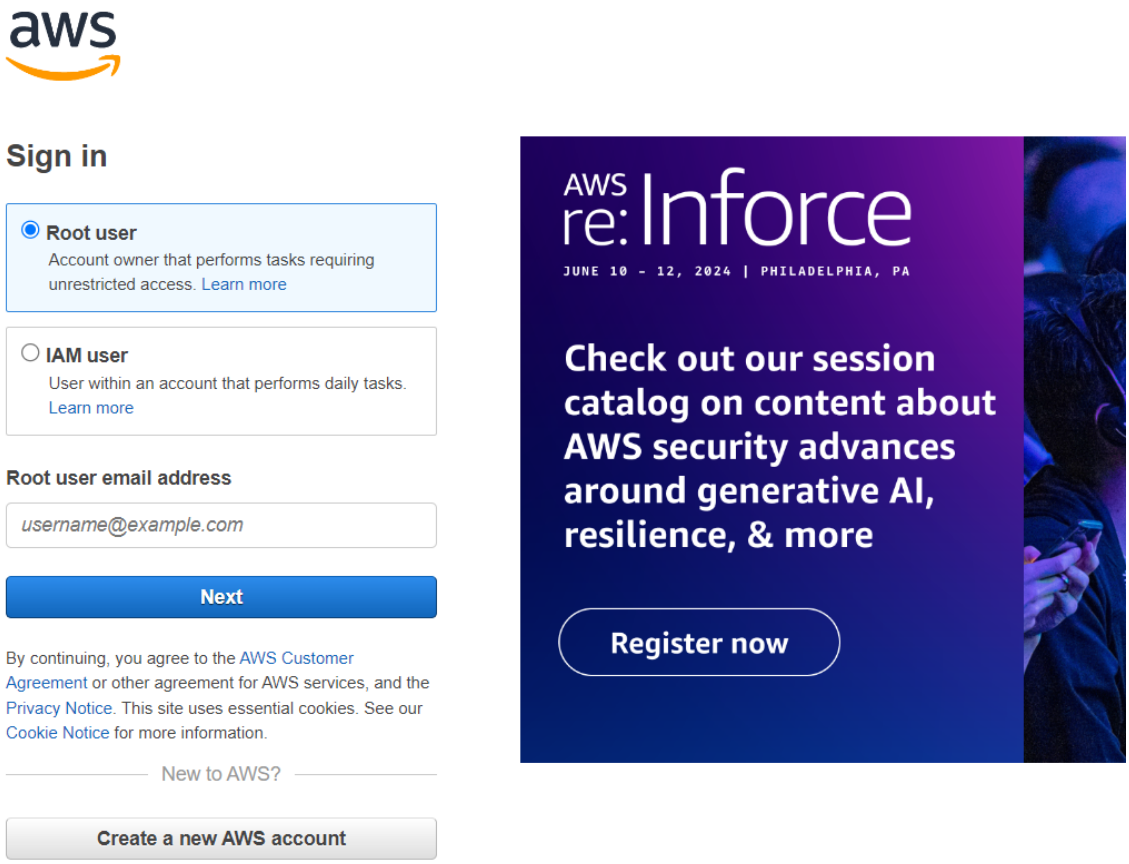
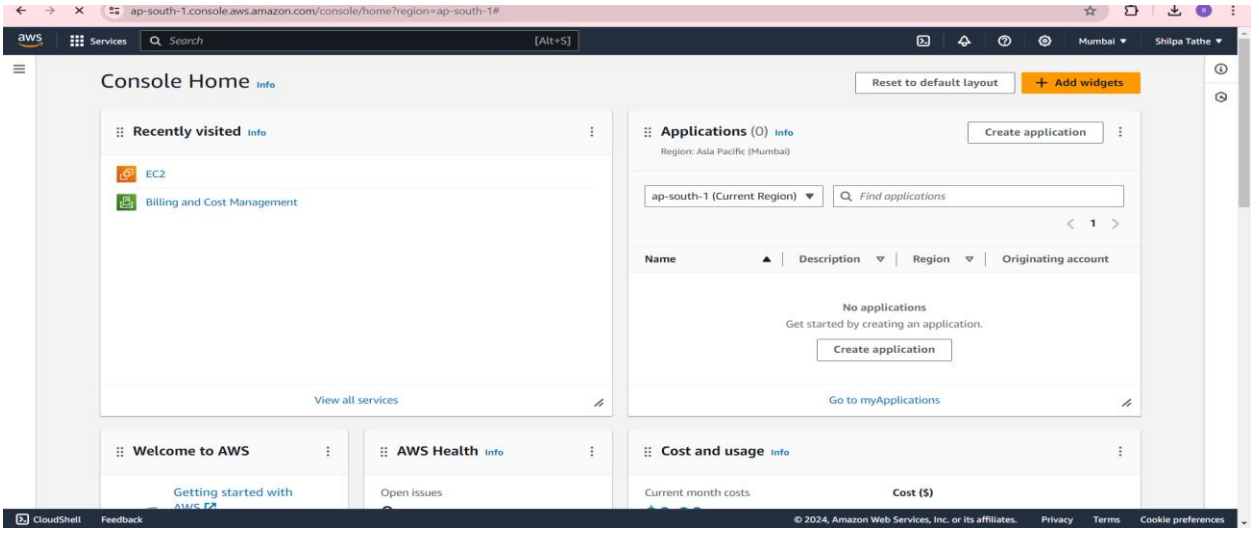
AWS Assignment

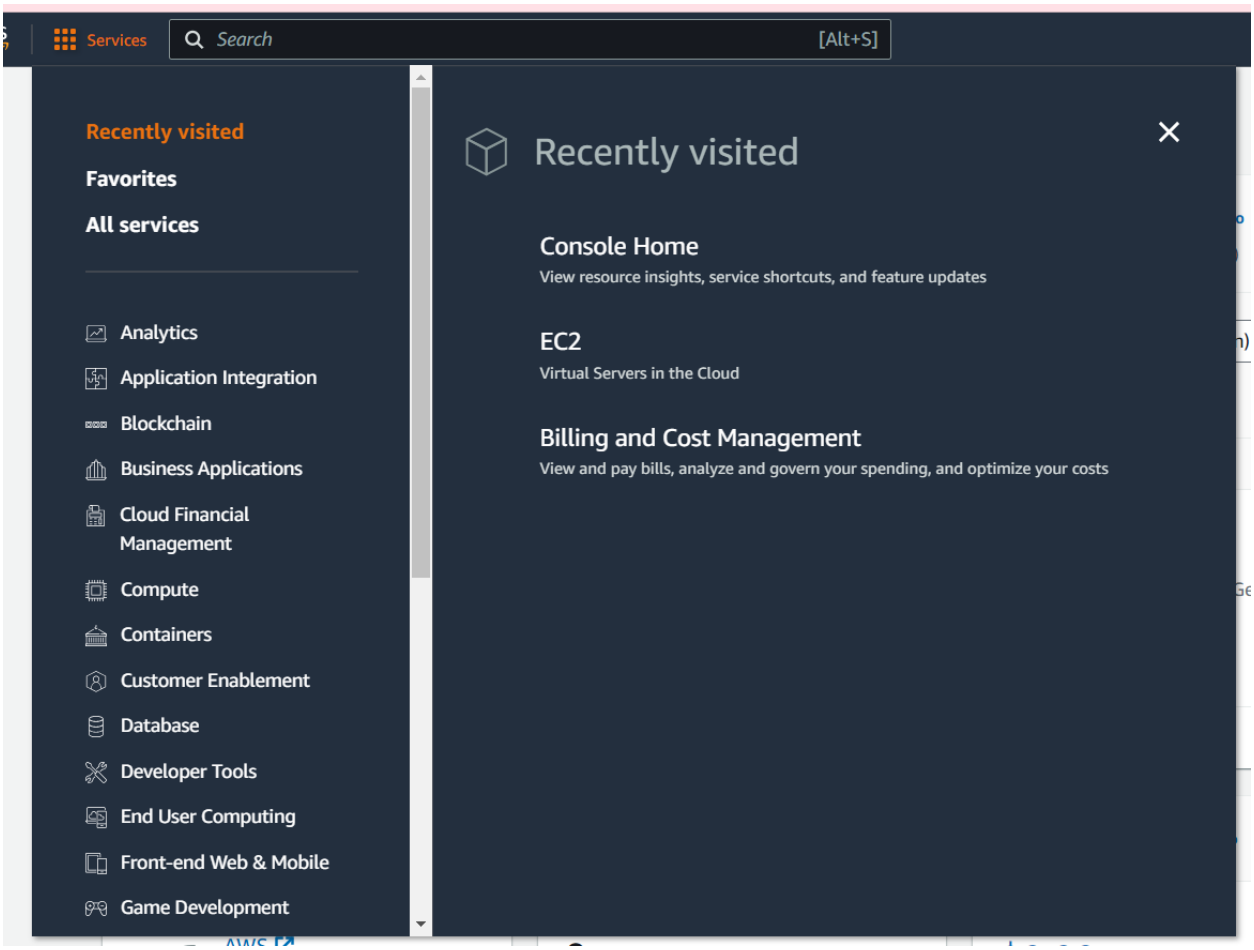
1]Log in : First of all we have to log in into AWS account by using email id and password.



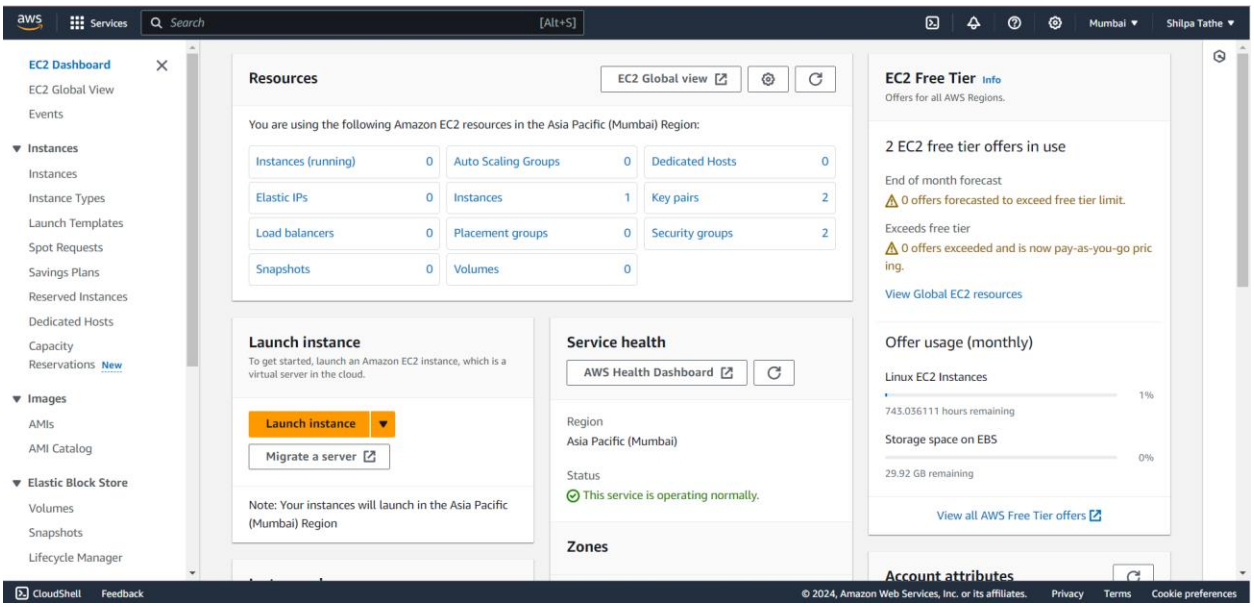
2]After log in ,open AWS console and below page will appear



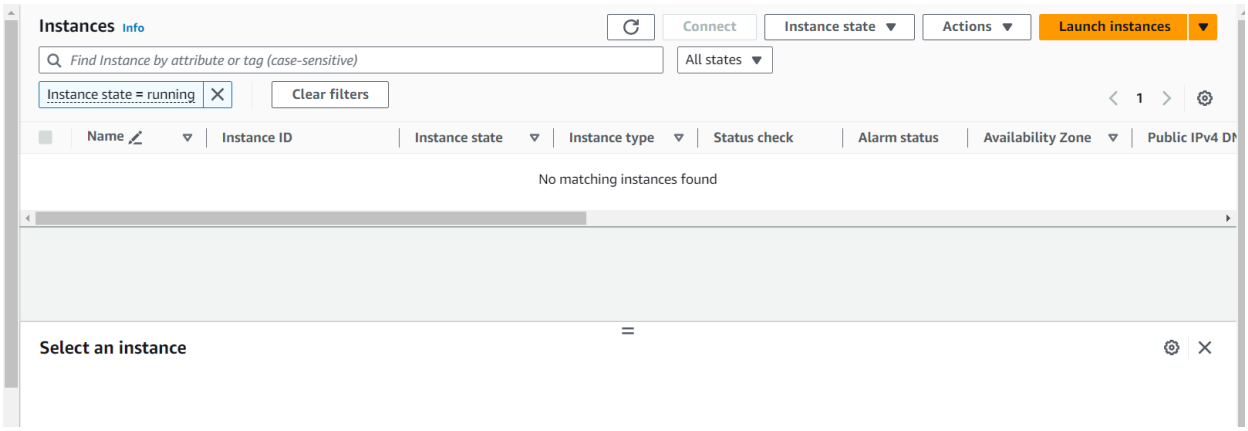
3]Click on services and select compute .After that click on EC2 Virtual Server



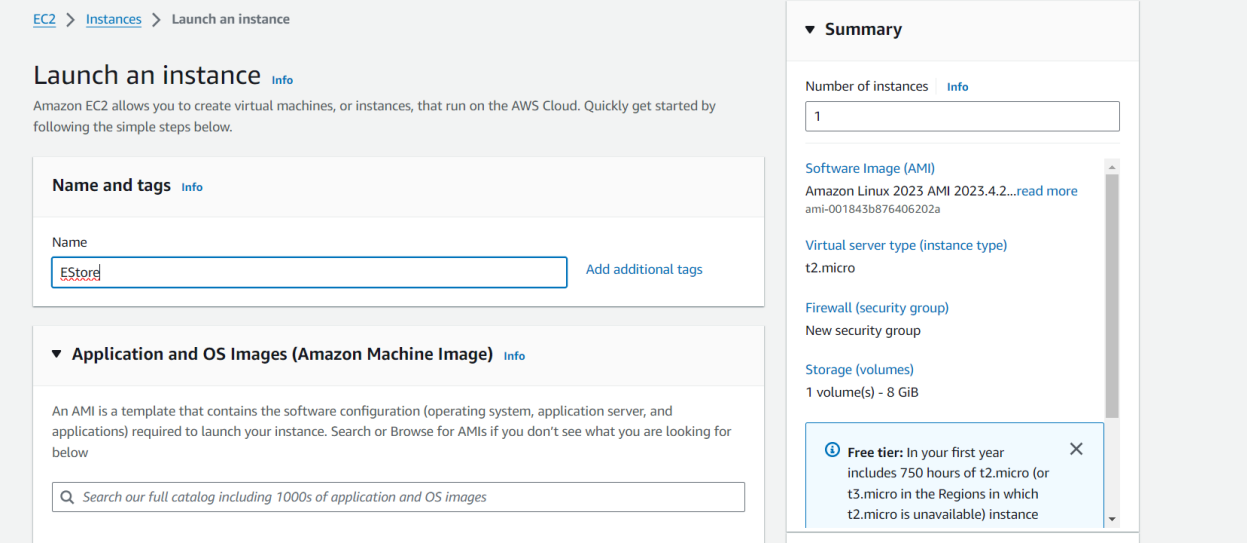
4]Click on instance(running)



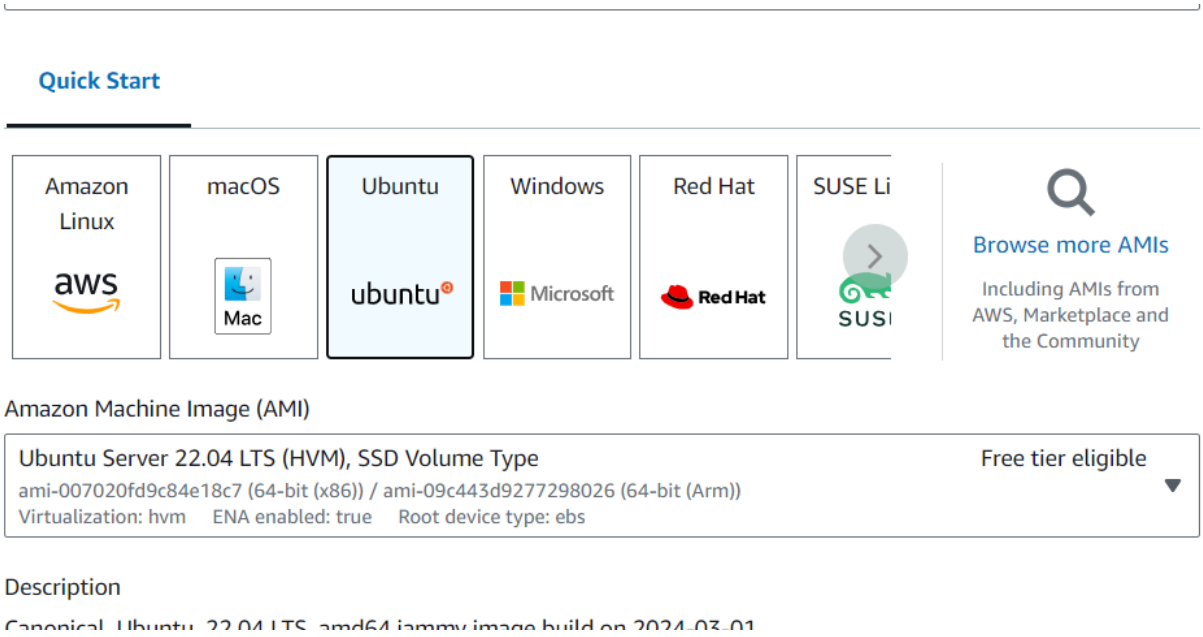
5]After that new page will be seen where you have to click on “Launch instances”



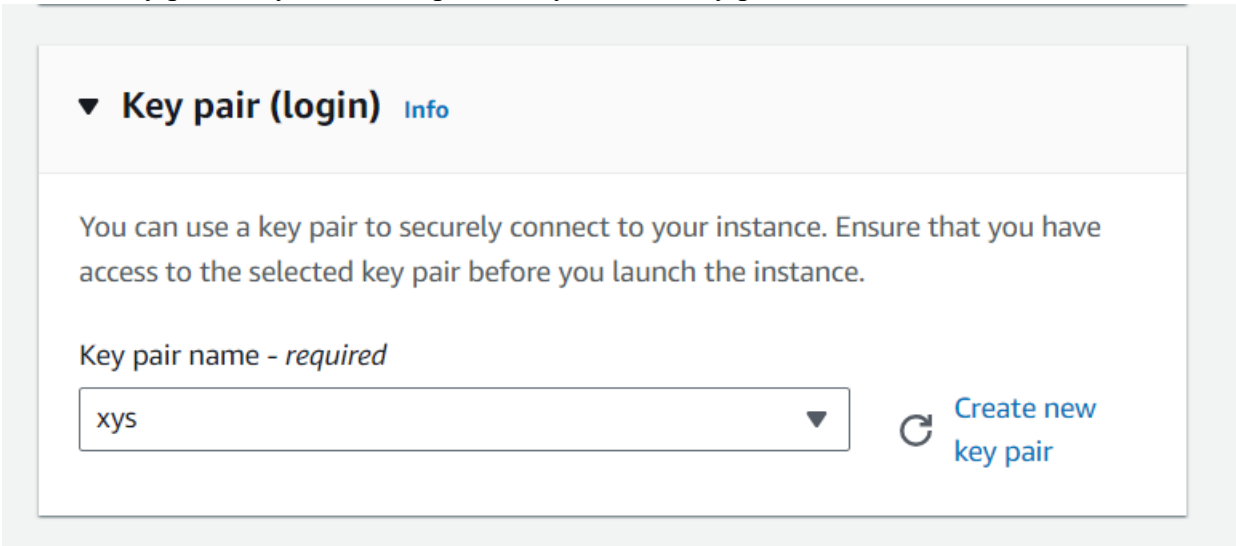
Give name to your instance:



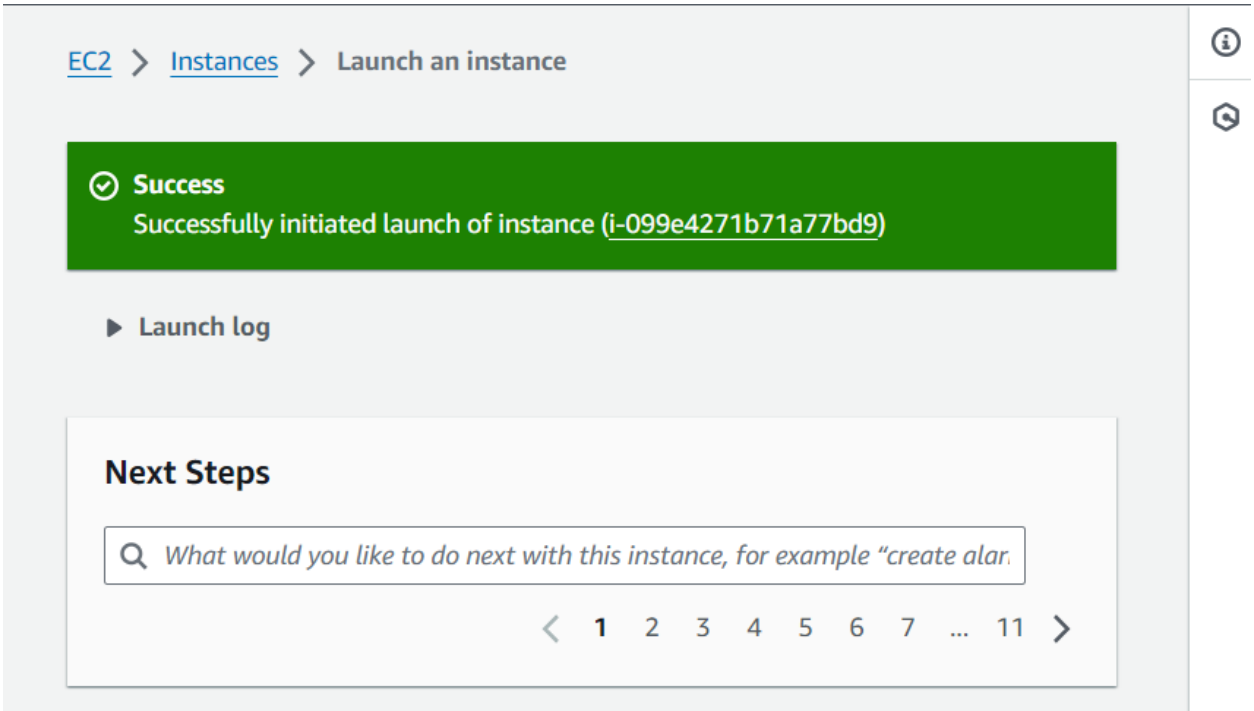
After that select ubuntu:



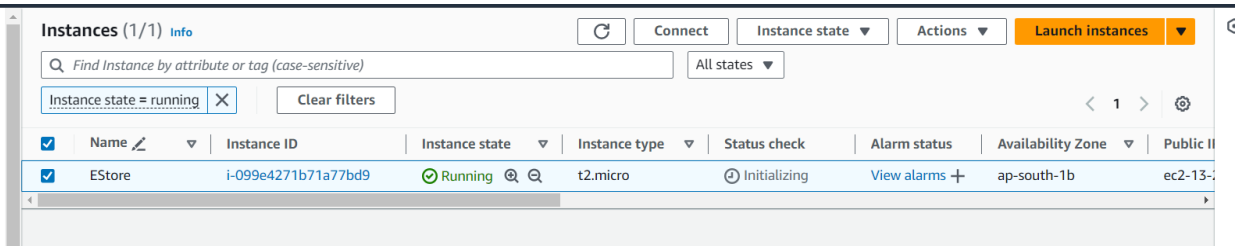
Create “Key pair” or you can use previously created Key pair:



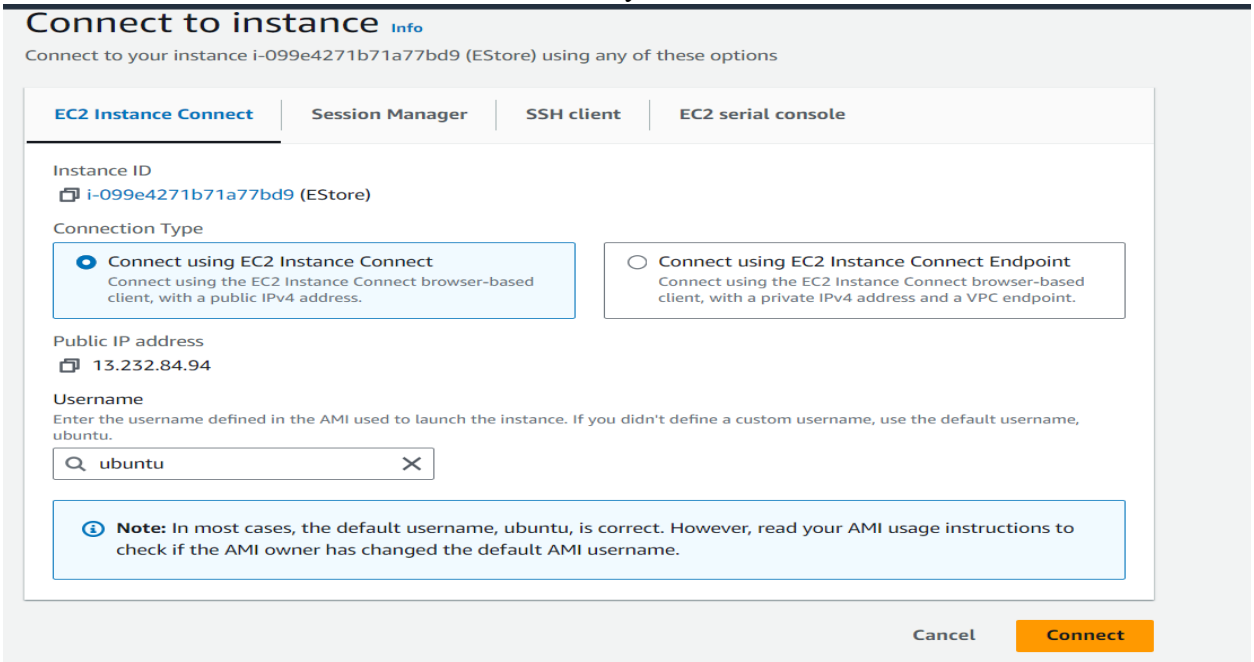
After that click on “Launch instance” and your Instance will be created successfully!!



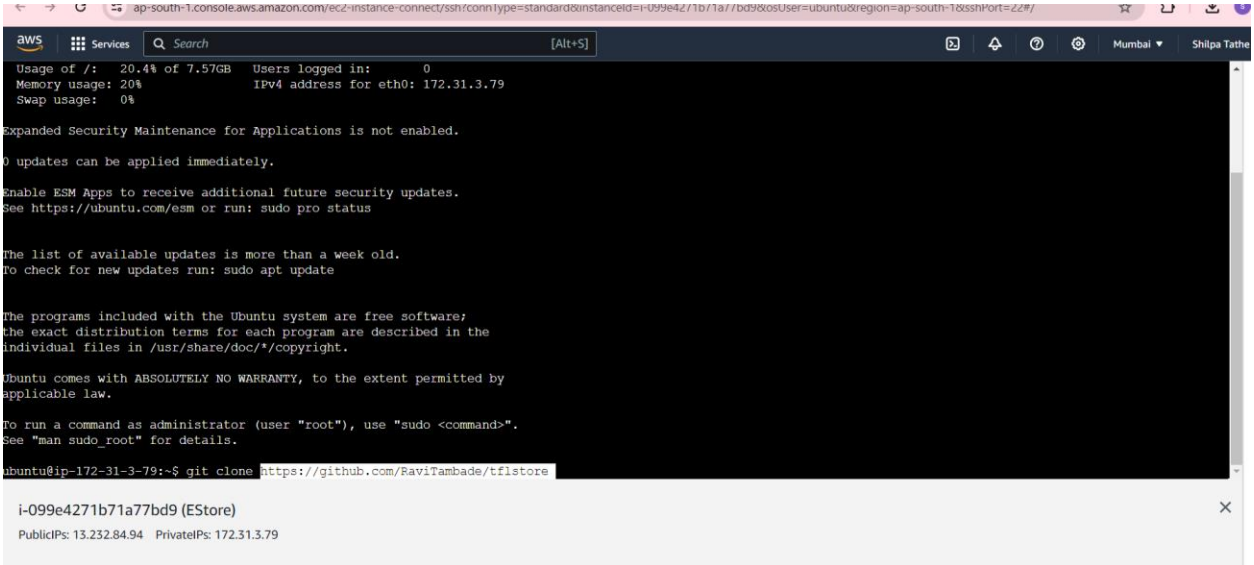
6]Now you have to run the instance for that check the box of instance and click on connect:



After that new window will be there and on that you have to click on connect:



After that this window will come:



7]Following steps are to create docker container:

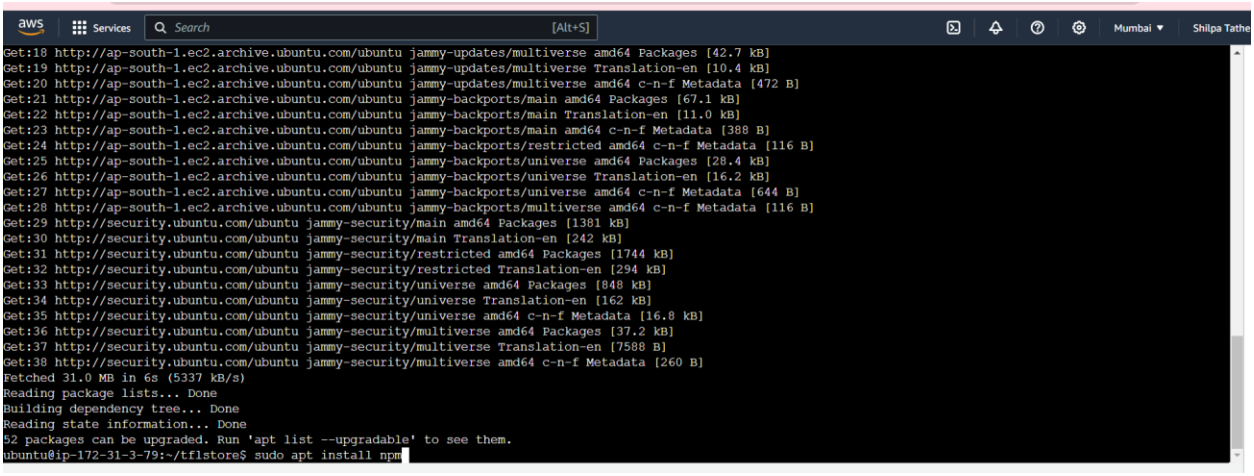
>git clone

>cd tflstore

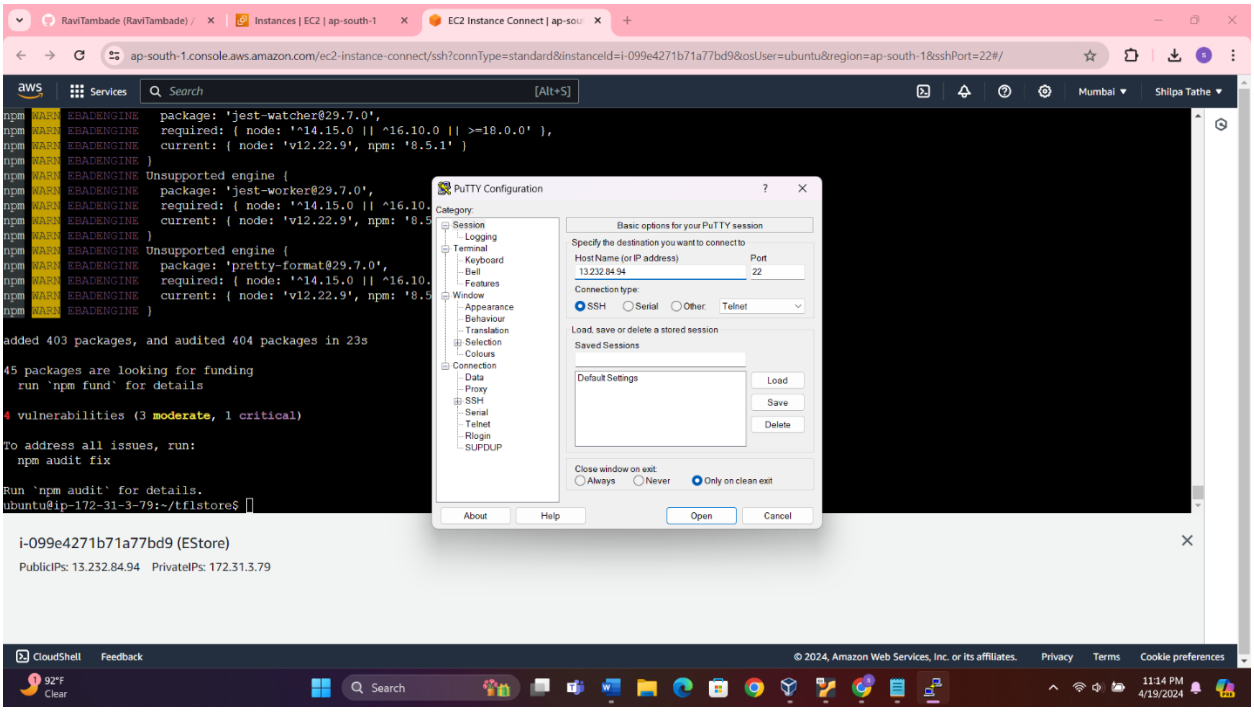
>sudo apt update

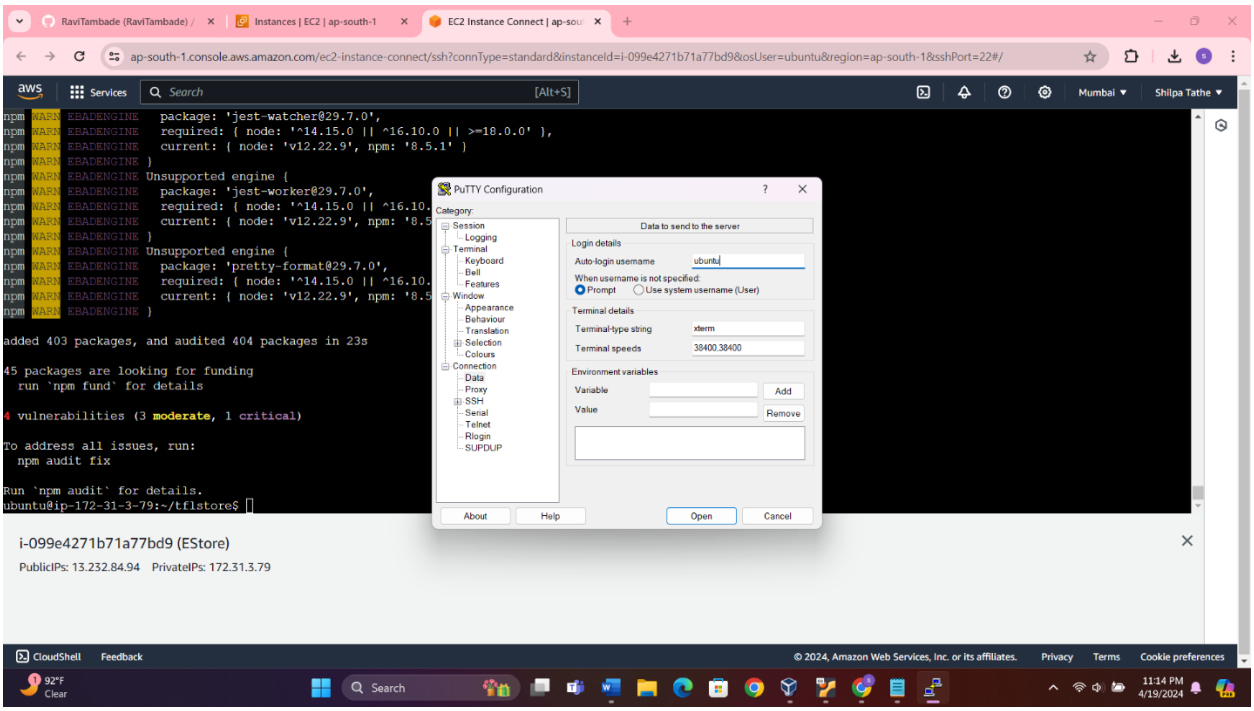
>sudo apt install npm

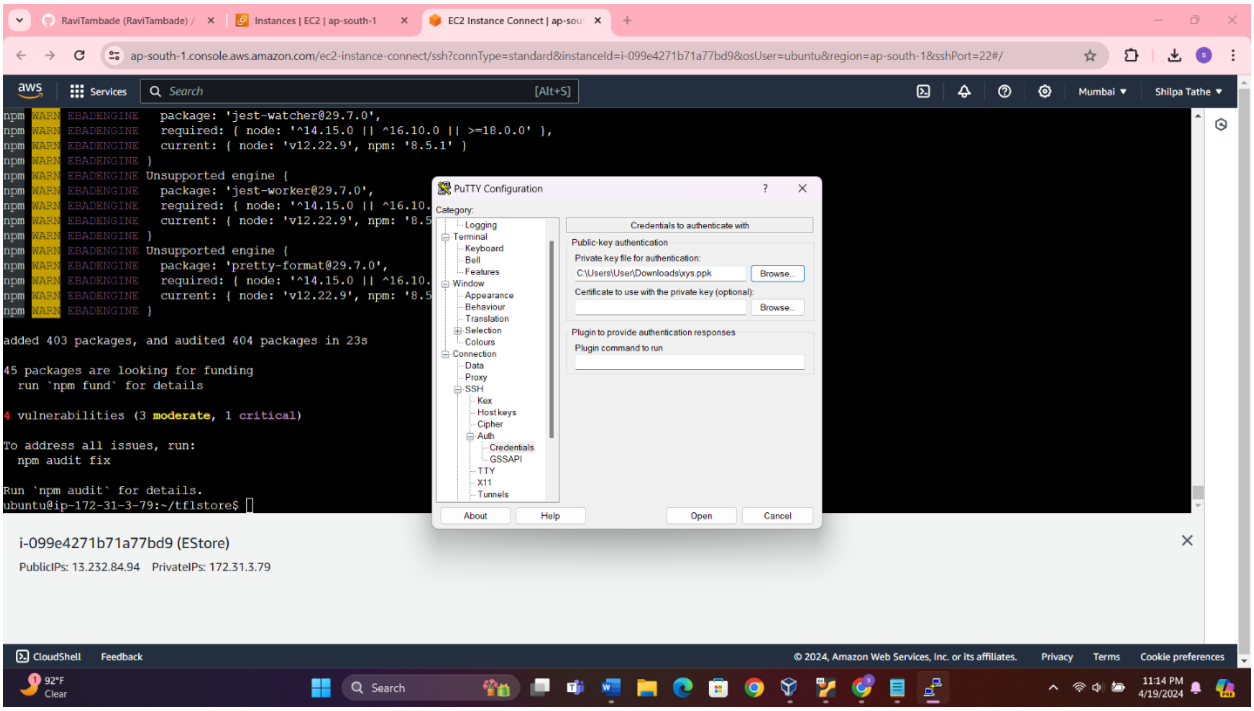
>npm install



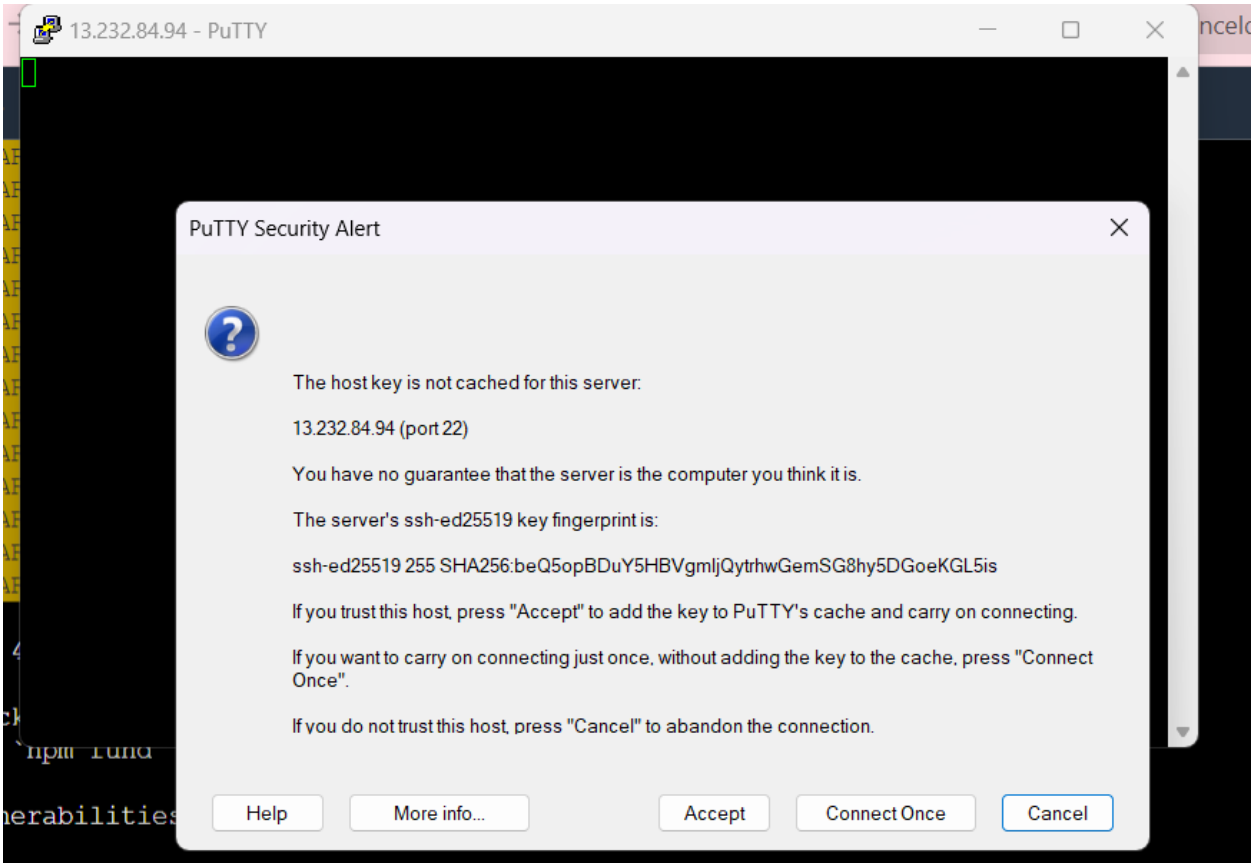
Set the putty Setting like this:




RaviTambade (RaviTambade) /
Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-sou
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-099e4271b71a77bd9&osUser=ubuntu&region=ap-south-1&sshPort=22#/
Services
Search
[Alt+S]
Mumbai
Shilpa Tathe
npm WARN EBADENGINE package: 'jest-watcher@29.7.0',
npm WARN EBADENGINE required: { node: '^14.15.0 || ^16.10.0 || >=18.0.0' },
npm WARN EBADENGINE current: { node: 'v12.22.9', npm: '8.5.1' }
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE package: 'jest-worker@29.7.0',
npm WARN EBADENGINE required: { node: '^14.15.0 || ^16.10.
npm WARN EBADENGINE current: { node: 'v12.22.9', npm: '8.5
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE package: 'pretty-format@29.7.0',
npm WARN EBADENGINE required: { node: '^14.15.0 || ^16.10.
npm WARN EBADENGINE current: { node: 'v12.22.9', npm: '8.5
npm WARN EBADENGINE }
added 403 packages, and audited 404 packages in 23s
45 packages are looking for funding
run `npm fund` for details
4 vulnerabilities (3 moderate, 1 critical)
To address all issues, run:
npm audit fix
Run `npm audit` for details.
ubuntu@ip-172-31-3-79:~/tflstore$
PuTTY Configuration
Category:
Session
Logging
Terminal
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
SSH
Serial
Telnet
Rlogin
SUPDUP
Data to send to the server
Login details
Auto-login username
ubuntu
When username is not specified:
Prompt
Use system username (User)
Terminal details
Terminal-type string
xterm
Terminal speeds
38400,38400
Environment variables
Variable
Add
Value
Remove
About
Help
Open
Cancel
i-099e4271b71a77bd9 (EStore)
PublicIPs: 13.232.84.94 PrivateIPs: 172.31.3.79
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
92°F
Clear
Search
11:14 PM
4/19/2024


RaviTambade (RaviTambade) /
Instances | EC2 | ap-south-1
EC2 Instance Connect | ap-sou
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-099e4271b71a77bd9&osUser=ubuntu&region=ap-south-1&sshPort=22#/
Services
Search
[Alt+S]
Mumbai
Shilpa Tathe
npm WARN EBADENGINE package: 'jest-watcher@29.7.0',
npm WARN EBADENGINE required: { node: '^14.15.0 || ^16.10.0 || >=18.0.0' },
npm WARN EBADENGINE current: { node: 'v12.22.9', npm: '8.5.1' }
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE package: 'jest-worker@29.7.0',
npm WARN EBADENGINE required: { node: '^14.15.0 || ^16.10.
npm WARN EBADENGINE current: { node: 'v12.22.9', npm: '8.5
npm WARN EBADENGINE }
npm WARN EBADENGINE Unsupported engine {
npm WARN EBADENGINE package: 'pretty-format@29.7.0',
npm WARN EBADENGINE required: { node: '^14.15.0 || ^16.10.
npm WARN EBADENGINE current: { node: 'v12.22.9', npm: '8.5
npm WARN EBADENGINE }
added 403 packages, and audited 404 packages in 23s
45 packages are looking for funding
run `npm fund` for details
4 vulnerabilities (3 moderate, 1 critical)
To address all issues, run:
npm audit fix
Run `npm audit` for details.
ubuntu@ip-172-31-3-79:~/tflstore$
PuTTY Configuration
Category:
Logging
Terminal
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
SSH
Kex
Host keys
Cipher
Auth
Credentials
GSSAPI
TTY
X11
Tunnels
Credentials to authenticate with
Public-key authentication
Private key file for authentication:
C:\Users\User\Downloads\xys.ppk
Browse...
Certificate to use with the private key (optional):
Browse...
Plugin to provide authentication responses
Plugin command to run
About
Help
Open
Cancel
i-099e4271b71a77bd9 (EStore)
PublicIPs: 13.232.84.94 PrivateIPs: 172.31.3.79
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
92°F
Clear
Search
11:14 PM
4/19/2024

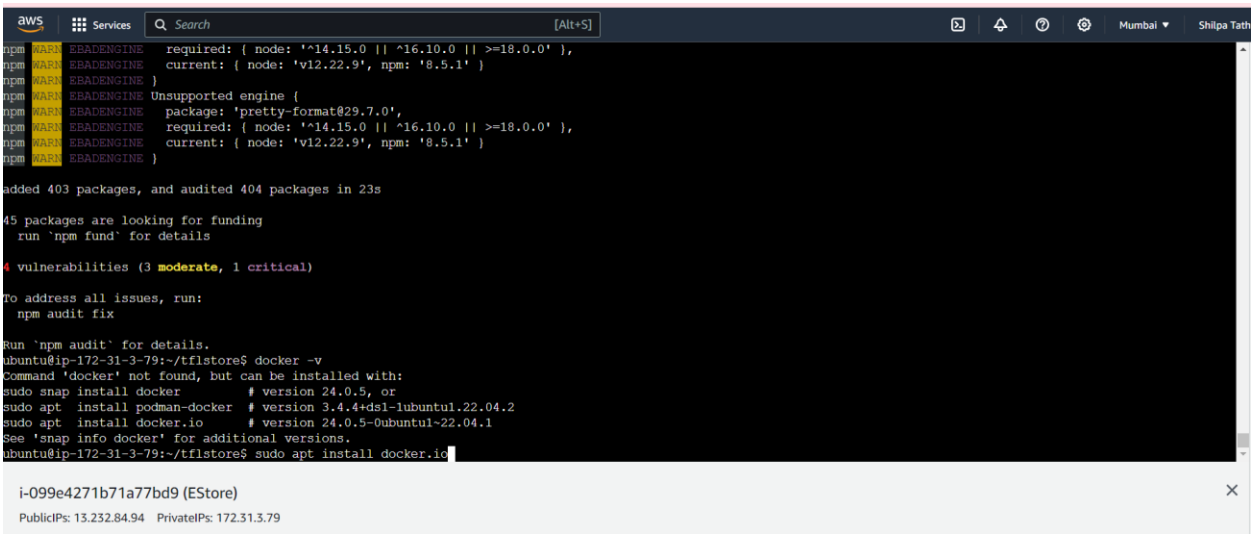
Click on accept:



Now see if docker already present or not .If it is not present then run command

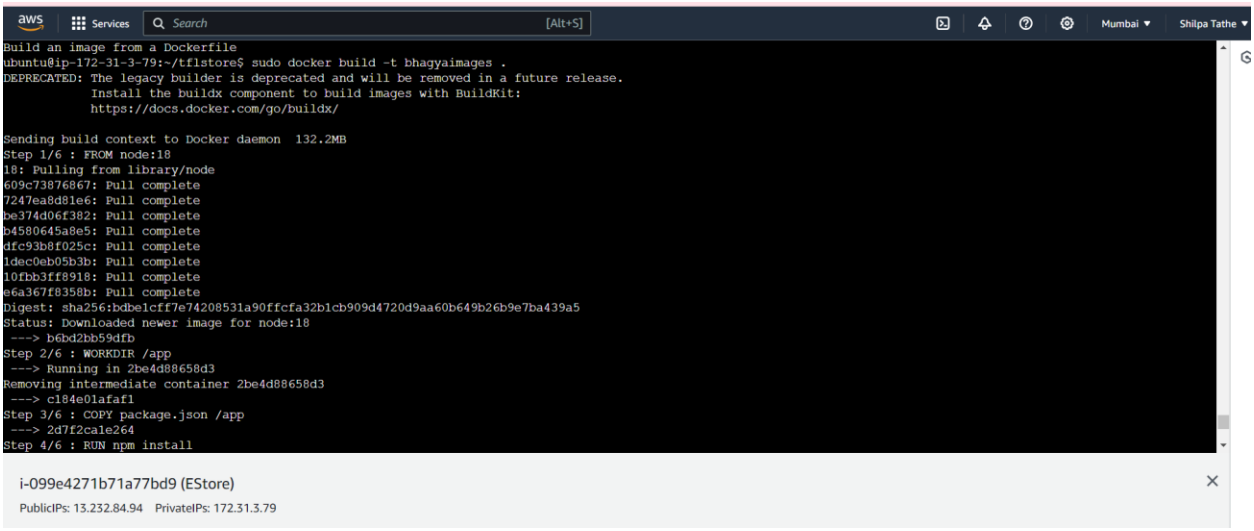
>docker -v

>sudo apt install docker.io



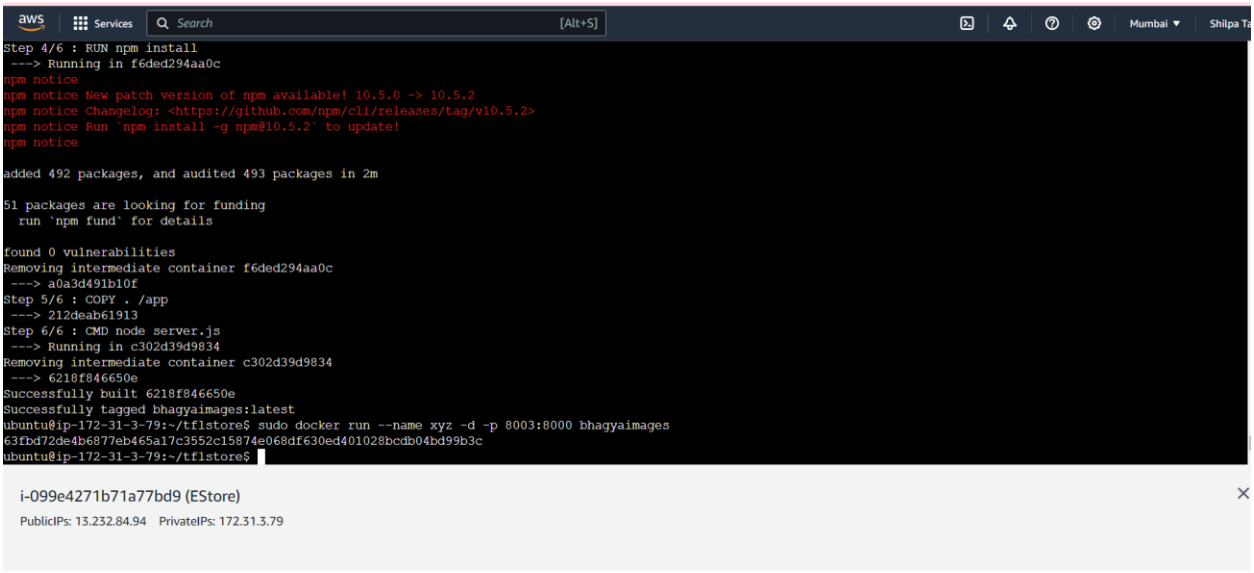
Now to build image:

>sudo docker build -t bhagyaimages .



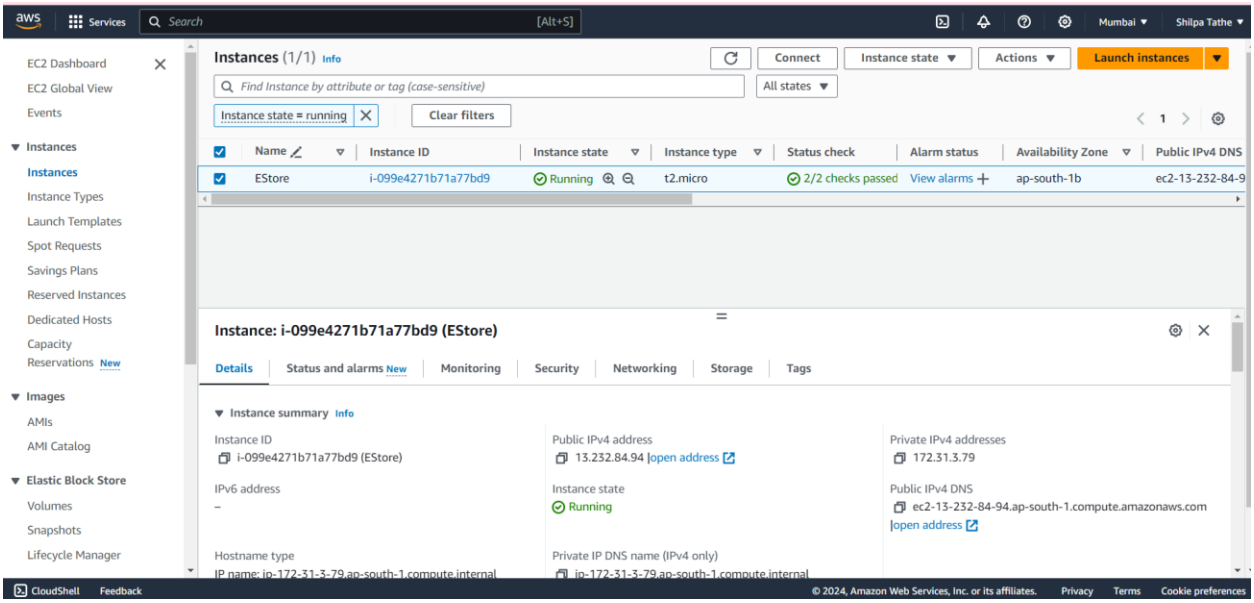
Now to create container:

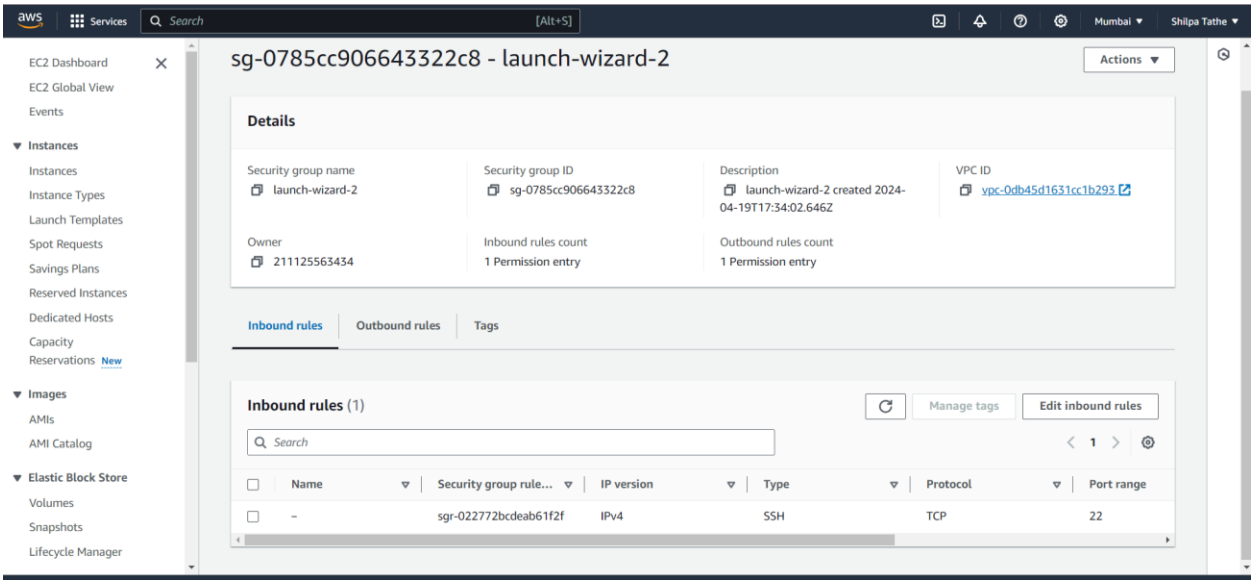
>sudo docker run –name xyz -d -p 8003:8000 bhagyaimages



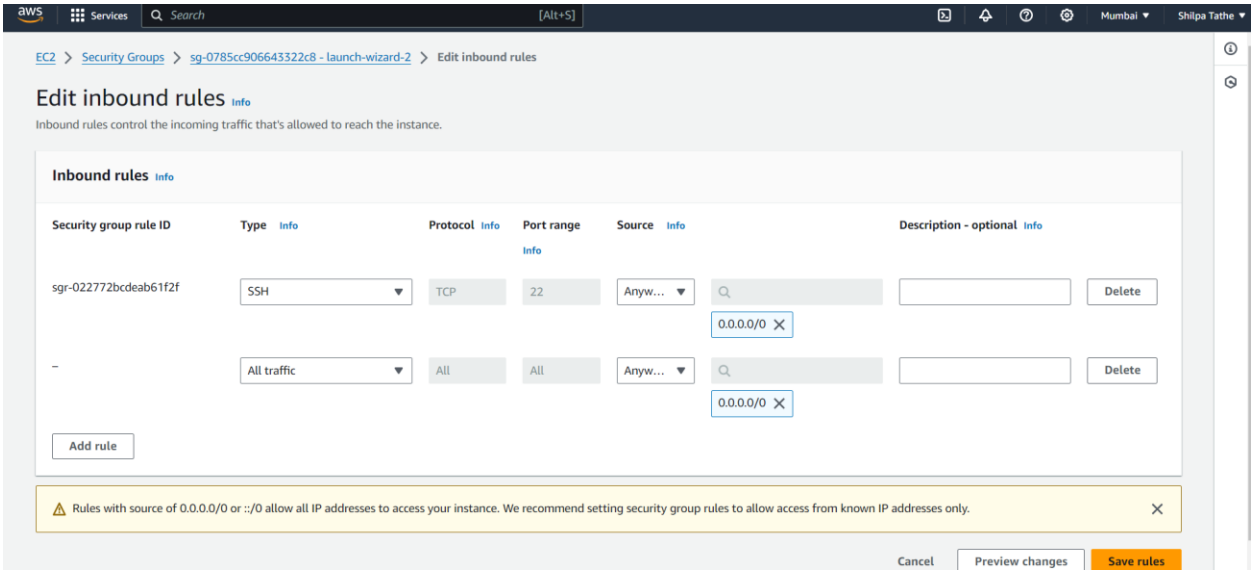
Now set the setting of security:

Go to the security after click on edit inbound rule

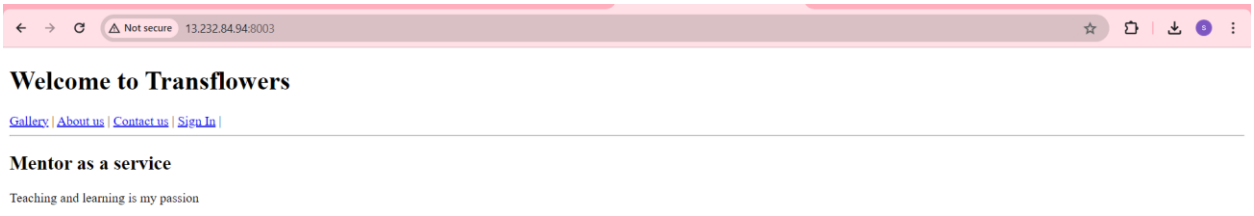




Now set Type to All traffic and source to anywhere IPV4 and click on save rules:



8]Type a IP address in search bar with port number and finally website will open like this!!

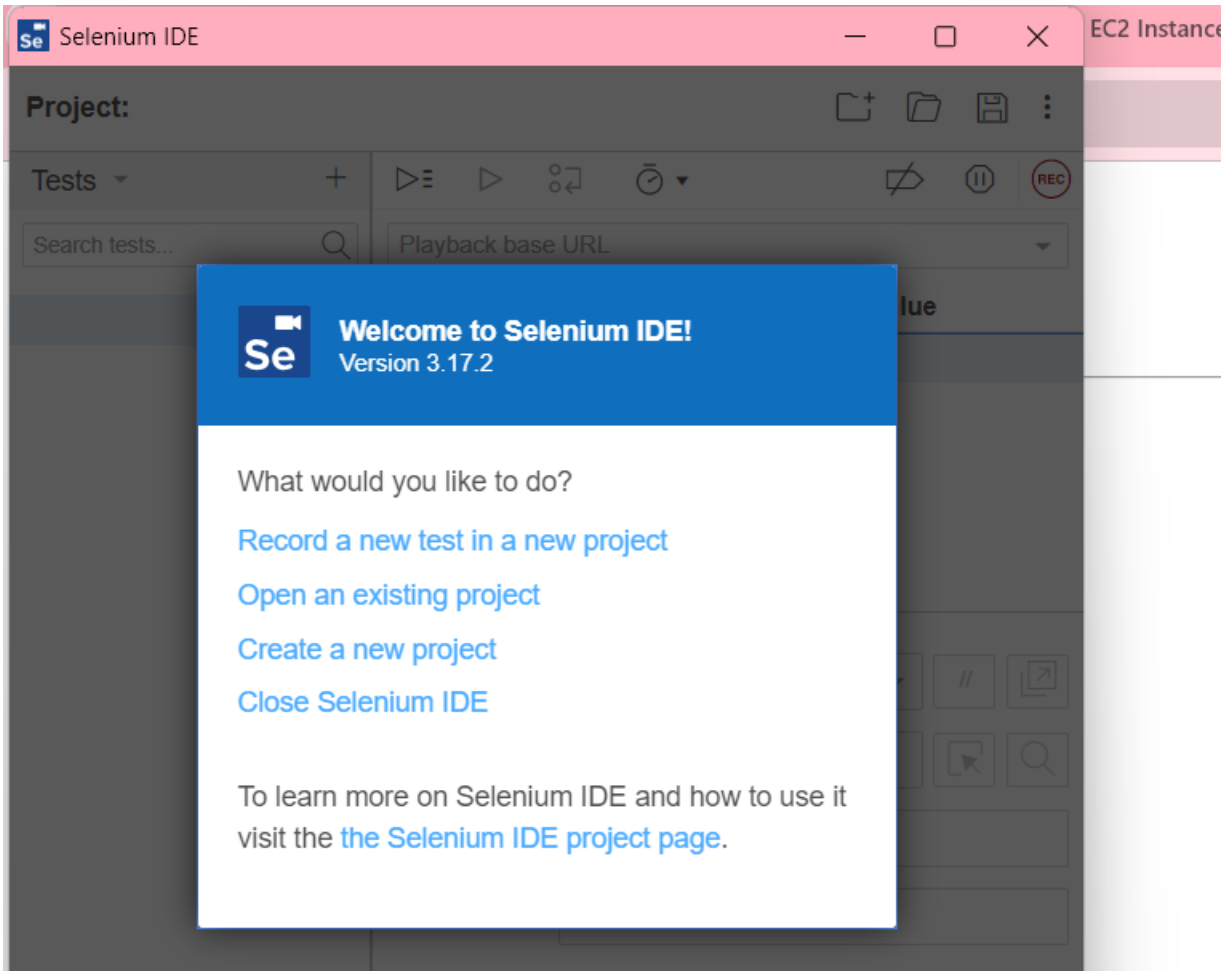


9]TESTING: Now for testing part

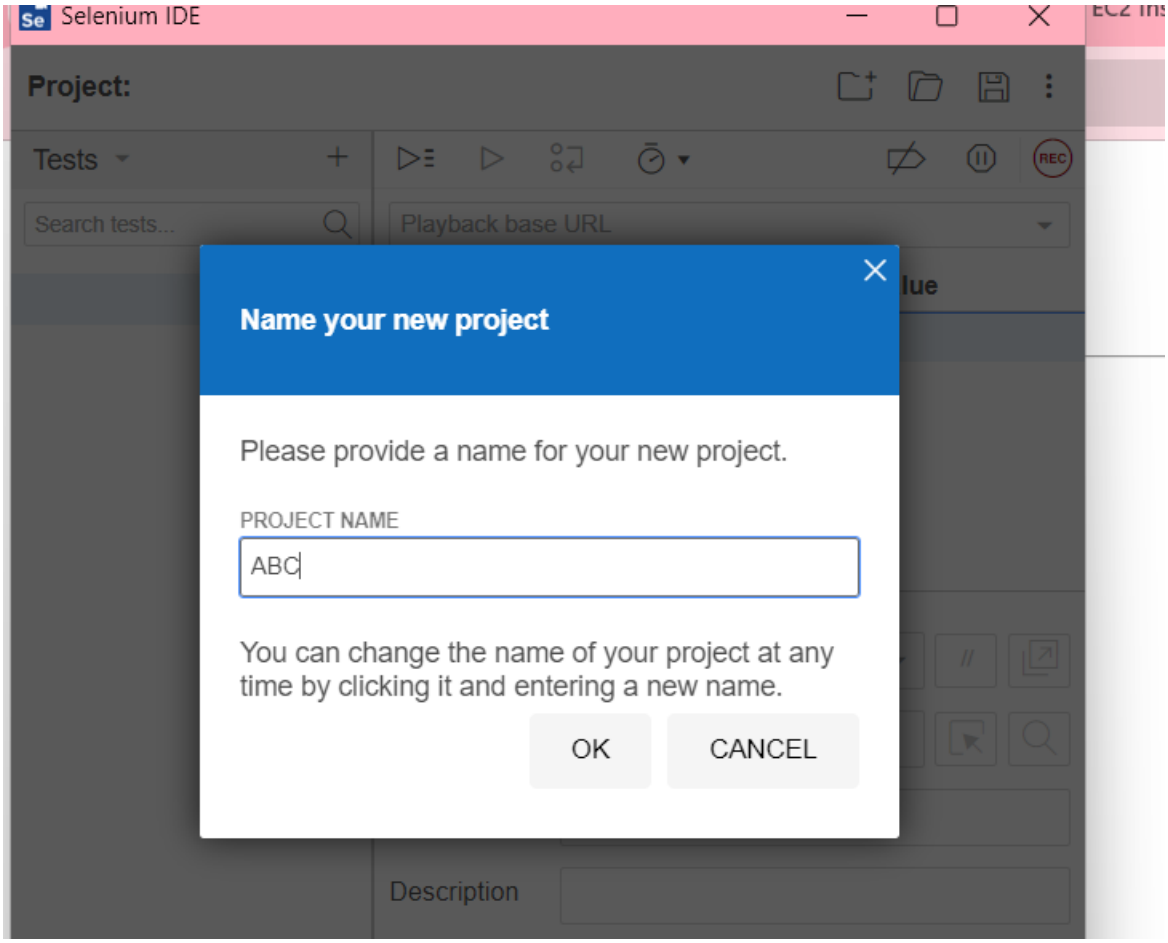
Download the selenium for chrome and add extension of it in chrome

Double click on selenium IDE

Select first option: Record a new test in a new project

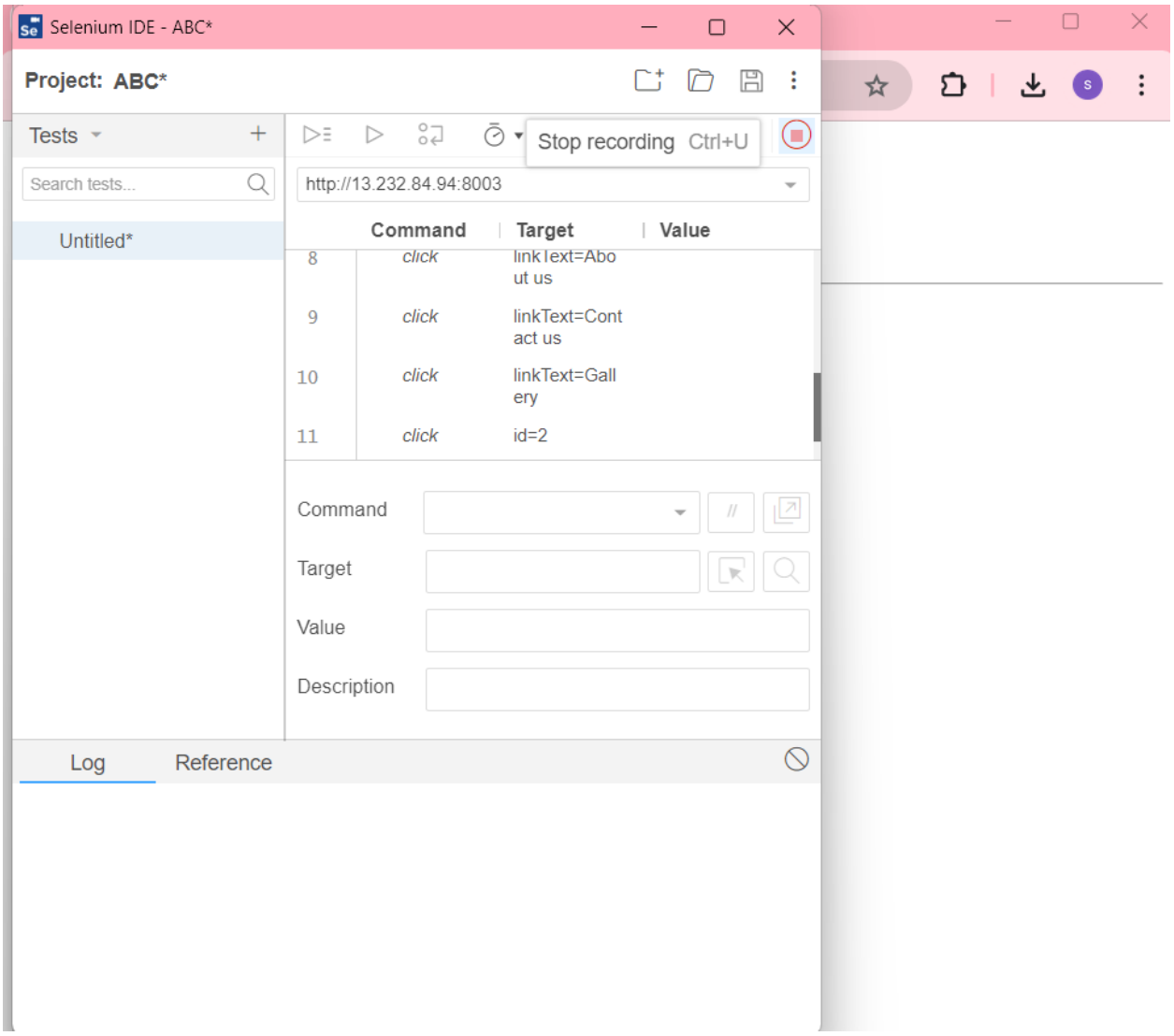


Set the Project name and click on OK

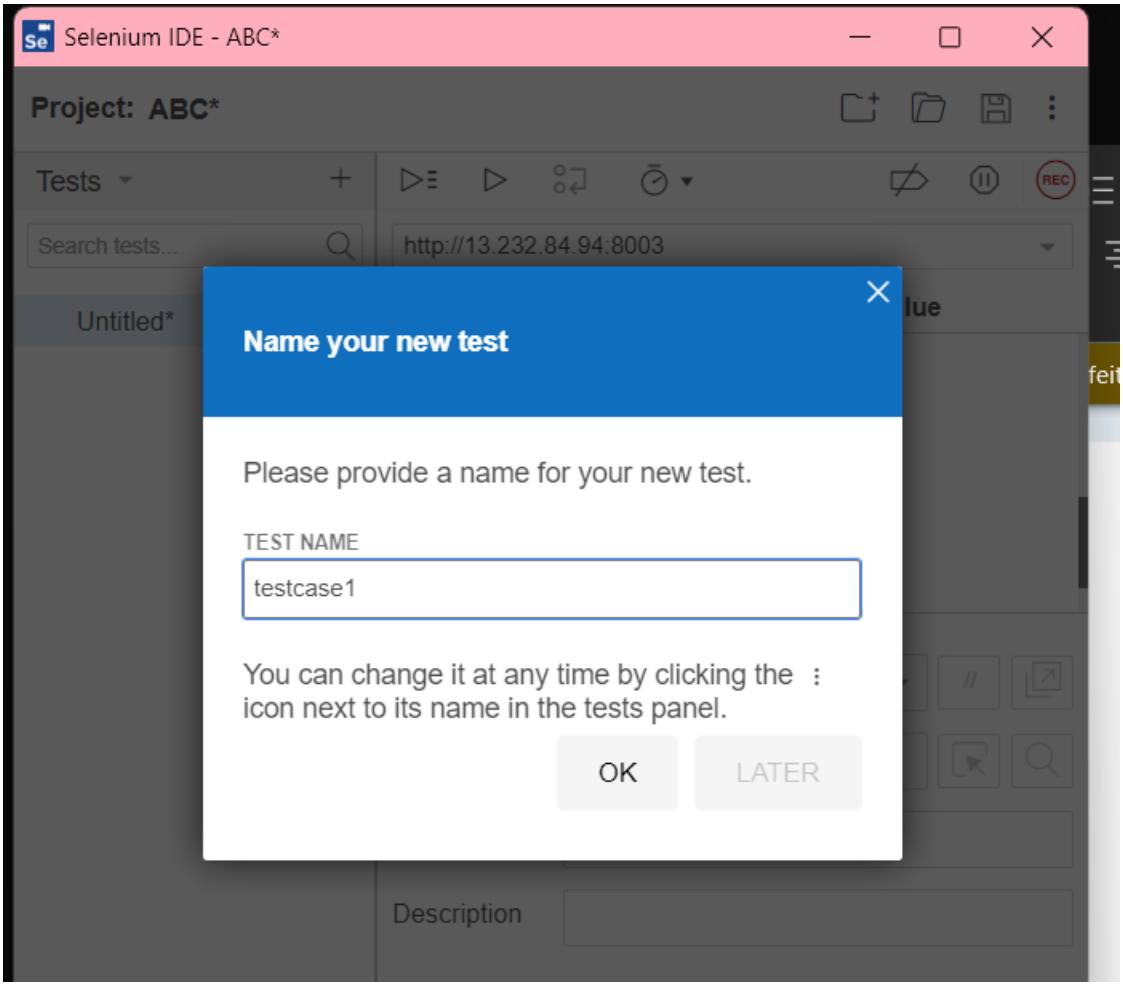


After that type a URL of website and click on start recording .

Now it will record all the steps you will perform in your website and after finishing the record click on stop button.


Selenium IDE - ABC*
Project: ABC*
Tests
Search tests...
Untitled*
Stop recording Ctrl+U
http://13.232.84.94:8003
Command
Target
Value
8 click linkText=About us
9 click linkText=Contact us
10 click linkText=Gallery
11 click id=2
Command
Target
Value
Description
Log
Reference

Now give the name to your test case and click on ok



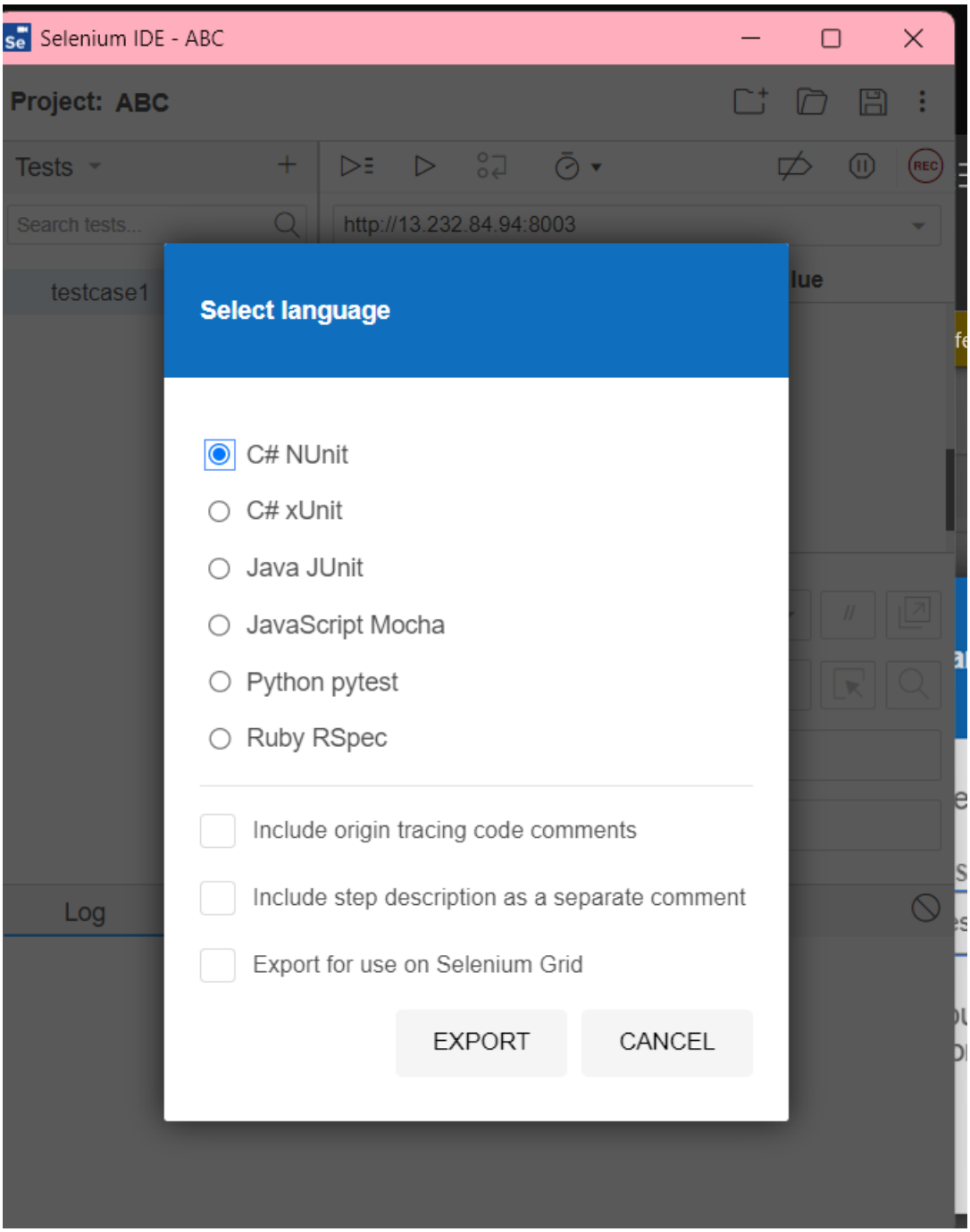
And the last if you want your test case in any language then right click on testcase and select EXPORT

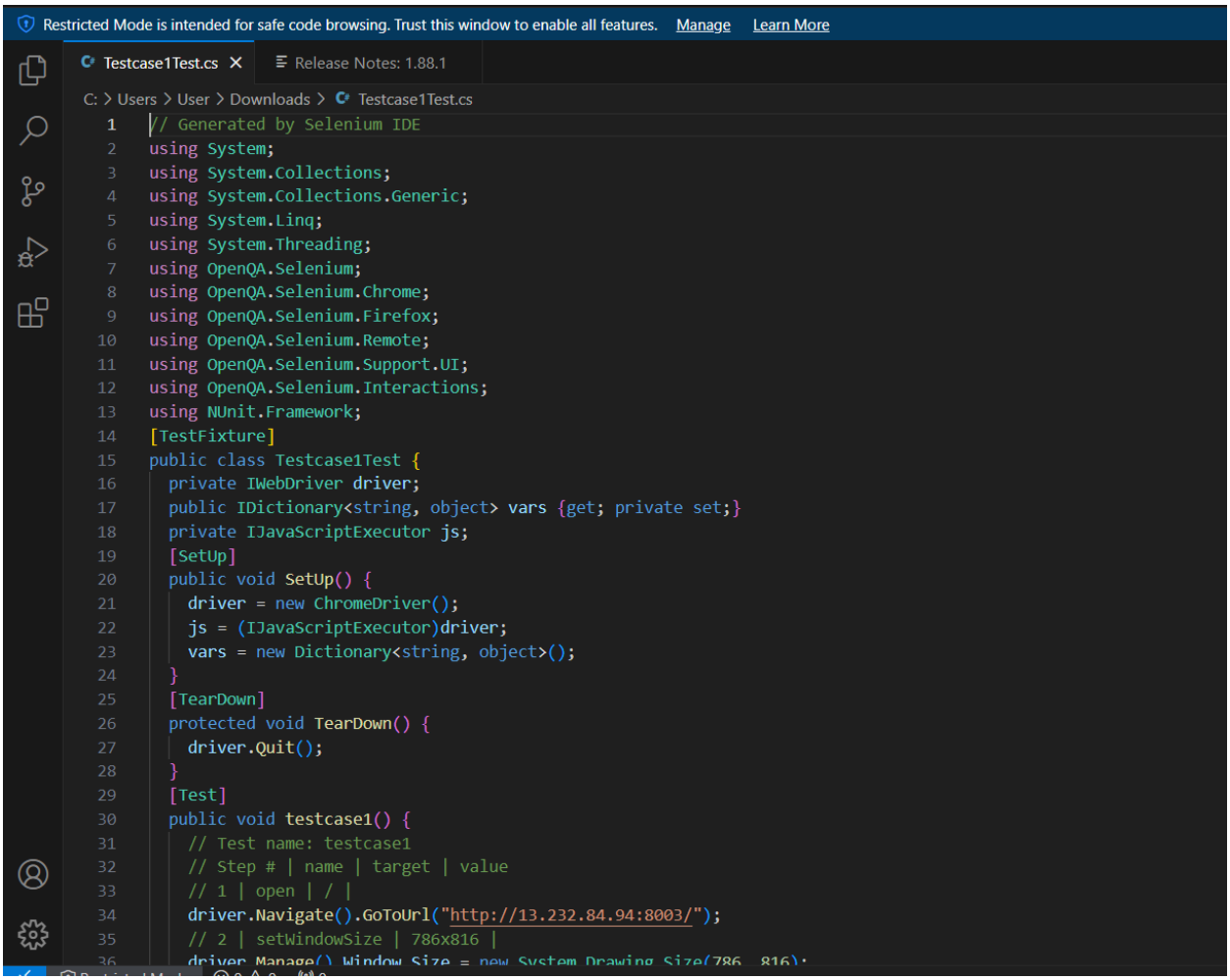
Option and after that select language in which you want your test case code.

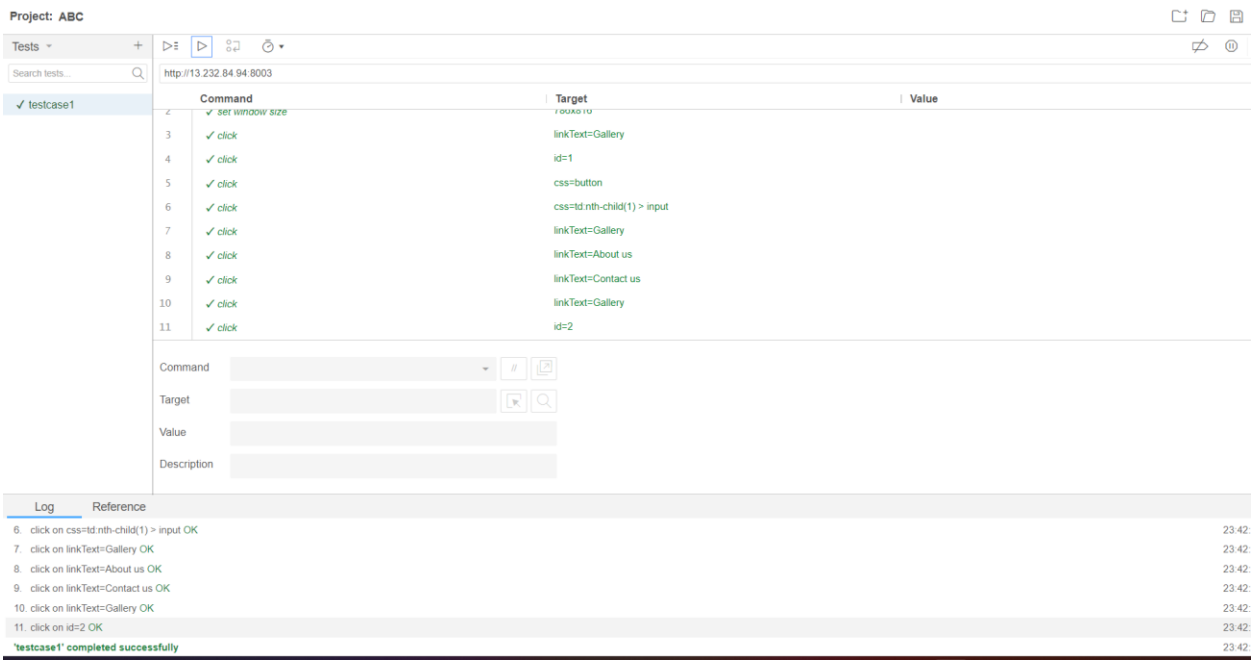
If you want to include comments in your code then check the boxes you want.

And at the end click on EXPORT.

Your file will be created successfully!!!


Selenium IDE - ABC
Project: ABC
Tests
Search tests...
http://13.232.84.94:8003
testcase1
Select language
C# NUnit
C# xUnit
Java JUnit
JavaScript Mocha
Python pytest
Ruby RSpec
Include origin tracing code comments
Include step description as a separate comment
Export for use on Selenium Grid
EXPORT
CANCEL
Log


Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More
Testcase1Test.cs
Release Notes: 1.88.1
C: > Users > User > Downloads > Testcase1Test.cs
// Generated by Selenium IDE
using System;
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using System.Threading;
using OpenQA.Selenium;
using OpenQA.Selenium.Chrome;
using OpenQA.Selenium.Firefox;
using OpenQA.Selenium.Remote;
using OpenQA.Selenium.Support.UI;
using OpenQA.Selenium.Interactions;
using NUnit.Framework;
[TestFixture]
public class Testcase1Test {
private IWebDriver driver;
public IDictionary<string, object> vars {get; private set;}
private IJavaScriptExecutor js;
[SetUp]
public void SetUp() {
driver = new ChromeDriver();
js = (IJavaScriptExecutor)driver;
vars = new Dictionary<string, object>();
}
[TearDown]
protected void TearDown() {
driver.Quit();
}
[Test]
public void testcase1() {
// Test name: testcase1
// Step # | name | target | value
// 1 | open | / |
driver.Navigate().GoToUrl("http://13.232.84.94:8003/");
// 2 | setWindowSize | 786x816 |


Project: ABC
Tests
testcase1
http://13.232.84.94:8003
Command
Target
Value
click
linkText=Gallery
id=1
css=button
css=td:nth-child(1) > input
linkText=Gallery
linkText=About us
linkText=Contact us
linkText=Gallery
id=2
Description
Log
Reference
6. click on css=td:nth-child(1) > input OK
7. click on linkText=Gallery OK
8. click on linkText=About us OK
9. click on linkText=Contact us OK
10. click on linkText=Gallery OK
11. click on id=2 OK
'testcase1' completed successfully