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only.

You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. Edit security groups

▼ AMI Details

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type - ami-0e1d30f2c40c4c701

Free tier eligible

Amazon Linux 2 comes with five years support. It provides Linux kernel 5.10 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is n...

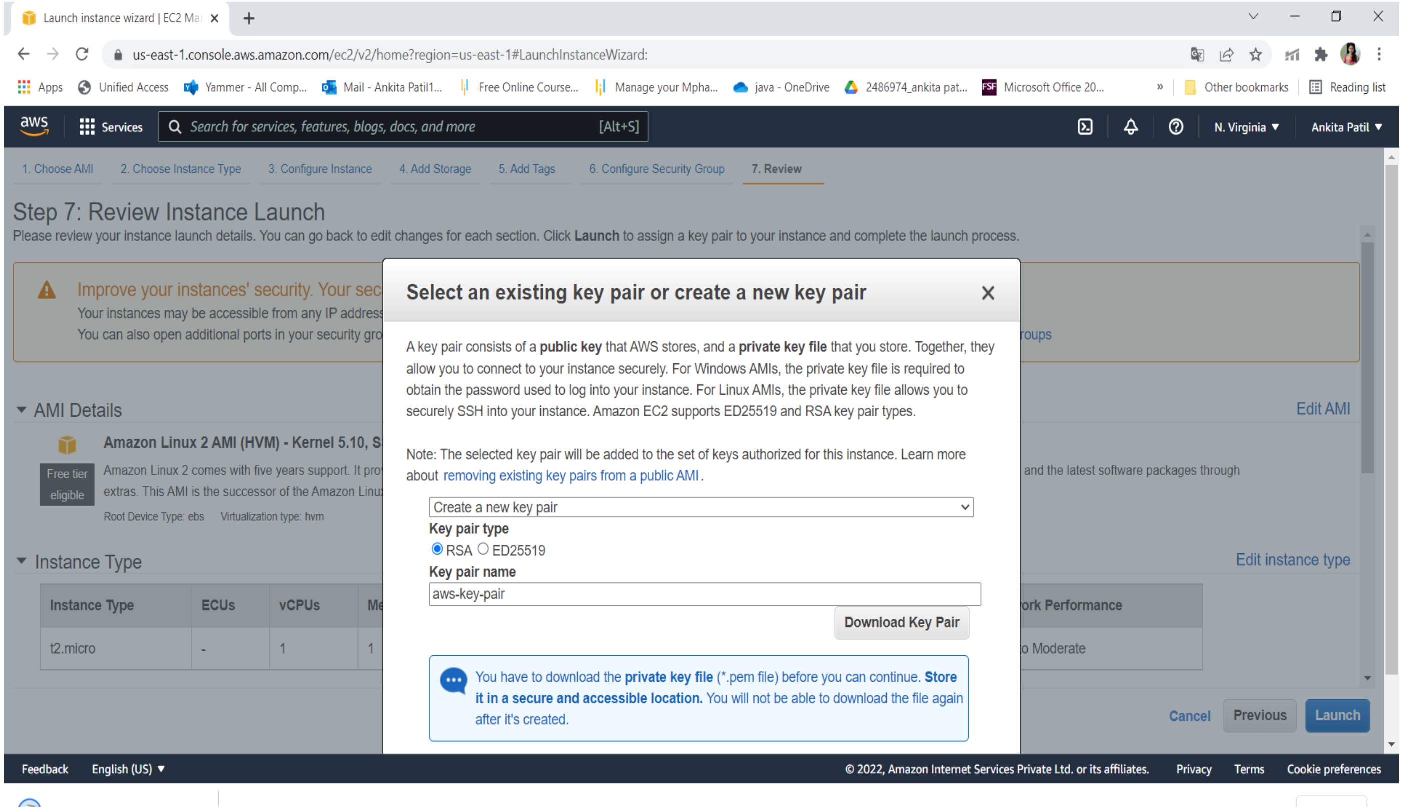
Root Device Type: ebs   Virtualization type: hvm

▼ Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

▼ Security Groups

8. Create key pair



Select an existing key pair or create a new key pair ✕

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance. Amazon EC2 supports ED25519 and RSA key pair types.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.

Create a new key pair

Key pair type

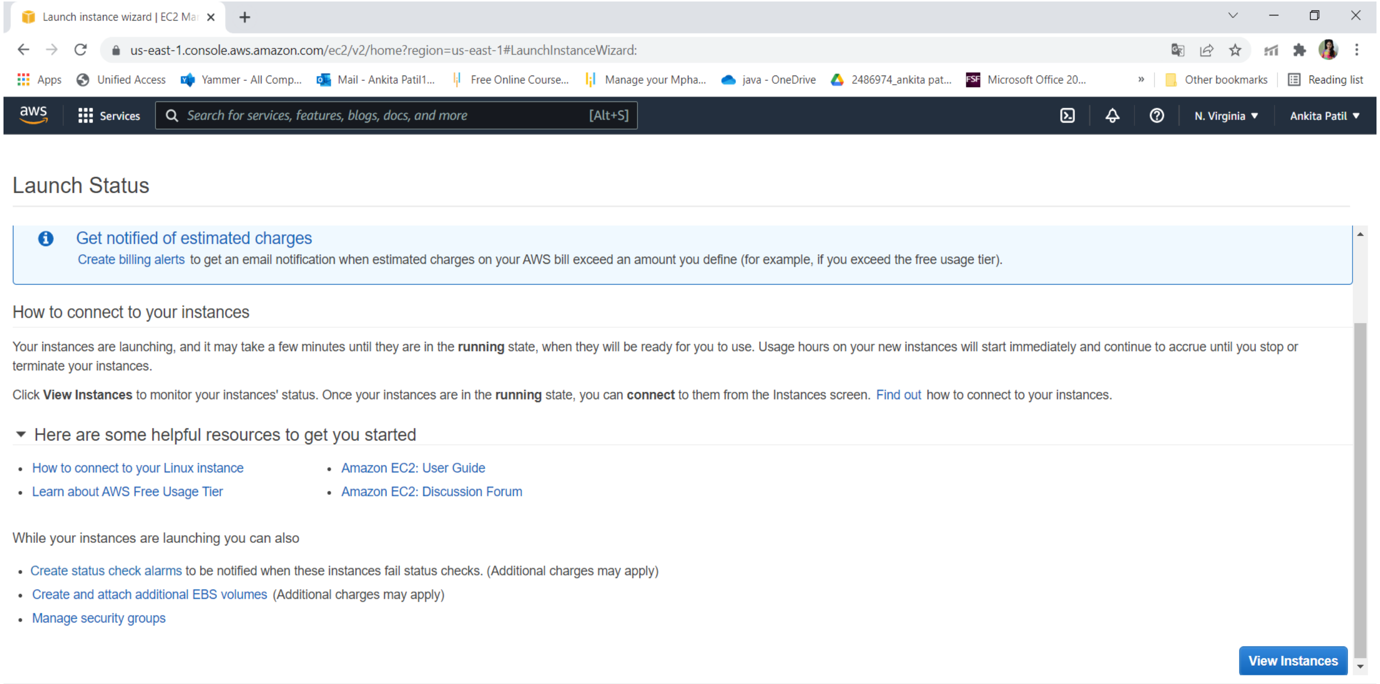
● RSA ○ ED25519

Key pair name

aws-key-pair

You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

9. Click on view instance



Launch Status

Get notified of estimated charges

Create billing alerts to get an email notification when estimated charges on your AWS bill exceed an amount you define (for example, if you exceed the free usage tier).

How to connect to your instances

Your instances are launching, and it may take a few minutes until they are in the **running** state, when they will be ready for you to use. Usage hours on your new instances will start immediately and continue to accrue until you stop or terminate your instances.

Click **View Instances** to monitor your instances' status. Once your instances are in the **running** state, you can **connect** to them from the Instances screen. Find out how to connect to your instances.

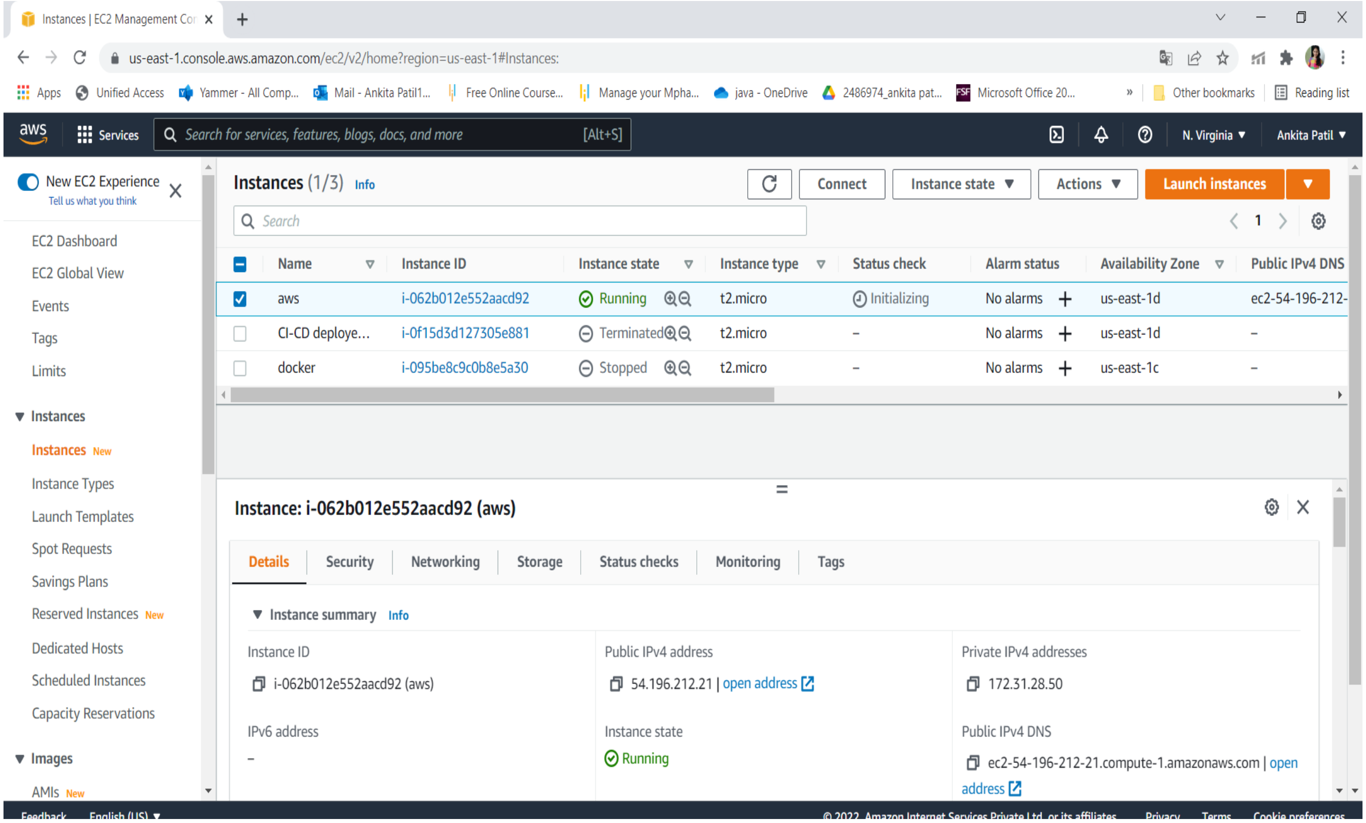
Here are some helpful resources to get you started

- How to connect to your Linux instance
- Learn about AWS Free Usage Tier
- Amazon EC2: User Guide
- Amazon EC2: Discussion Forum

While your instances are launching you can also

- Create status check alarms to be notified when these instances fail status checks. (Additional charges may apply)
- Create and attach additional EBS volumes (Additional charges may apply)
- Manage security groups

10.Connect the instance



Instances (1/3) Info

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
☑	aws	i-062b012e552aacd92	✓ Running	t2.micro	Initializing	No alarms +	us-east-1d	ec2-54-196-212-
□	CI-CD deploye...	i-0f15d3d127305e881	Terminated	t2.micro	–	No alarms +	us-east-1d	–
□	docker	i-095be8c9c0b8e5a30	Stopped	t2.micro	–	No alarms +	us-east-1c	–

Instance: i-062b012e552aacd92 (aws)

Instance summary Info

Instance ID
i-062b012e552aacd92 (aws)

Public IPv4 address
54.196.212.21 | open address

Private IPv4 addresses
172.31.28.50

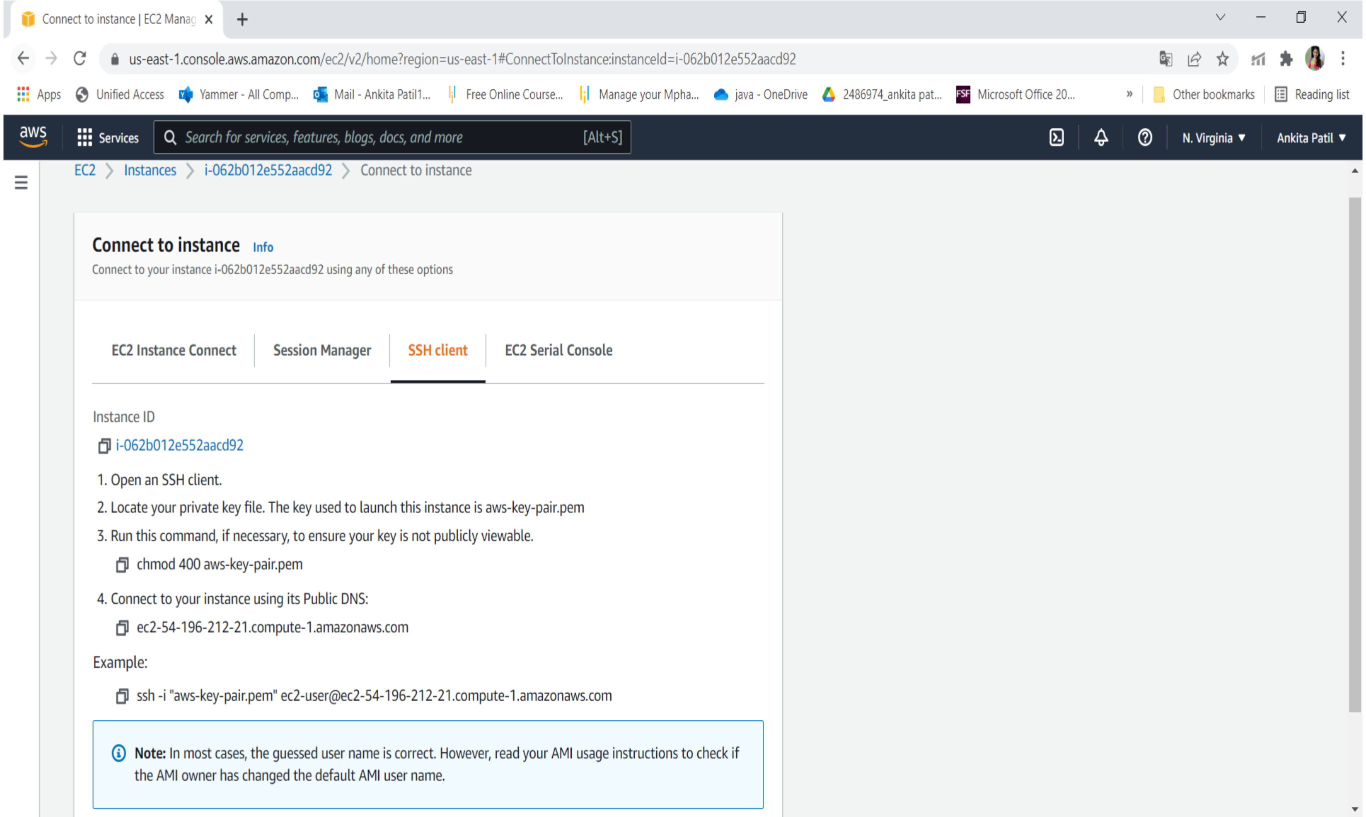
IPv6 address
–

Instance state
✓ Running

Public IPv4 DNS
ec2-54-196-212-21.compute-1.amazonaws.com | open address

Feedback
English (US)
© 2022, Amazon Internet Services Private Ltd. or its affiliates.
Privacy
Terms
Cookie preferences

- ## Step 3: Connect to EC2 instances:


Connect to instance | EC2 Manag
us-east-1.console.aws.amazon.com/ec2/v2/home?region=us-east-1#ConnectToInstance:instanceId=i-062b012e552aacd92
Apps
Unified Access
Yammer - All Comp...
Mail - Ankita Patil1...
Free Online Course...
Manage your Mpha...
java - OneDrive
2486974_ankita pat...
Microsoft Office 20...
Other bookmarks
Reading list
aws
Services
Search for services, features, blogs, docs, and more
[Alt+S]
N. Virginia ▼
Ankita Patil ▼
EC2 > Instances > i-062b012e552aacd92 > Connect to instance

Connect to instance Info

Connect to your instance i-062b012e552aacd92 using any of these options

Instance ID

i-062b012e552aacd92

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is aws-key-pair.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.

   ```
   chmod 400 aws-key-pair.pem
   ```

4. Connect to your instance using its Public DNS:

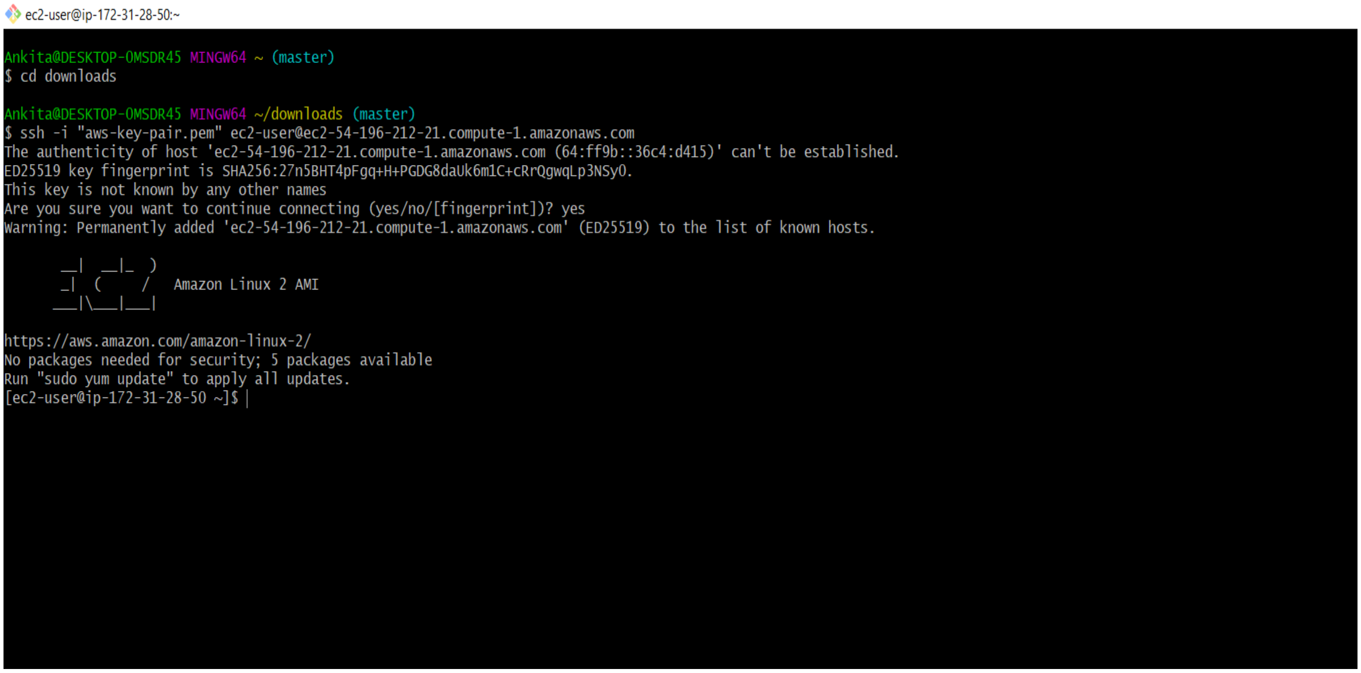
   ```
   ec2-54-196-212-21.compute-1.amazonaws.com
   ```

Example:

```
ssh -i "aws-key-pair.pem" ec2-user@ec2-54-196-212-21.compute-1.amazonaws.com
```

Note: In most cases, the guessed user name is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI user name.

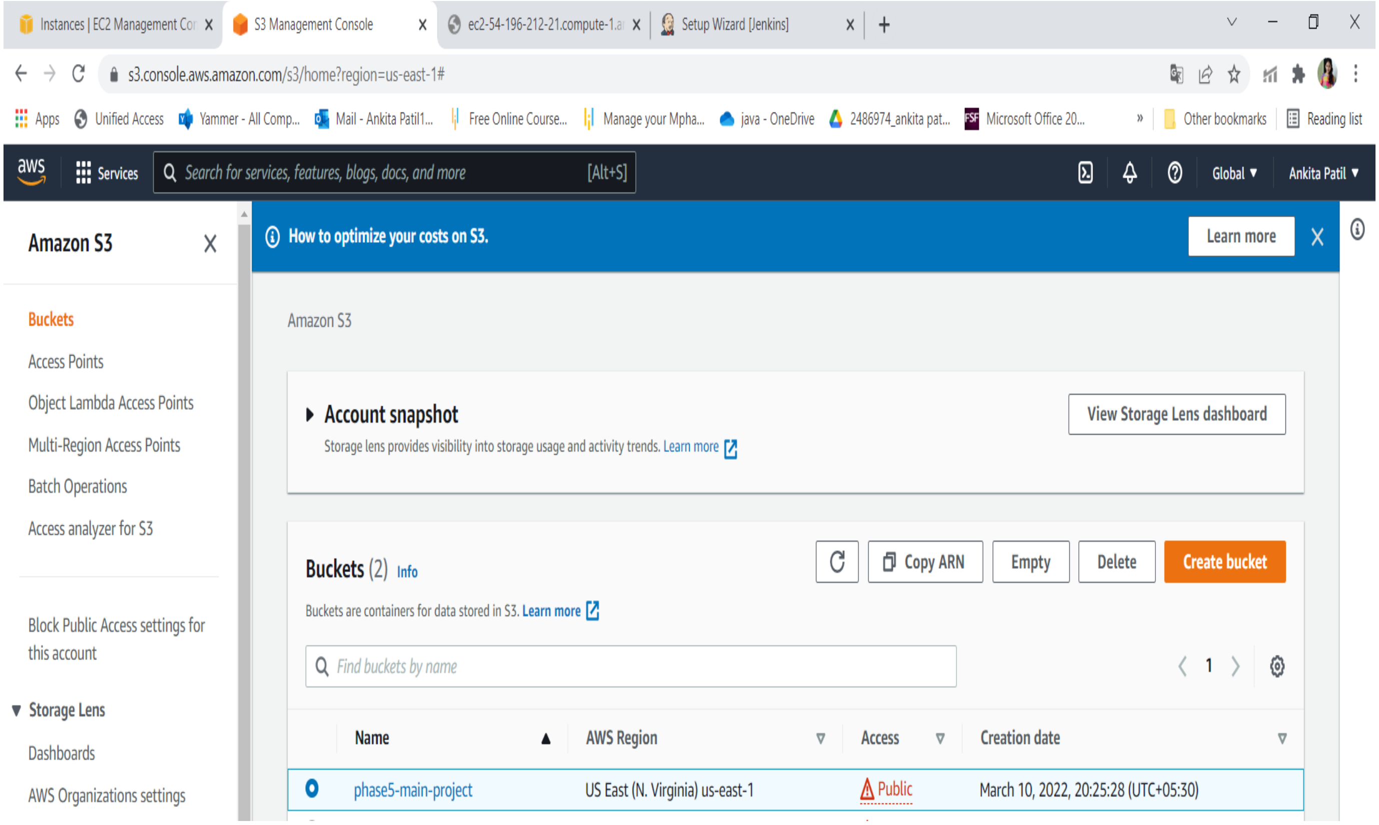
11. Click on Connect on EC2 dashboard & Run the ssh command provided



```
Ankita@DESKTOP-OMSDR45 MINGW64 ~ (master)
$ cd downloads

Ankita@DESKTOP-OMSDR45 MINGW64 ~/downloads (master)
$ ssh -i "aws-key-pair.pem" ec2-user@ec2-54-196-212-21.compute-1.amazonaws.com
The authenticity of host 'ec2-54-196-212-21.compute-1.amazonaws.com (64:ff9b::36c4:d415)' can't be established.
ED25519 key fingerprint is SHA256:27n5BHT4pFgq+H+PGDG8daUk6m1C+cRrQgwqLp3NSy0.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-196-212-21.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

       __|  __|_  )
       _|  (     /   Amazon Linux 2 AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-2/
No packages needed for security; 5 packages available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-28-50 ~]$ |
```

Step 4: Creating S3 Bucket:

1. Create Bucket to store jar file:

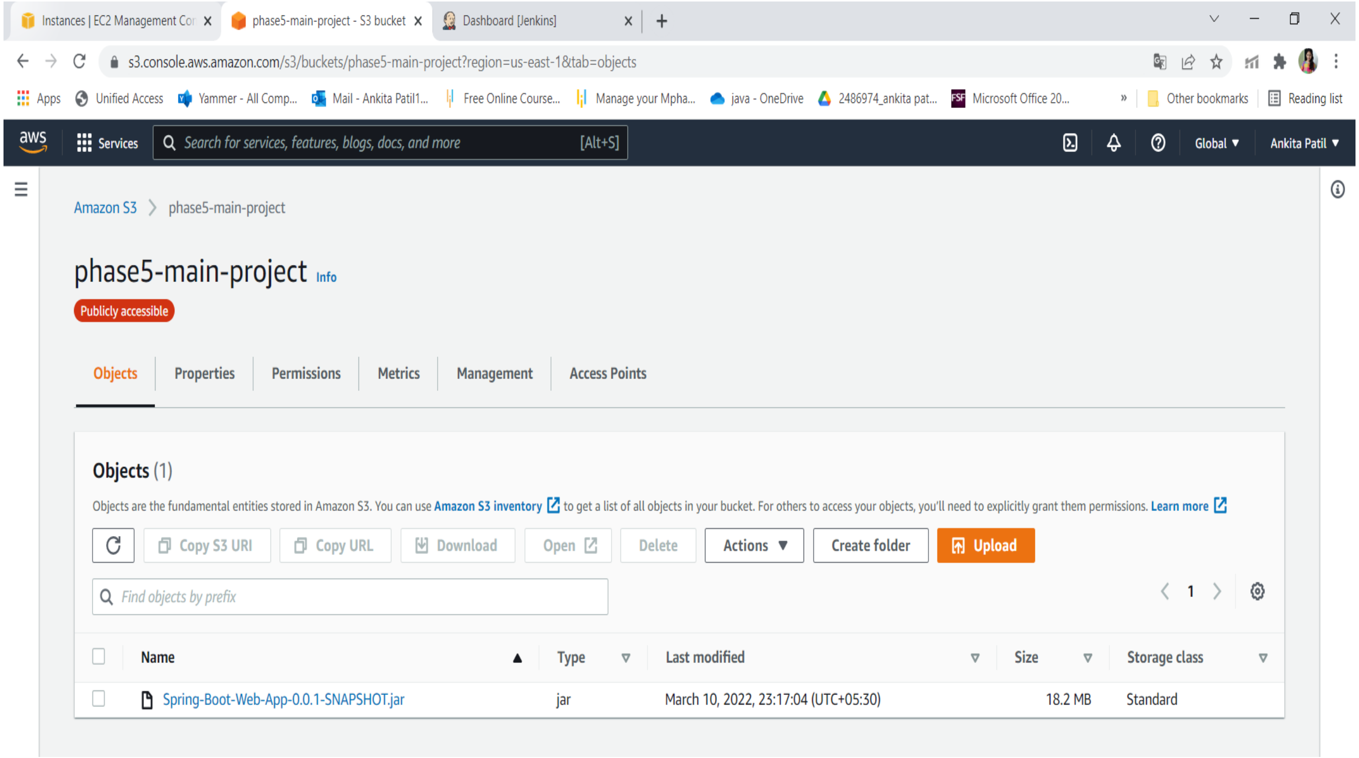


Storage lens provides visibility into storage usage and activity trends. Learn more

Buckets are containers for data stored in S3. Learn more

	Name	AWS Region	Access	Creation date
●	phase5-main-project	US East (N. Virginia) us-east-1	Public	March 10, 2022, 20:25:28 (UTC+05:30)

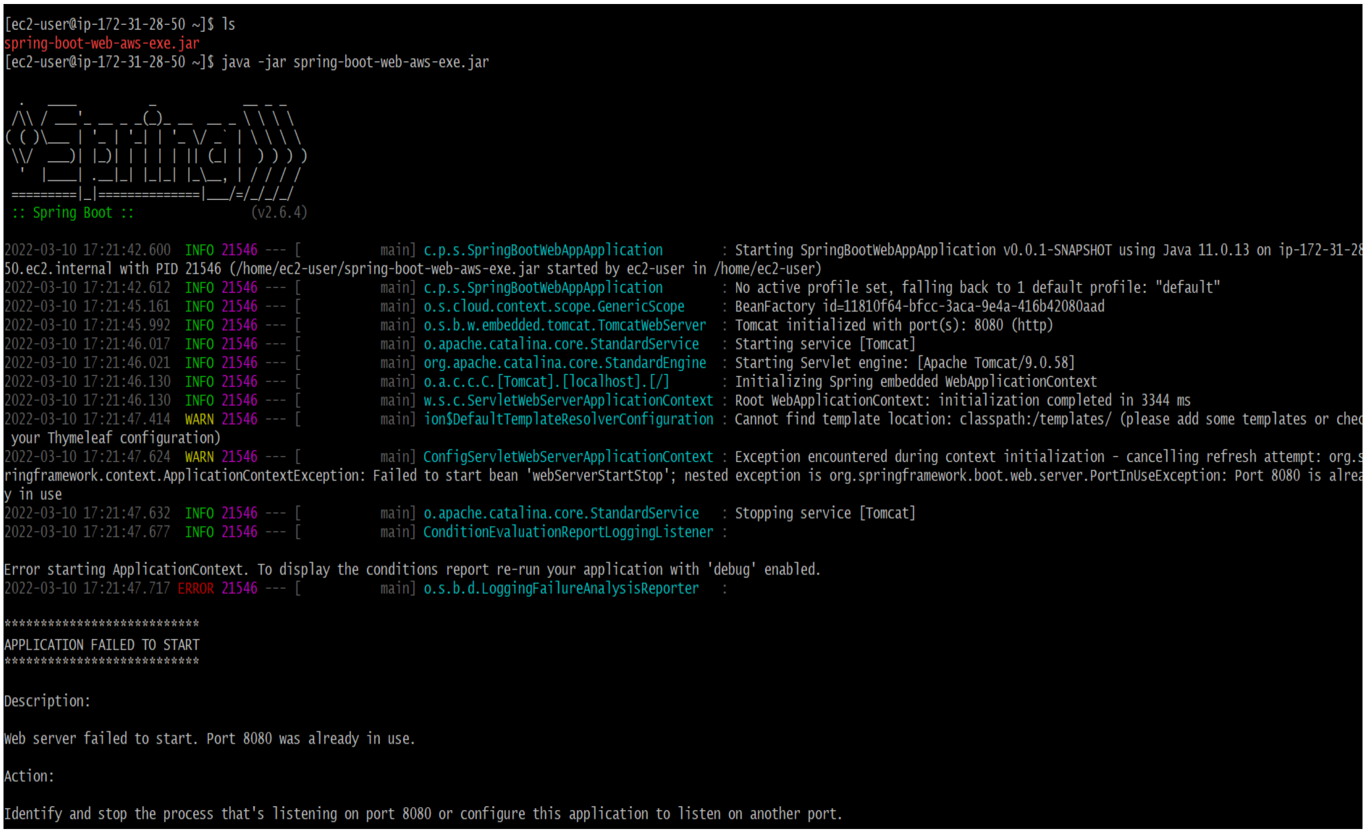
Step 5:Adding jar file:



Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

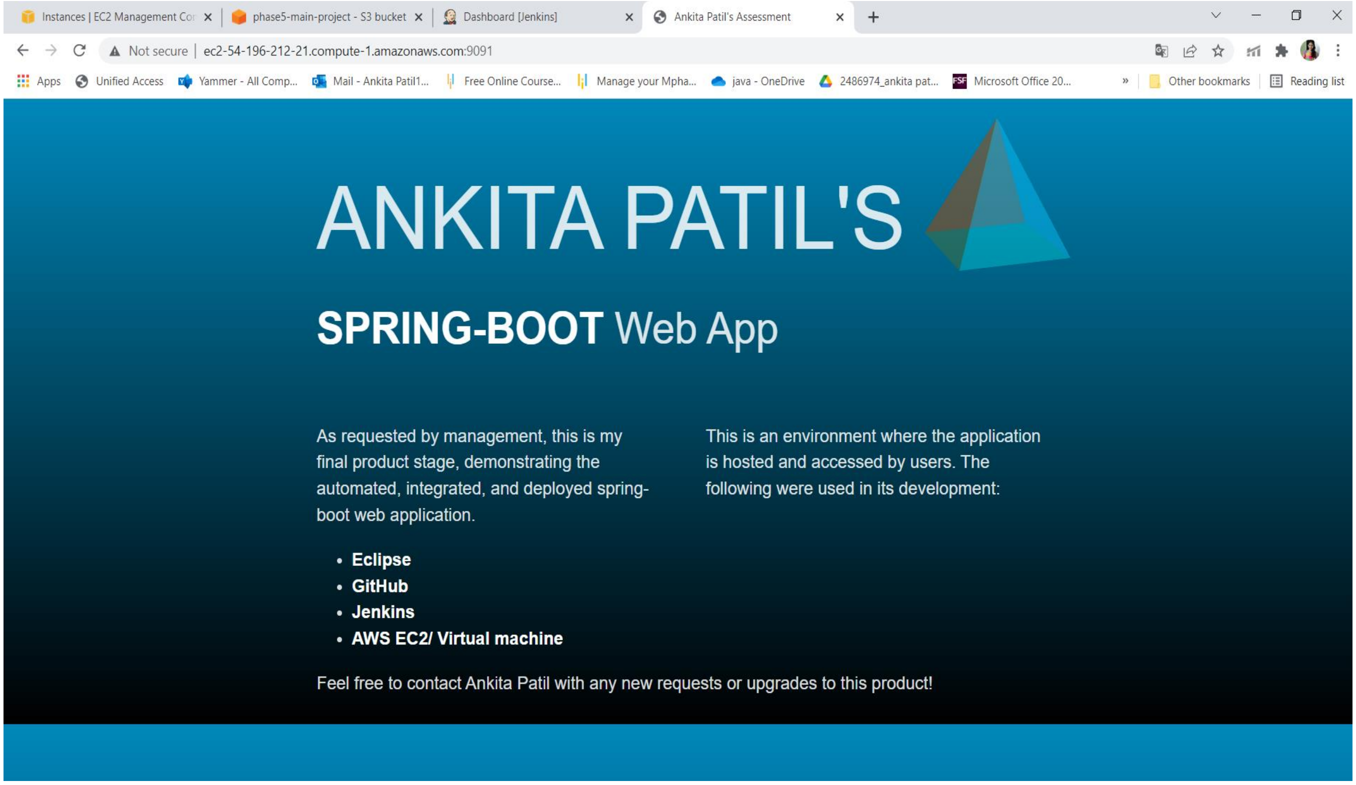
□	Name	Type	Last modified	Size	Storage class
□	Spring-Boot-Web-App-0.0.1-SNAPSHOT.jar	jar	March 10, 2022, 23:17:04 (UTC+05:30)	18.2 MB	Standard

- Run program through instance:



```
[ec2-user@ip-172-31-28-50 ~]$ ls
spring-boot-web-aws-exe.jar
[ec2-user@ip-172-31-28-50 ~]$ java -jar spring-boot-web-aws-exe.jar

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::                (v2.6.4)

2022-03-10 17:21:42.600  INFO 21546 --- [           main] c.p.s.SpringBootWebAppApplication        : Starting SpringBootWebAppApplication v0.0.1-SNAPSHOT using Java 11.0.13 on ip-172-31-28
50.ec2.internal with PID 21546 (/home/ec2-user/spring-boot-web-aws-exe.jar started by ec2-user in /home/ec2-user)
2022-03-10 17:21:42.612  INFO 21546 --- [           main] c.p.s.SpringBootWebAppApplication        : No active profile set, falling back to 1 default profile: "default"
2022-03-10 17:21:45.161  INFO 21546 --- [           main] o.s.cloud.context.scope.GenericScope     : BeanFactory id=11810f64-bfcc-3aca-9e4a-416b42080aad
2022-03-10 17:21:45.992  INFO 21546 --- [           main] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port(s): 8080 (http)
2022-03-10 17:21:46.017  INFO 21546 --- [           main] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2022-03-10 17:21:46.021  INFO 21546 --- [           main] org.apache.catalina.core.StandardEngine  : Starting Servlet engine: [Apache Tomcat/9.0.58]
2022-03-10 17:21:46.130  INFO 21546 --- [           main] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2022-03-10 17:21:46.130  INFO 21546 --- [           main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 3344 ms
2022-03-10 17:21:47.414  WARN 21546 --- [           main] ion$DefaultTemplateResolverConfiguration : Cannot find template location: classpath:/templates/ (please add some templates or chec
 your Thymeleaf configuration)
2022-03-10 17:21:47.624  WARN 21546 --- [           main] ConfigServletWebServerApplicationContext : Exception encountered during context initialization - cancelling refresh attempt: org.s
ringframework.context.ApplicationContextException: Failed to start bean 'webServerStartStop'; nested exception is org.springframework.boot.web.server.PortInUseException: Port 8080 is alrea
y in use
2022-03-10 17:21:47.632  INFO 21546 --- [           main] o.apache.catalina.core.StandardService   : Stopping service [Tomcat]
2022-03-10 17:21:47.677  INFO 21546 --- [           main] ConditionEvaluationReportLoggingListener :

Error starting ApplicationContext. To display the conditions report re-run your application with 'debug' enabled.
2022-03-10 17:21:47.717 ERROR 21546 --- [           main] o.s.b.d.LoggingFailureAnalysisReporter   :

***************************
APPLICATION FAILED TO START
***************************

Description:

Web server failed to start. Port 8080 was already in use.

Action:

Identify and stop the process that's listening on port 8080 or configure this application to listen on another port.
```

- Output:



ANKITA PATIL'S

SPRING-BOOT Web App

As requested by management, this is my final product stage, demonstrating the automated, integrated, and deployed spring-boot web application.

This is an environment where the application is hosted and accessed by users. The following were used in its development:

- **Eclipse**
- **GitHub**
- **Jenkins**
- **AWS EC2/ Virtual machine**

Feel free to contact Ankita Patil with any new requests or upgrades to this product!

Step 6: Pushing the code to your GitHub repositories

- Open your command prompt and navigate to the folder where you have created your files.

```
cd <folder path>
```

- Initialize your repository using the following command:

```
git init
```

- Add all the files to your git repository using the following command:

```
git add .
```

- Commit the changes using the following command:

```
git commit .  -m “Changes have been committed.”
```

- Push the files to the folder you initially created using the following command:

```
git push -u origin master
```