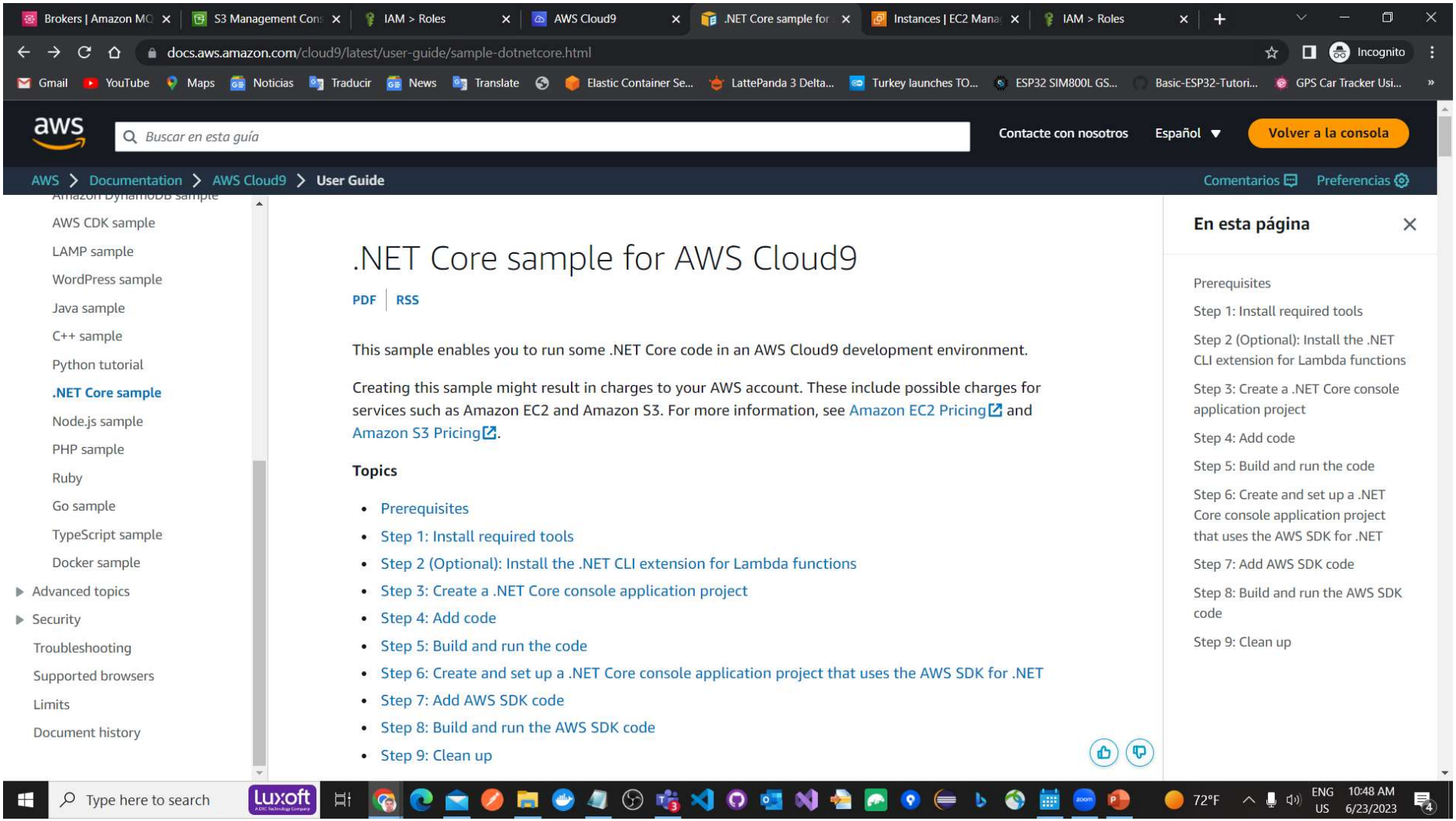
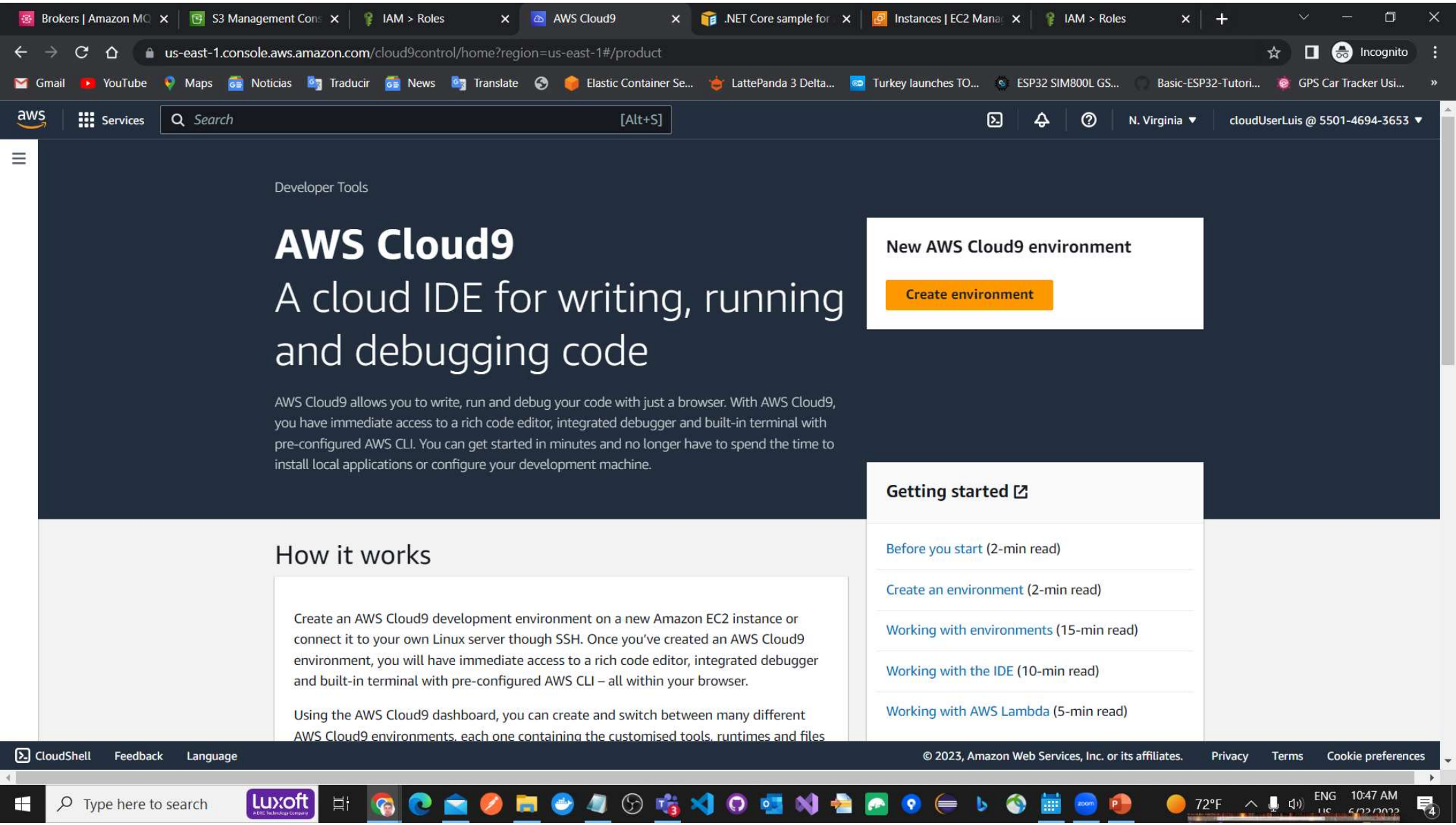
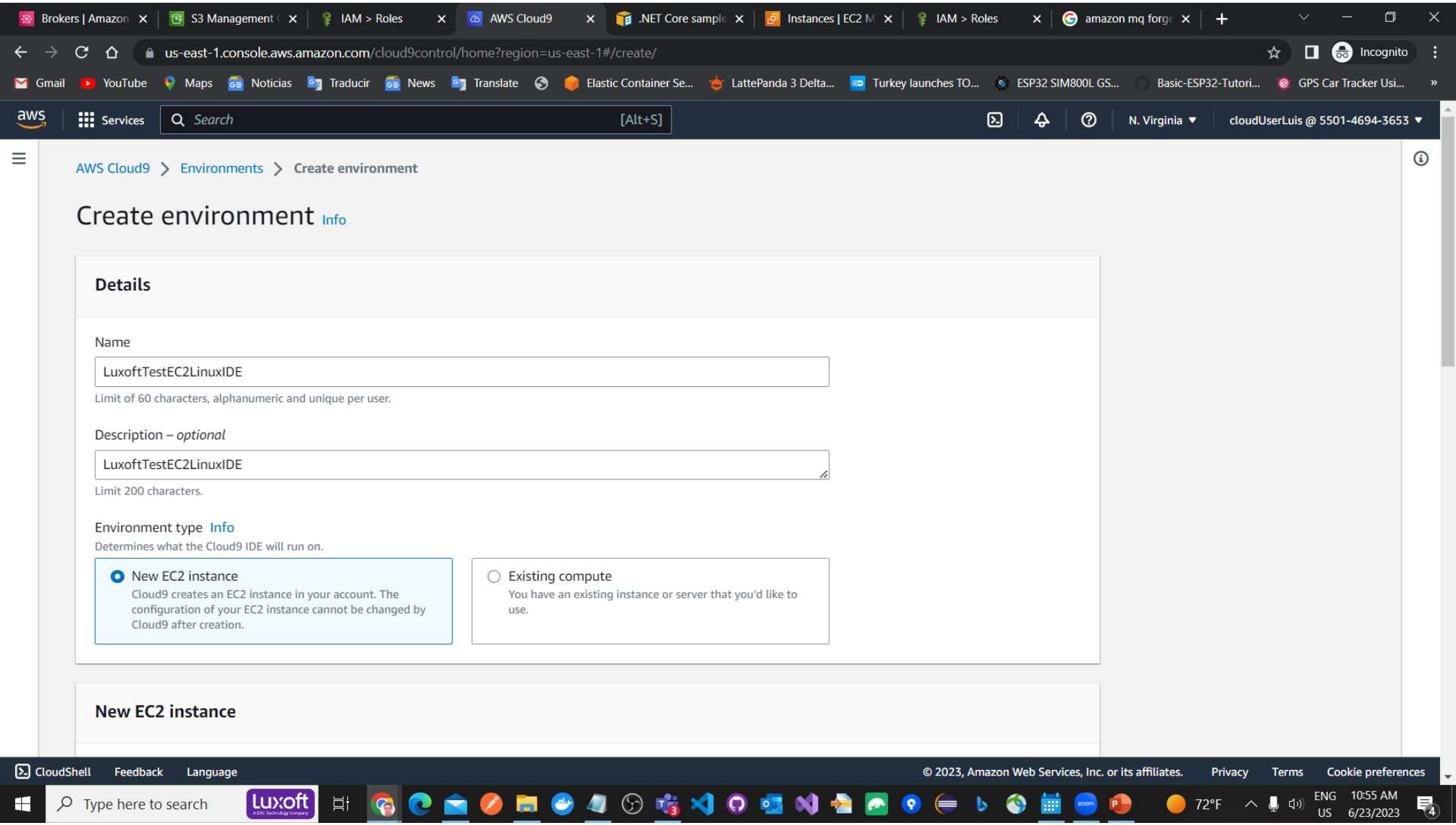
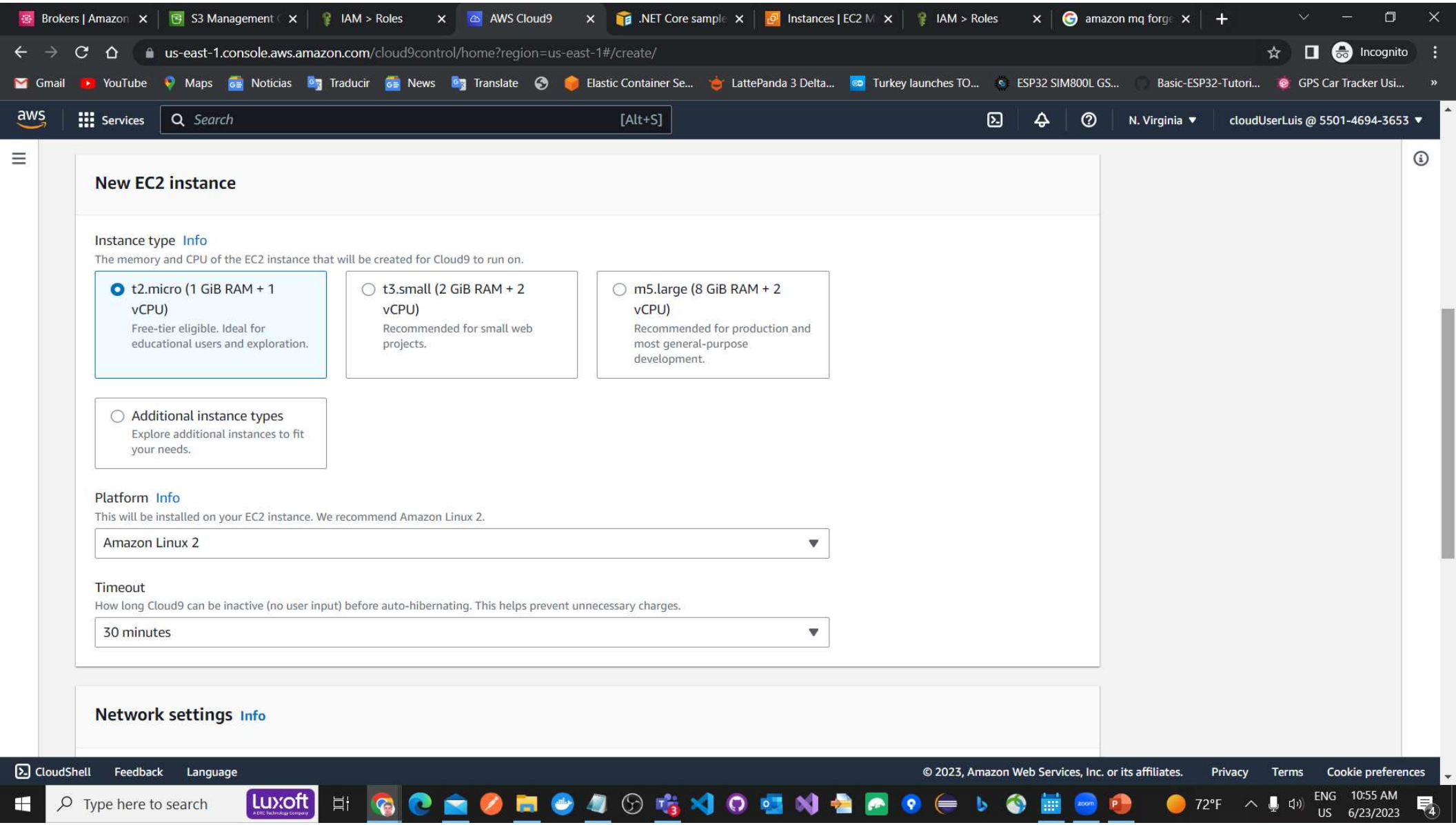
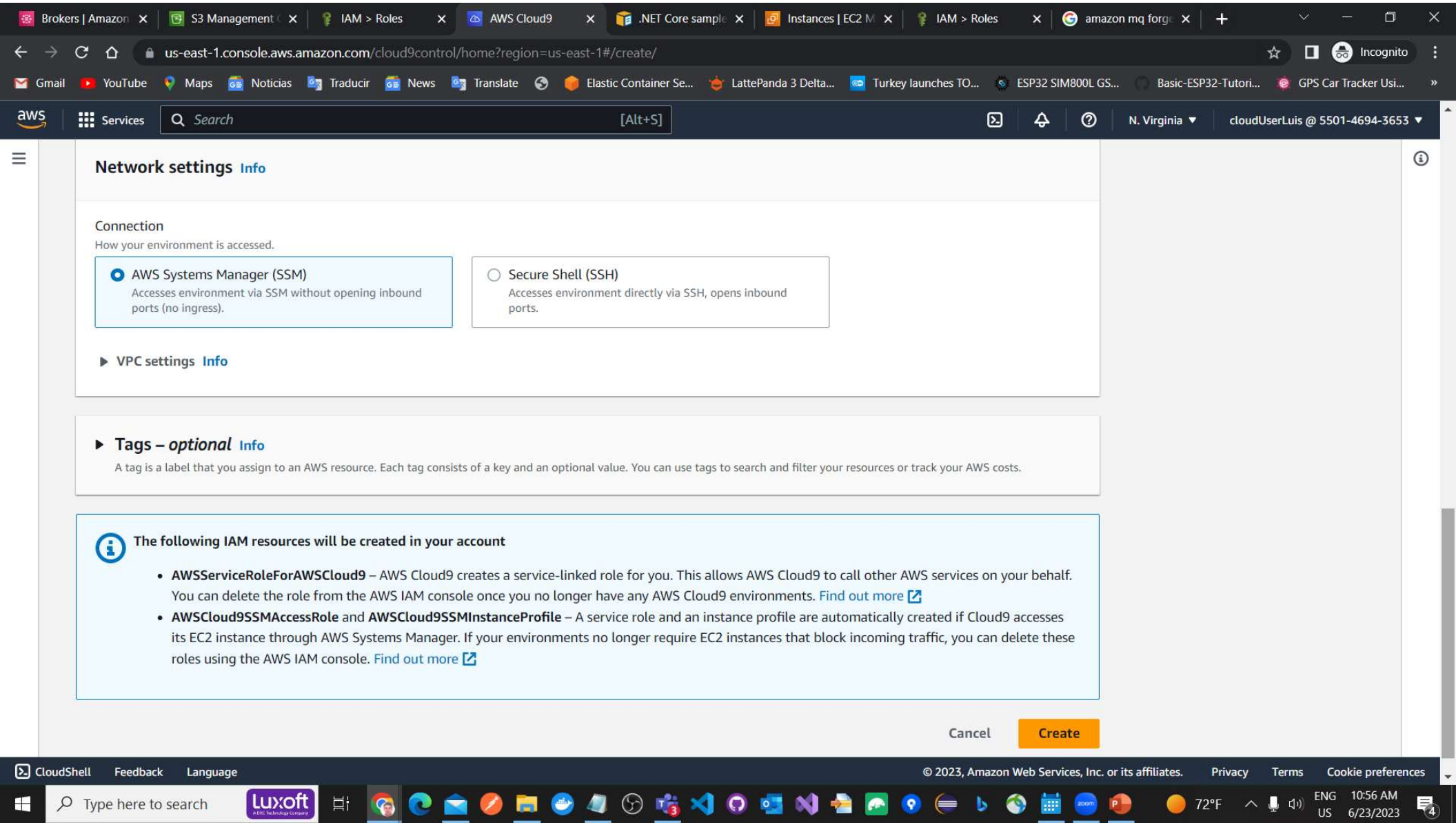


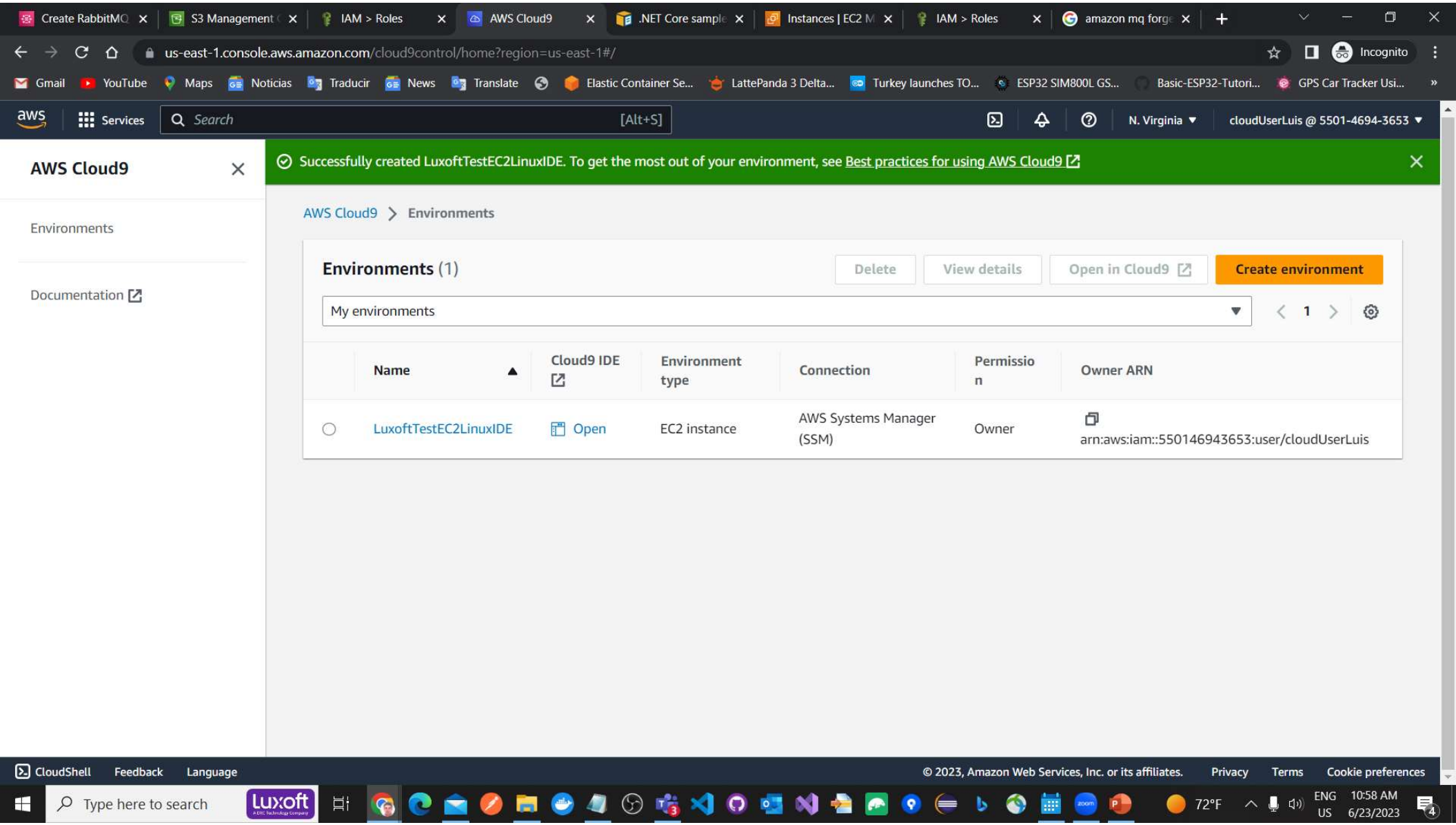
https://docs.aws.amazon.com/cloud9/latest/user-guide/sample-dotnetcore.html

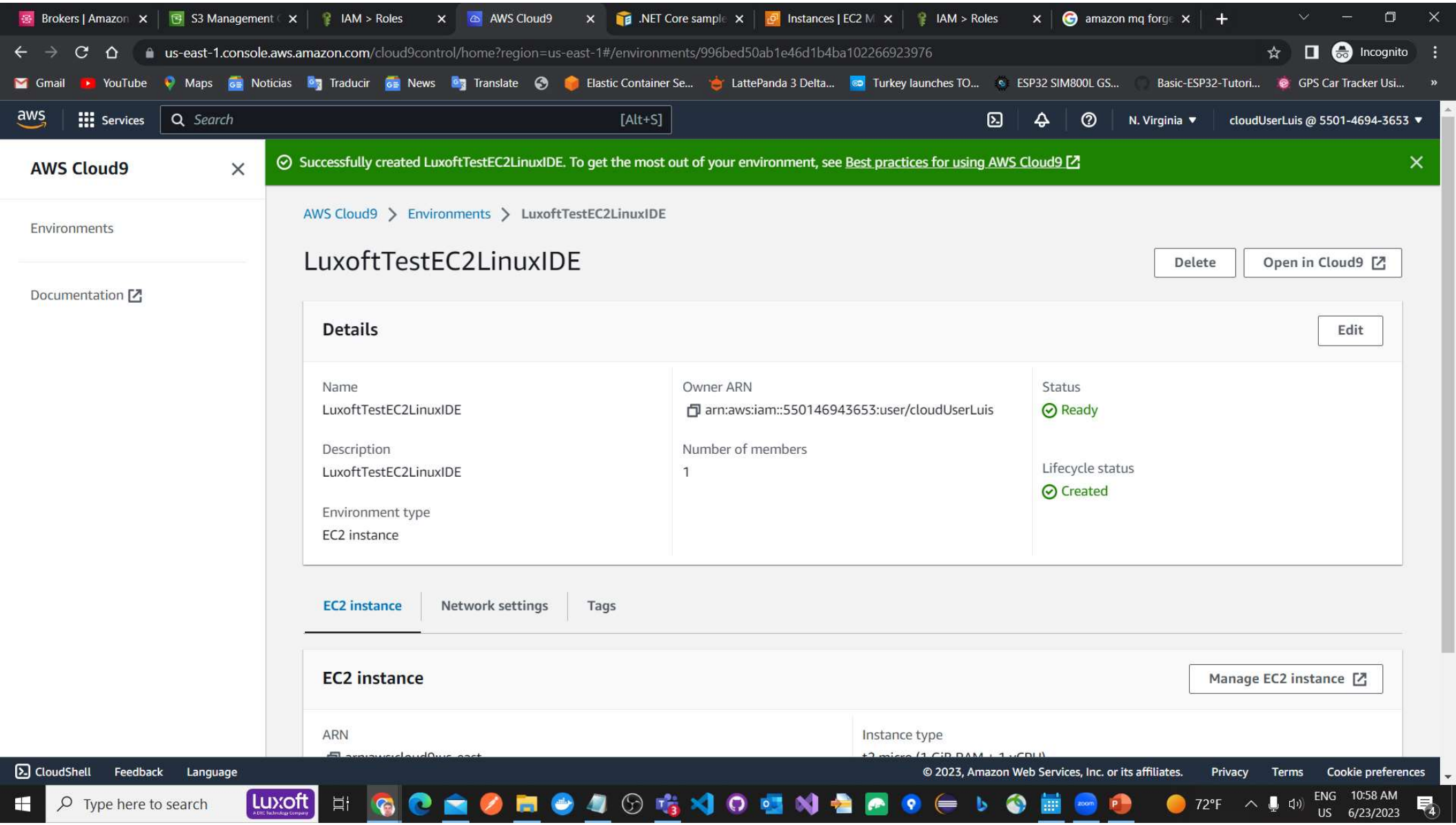

Brokers | Amazon MQ
S3 Management Cons
IAM > Roles
AWS Cloud9
.NET Core sample for
Instances | EC2 Mana
IAM > Roles
us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/product
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
Developer Tools
AWS Cloud9
A cloud IDE for writing, running and debugging code
AWS Cloud9 allows you to write, run and debug your code with just a browser. With AWS Cloud9, you have immediate access to a rich code editor, integrated debugger and built-in terminal with pre-configured AWS CLI. You can get started in minutes and no longer have to spend the time to install local applications or configure your development machine.
New AWS Cloud9 environment
Create environment
Getting started
Before you start (2-min read)
Create an environment (2-min read)
Working with environments (15-min read)
Working with the IDE (10-min read)
Working with AWS Lambda (5-min read)
How it works
Create an AWS Cloud9 development environment on a new Amazon EC2 instance or connect it to your own Linux server though SSH. Once you've created an AWS Cloud9 environment, you will have immediate access to a rich code editor, integrated debugger and built-in terminal with pre-configured AWS CLI – all within your browser.
Using the AWS Cloud9 dashboard, you can create and switch between many different AWS Cloud9 environments, each one containing the customised tools, runtimes and files
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
72°F
ENG
10:47 AM

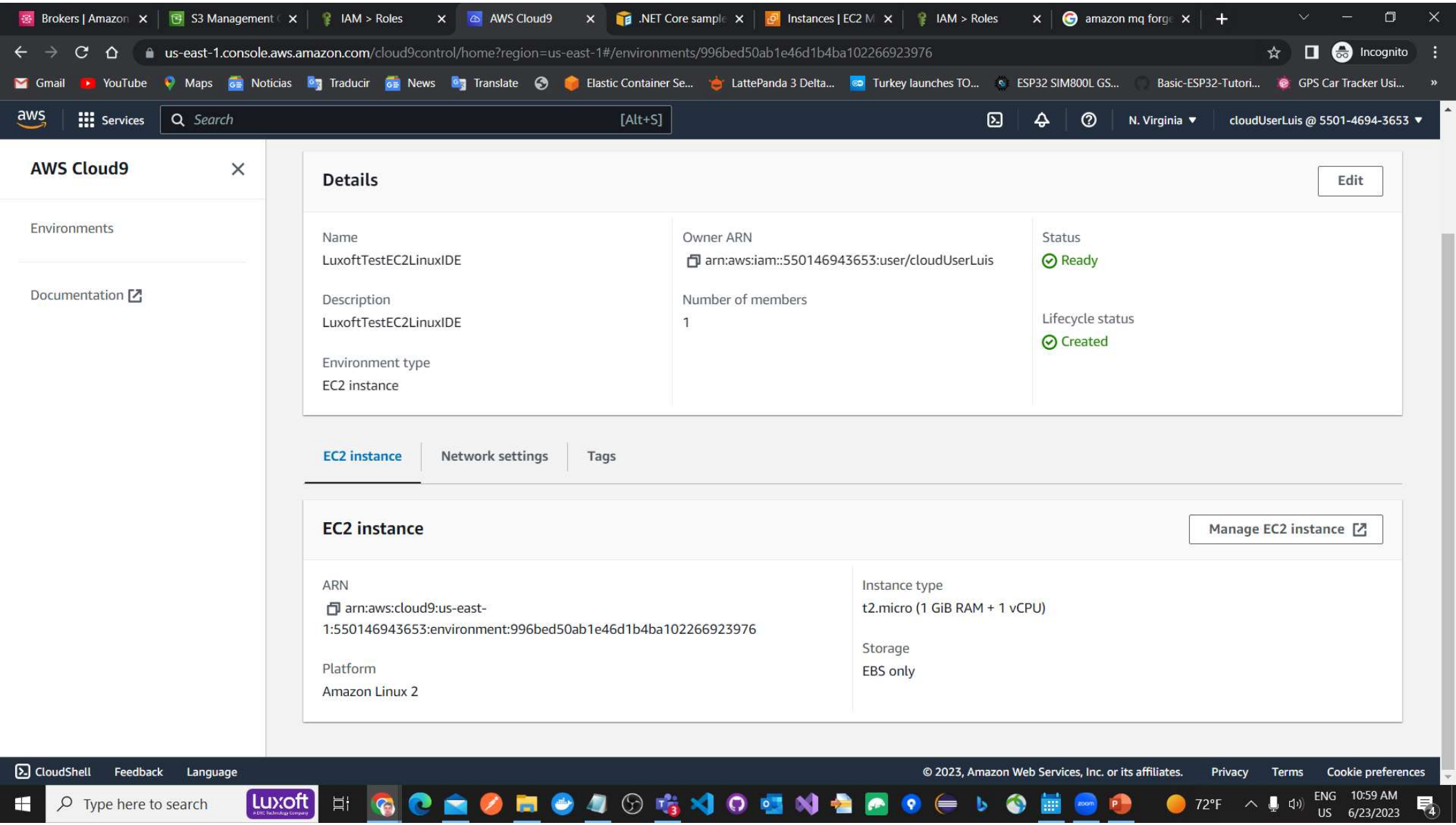

Brokers | Amazon
S3 Management
IAM > Roles
AWS Cloud9
.NET Core sample
Instances | EC2 M
IAM > Roles
amazon mq forge
us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/create/
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
AWS Cloud9 > Environments > Create environment
Create environment Info
Details
Name
LuxoftTestEC2LinuxIDE
Limit of 60 characters, alphanumeric and unique per user.
Description – optional
LuxoftTestEC2LinuxIDE
Limit 200 characters.
Environment type Info
Determines what the Cloud9 IDE will run on.
New EC2 instance
Cloud9 creates an EC2 instance in your account. The configuration of your EC2 instance cannot be changed by Cloud9 after creation.
Existing compute
You have an existing instance or server that you'd like to use.
New EC2 instance
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
72°F
ENG
US
10:55 AM
6/23/2023

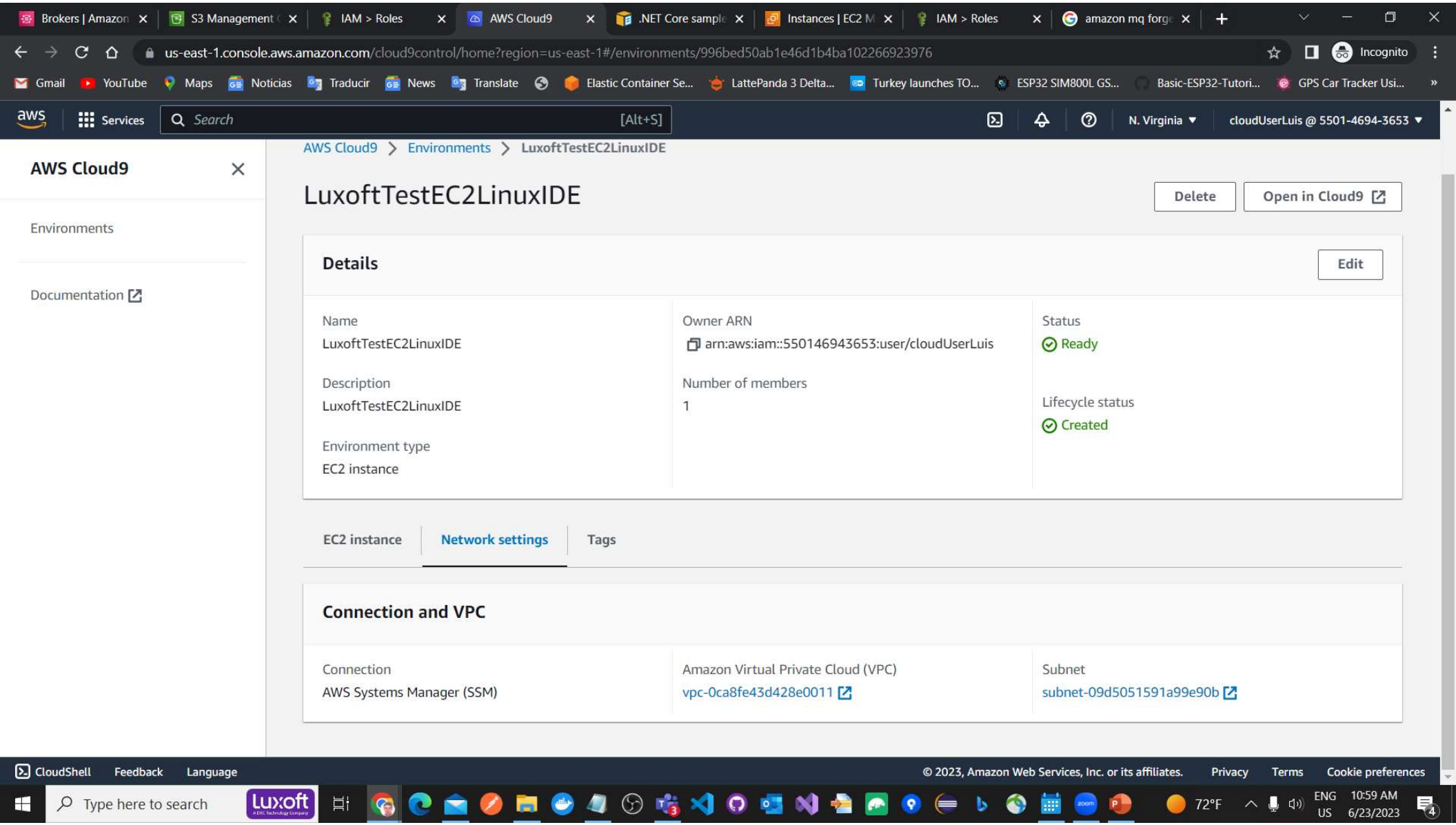

Brokers | Amazon
S3 Management
IAM > Roles
AWS Cloud9
.NET Core sample
Instances | EC2 M
IAM > Roles
amazon mq forge
us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/create/
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
New EC2 instance
Instance type Info
The memory and CPU of the EC2 instance that will be created for Cloud9 to run on.
t2.micro (1 GiB RAM + 1 vCPU)
Free-tier eligible. Ideal for educational users and exploration.
t3.small (2 GiB RAM + 2 vCPU)
Recommended for small web projects.
m5.large (8 GiB RAM + 2 vCPU)
Recommended for production and most general-purpose development.
Additional instance types
Explore additional instances to fit your needs.
Platform Info
This will be installed on your EC2 instance. We recommend Amazon Linux 2.
Amazon Linux 2
Timeout
How long Cloud9 can be inactive (no user input) before auto-hibernating. This helps prevent unnecessary charges.
30 minutes
Network settings Info
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
72°F
ENG
US
10:55 AM
6/23/2023

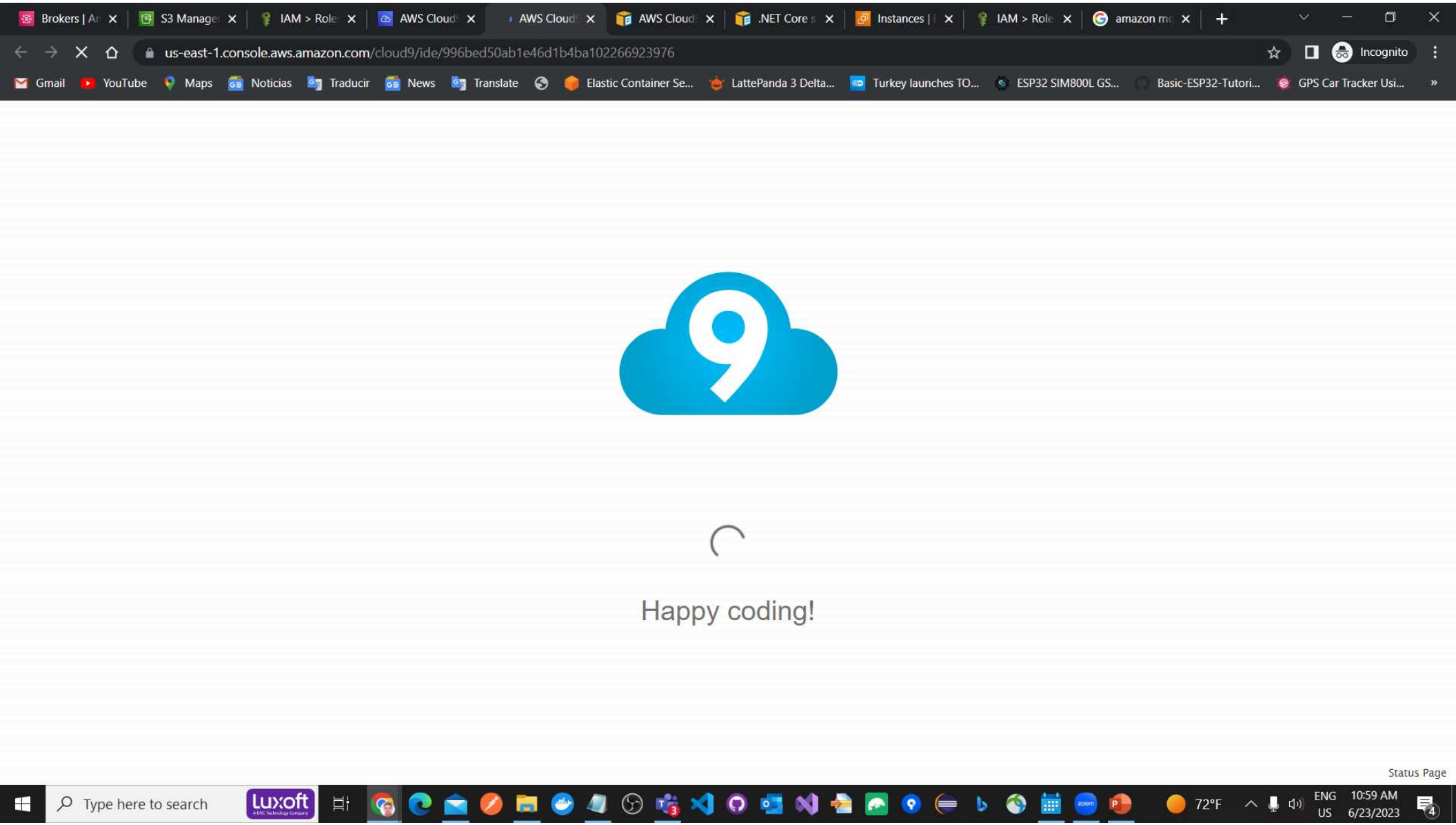

Brokers | Amazon
S3 Management
IAM > Roles
AWS Cloud9
.NET Core sample
Instances | EC2 M
IAM > Roles
amazon mq forge
us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/create/
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
Network settings Info
Connection
How your environment is accessed.
AWS Systems Manager (SSM)
Accesses environment via SSM without opening inbound ports (no ingress).
Secure Shell (SSH)
Accesses environment directly via SSH, opens inbound ports.
VPC settings Info
Tags – optional Info
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.
The following IAM resources will be created in your account
AWSServiceRoleForAWSCloud9 – AWS Cloud9 creates a service-linked role for you. This allows AWS Cloud9 to call other AWS services on your behalf. You can delete the role from the AWS IAM console once you no longer have any AWS Cloud9 environments. Find out more
AWSCloud9SSMAccessRole and AWSCloud9SSMInstanceProfile – A service role and an instance profile are automatically created if Cloud9 accesses its EC2 instance through AWS Systems Manager. If your environments no longer require EC2 instances that block incoming traffic, you can delete these roles using the AWS IAM console. Find out more
Cancel
Create
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
72°F
ENG
US
10:56 AM
6/23/2023

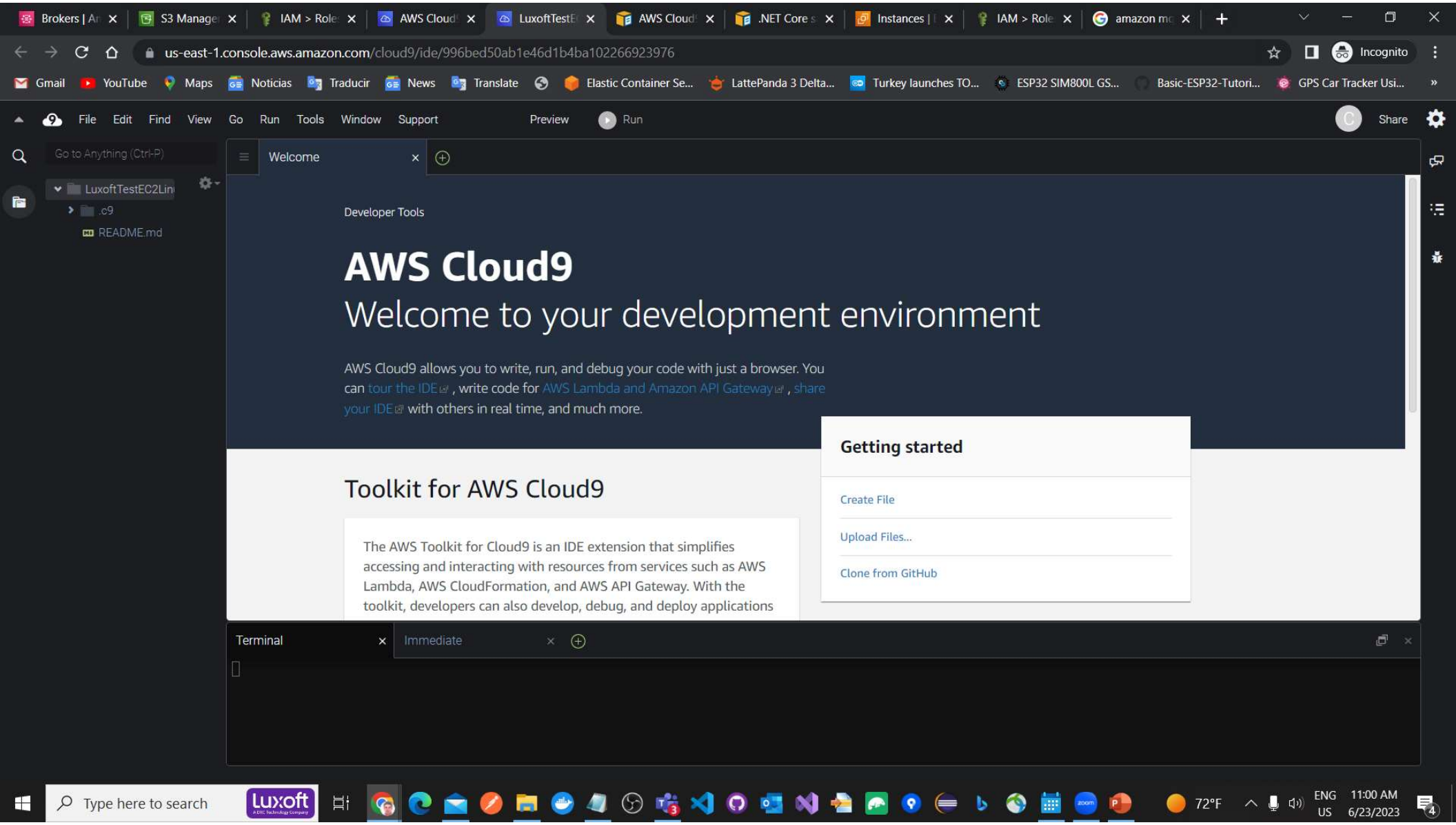

Create RabbitMQ
S3 Management
IAM > Roles
AWS Cloud9
.NET Core sample
Instances | EC2 M
IAM > Roles
amazon mq forge
us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
AWS Cloud9
Environments
Documentation
Successfully created LuxoftTestEC2LinuxIDE. To get the most out of your environment, see Best practices for using AWS Cloud9
AWS Cloud9 > Environments
Environments (1)
Delete
View details
Open in Cloud9
Create environment
My environments
1
Name
Cloud9 IDE
Environment type
Connection
Permission
Owner ARN
LuxoftTestEC2LinuxIDE
Open
EC2 instance
AWS Systems Manager (SSM)
Owner
arn:aws:iam::550146943653:user/cloudUserLuis
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
72°F
ENG
US
10:58 AM
6/23/2023

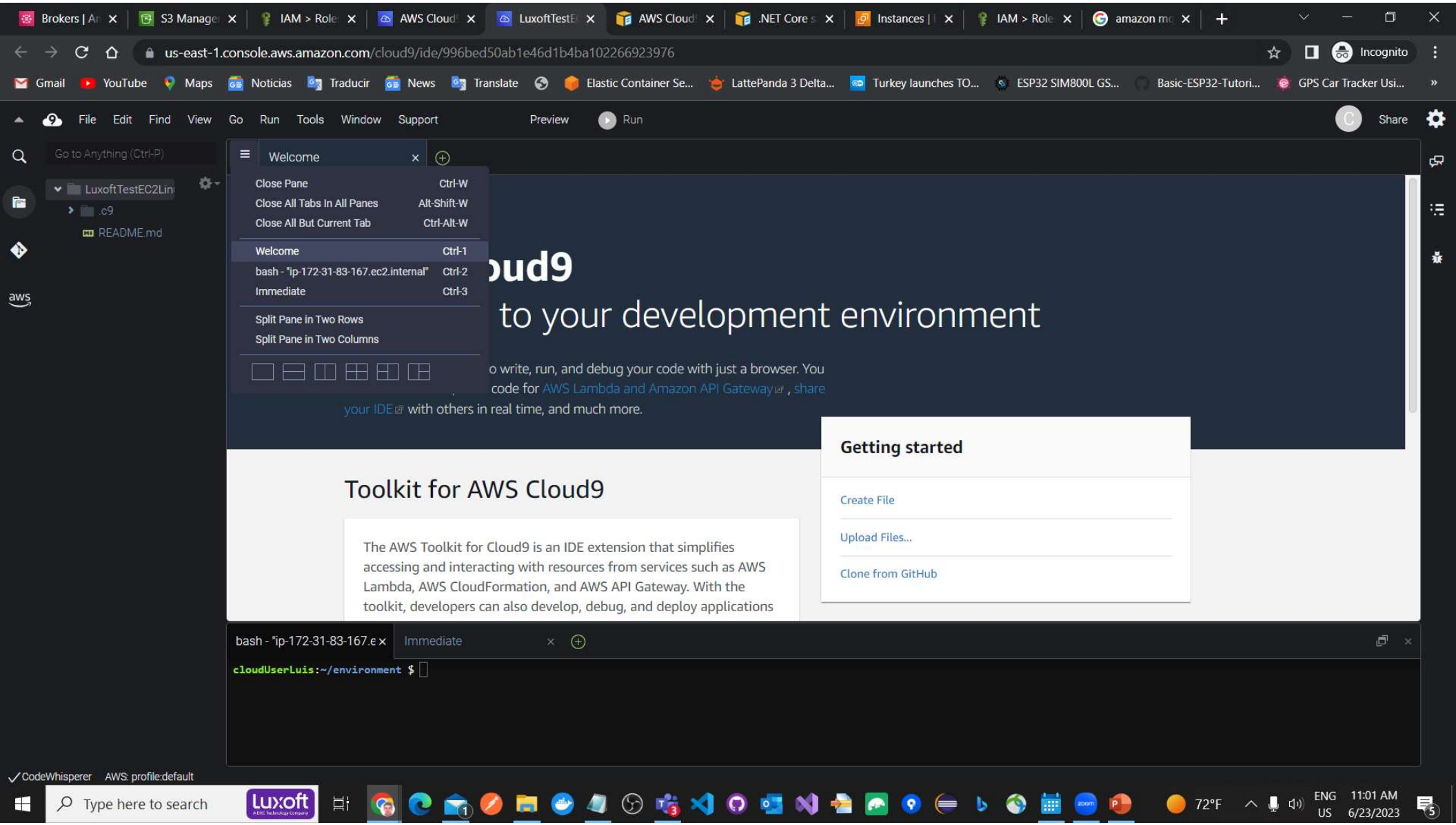

Brokers | Amazon
S3 Management
IAM > Roles
AWS Cloud9
.NET Core sample
Instances | EC2 M
IAM > Roles
amazon mq forge
us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/environments/996bed50ab1e46d1b4ba102266923976
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
AWS Cloud9
Environments
Documentation
Successfully created LuxoftTestEC2LinuxIDE. To get the most out of your environment, see Best practices for using AWS Cloud9
AWS Cloud9 > Environments > LuxoftTestEC2LinuxIDE
LuxoftTestEC2LinuxIDE
Delete
Open in Cloud9
Details
Edit
Name
LuxoftTestEC2LinuxIDE
Description
LuxoftTestEC2LinuxIDE
Environment type
EC2 instance
Owner ARN
arn:aws:iam::550146943653:user/cloudUserLuis
Number of members
1
Status
Ready
Lifecycle status
Created
EC2 instance
Network settings
Tags
EC2 instance
Manage EC2 instance
ARN
Instance type
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
72°F
ENG
US
10:58 AM
6/23/2023

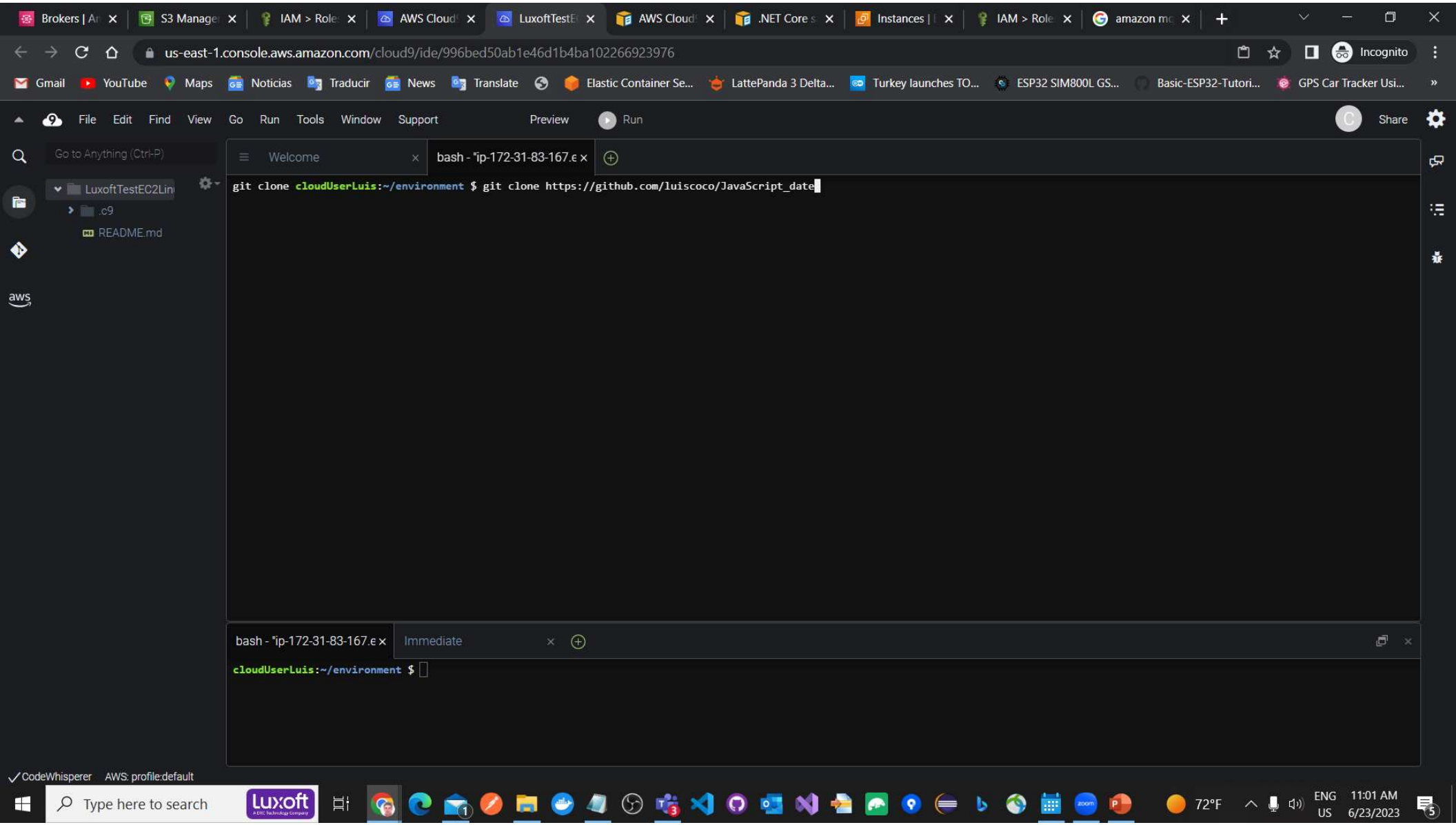

Brokers | Amazon
S3 Management
IAM > Roles
AWS Cloud9
.NET Core sample
Instances | EC2 M
IAM > Roles
amazon mq forge
us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/environments/996bed50ab1e46d1b4ba102266923976
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
AWS Cloud9
Environments
Documentation
Details
Edit
Name
LuxoftTestEC2LinuxIDE
Description
LuxoftTestEC2LinuxIDE
Environment type
EC2 instance
Owner ARN
arn:aws:iam::550146943653:user/cloudUserLuis
Number of members
1
Status
Ready
Lifecycle status
Created
EC2 instance
Network settings
Tags
EC2 instance
Manage EC2 instance
ARN
arn:aws:cloud9:us-east-1:550146943653:environment:996bed50ab1e46d1b4ba102266923976
Platform
Amazon Linux 2
Instance type
t2.micro (1 GiB RAM + 1 vCPU)
Storage
EBS only
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
72°F
ENG
US
10:59 AM
6/23/2023

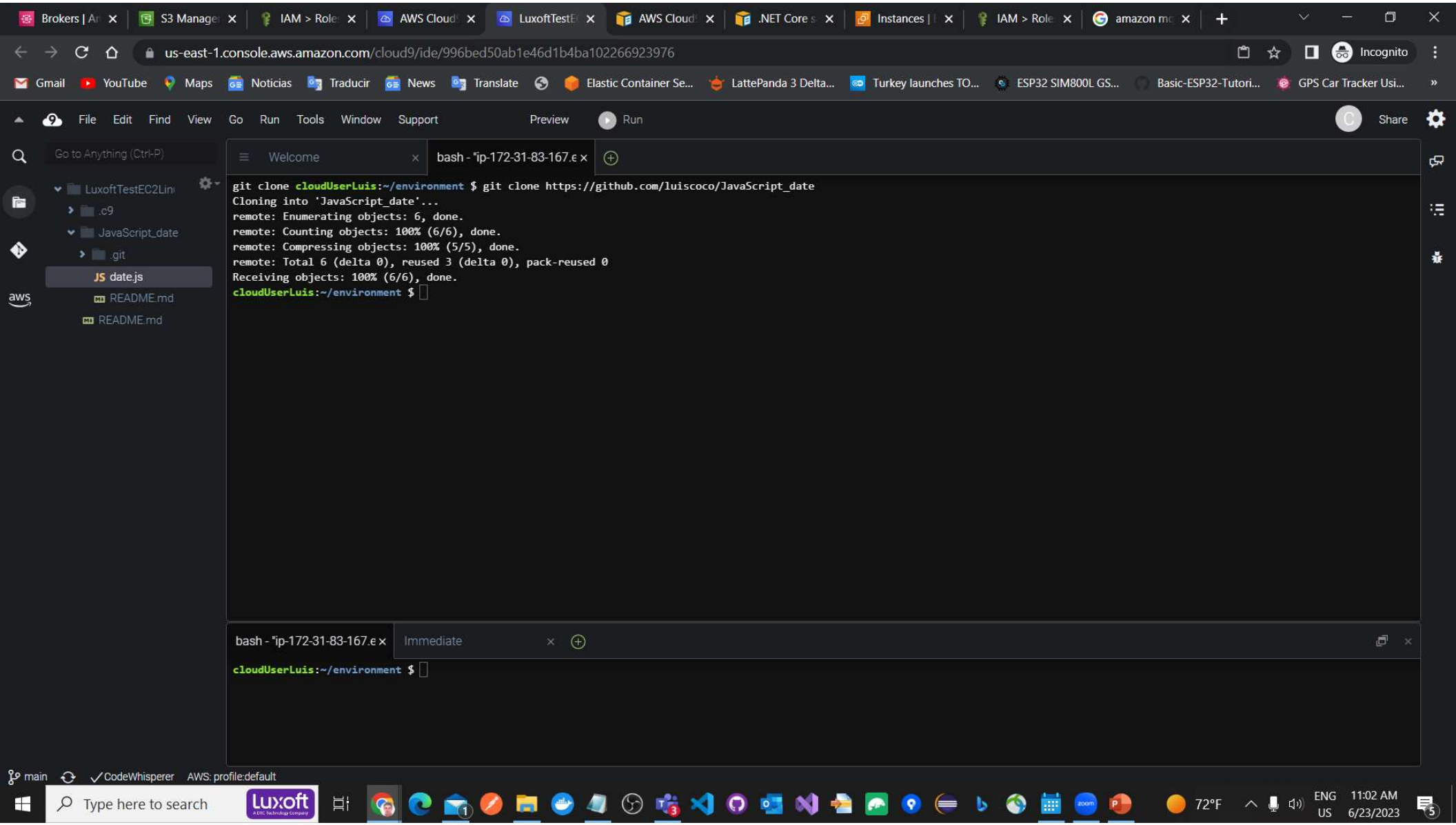

Brokers | Amazon
S3 Management
IAM > Roles
AWS Cloud9
.NET Core sample
Instances | EC2 M
IAM > Roles
amazon mq forge
us-east-1.console.aws.amazon.com/cloud9control/home?region=us-east-1#/environments/996bed50ab1e46d1b4ba102266923976
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
AWS Cloud9
Environments
Documentation
AWS Cloud9 > Environments > LuxoftTestEC2LinuxIDE
LuxoftTestEC2LinuxIDE
Delete
Open in Cloud9
Details
Edit
Name
LuxoftTestEC2LinuxIDE
Description
LuxoftTestEC2LinuxIDE
Environment type
EC2 instance
Owner ARN
arn:aws:iam::550146943653:user/cloudUserLuis
Number of members
1
Status
Ready
Lifecycle status
Created
EC2 instance
Network settings
Tags
Connection and VPC
Connection
AWS Systems Manager (SSM)
Amazon Virtual Private Cloud (VPC)
vpc-0ca8fe43d428e0011
Subnet
subnet-09d5051591a99e90b
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
72°F
ENG
US
10:59 AM
6/23/2023

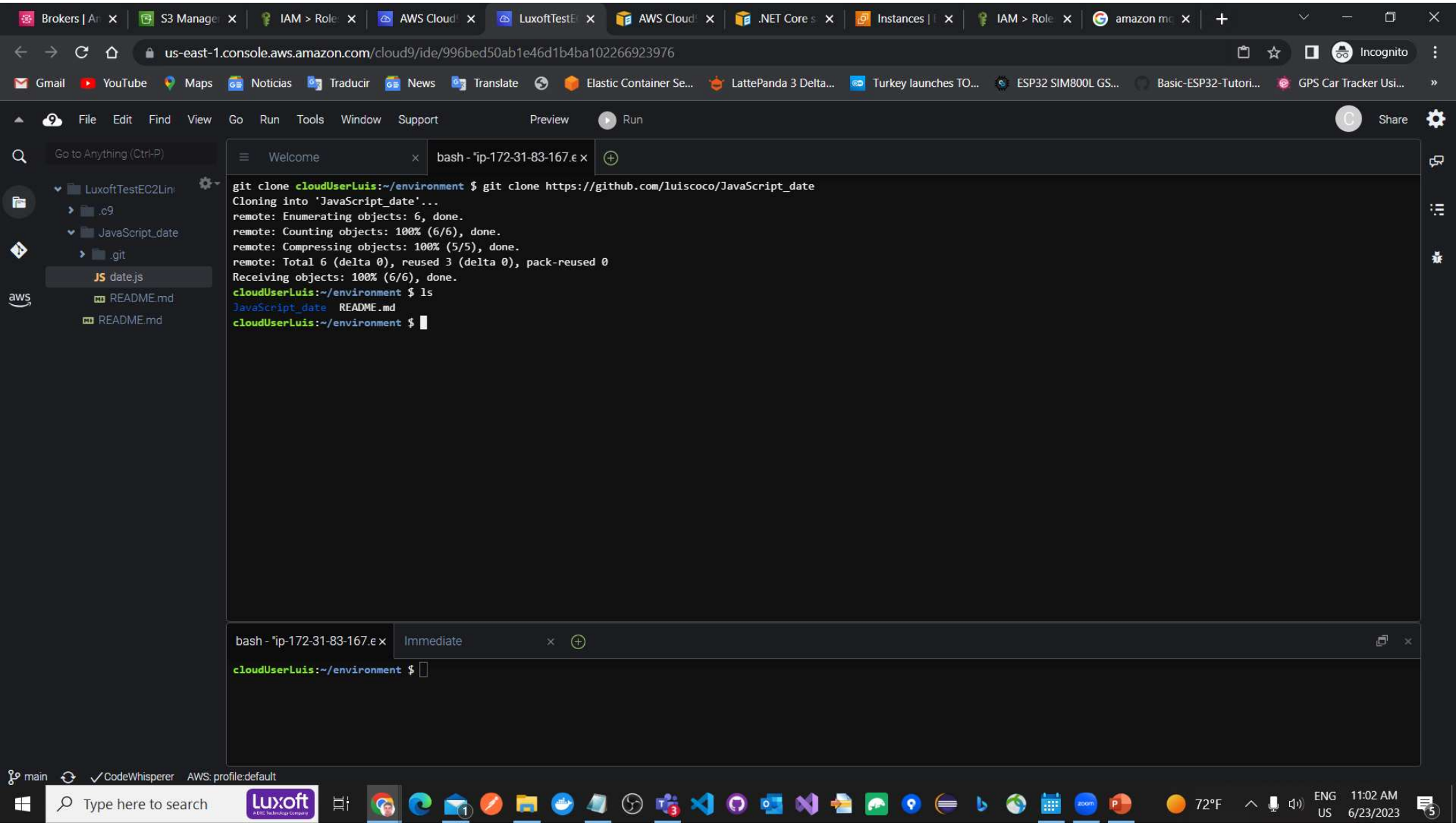

us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
Happy coding!
Status Page
Type here to search
72°F
ENG
US
10:59 AM
6/23/2023

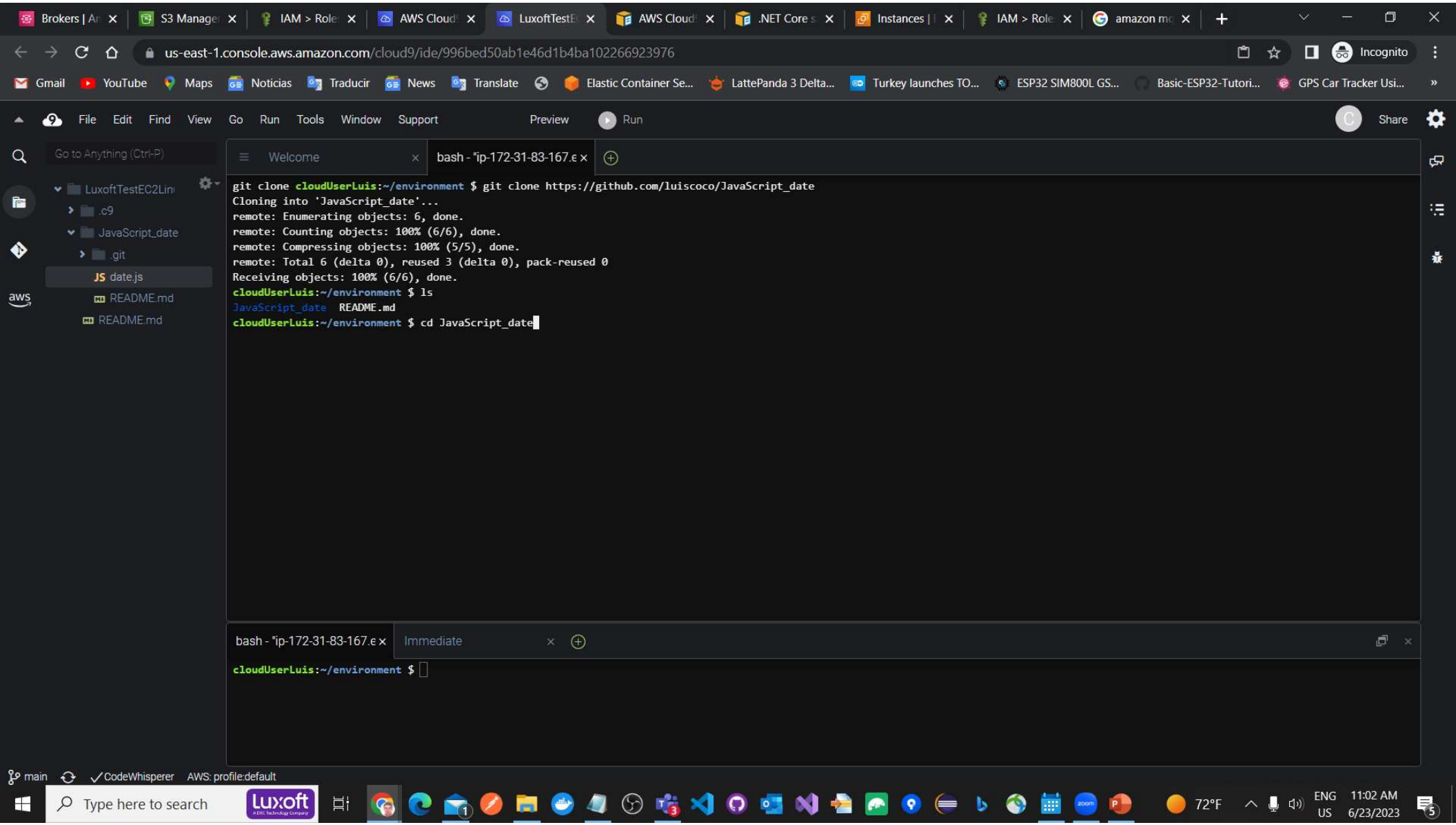

us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
LuxoftTestEC2Lin
.c9
README.md
Welcome
Developer Tools
AWS Cloud9
Welcome to your development environment
AWS Cloud9 allows you to write, run, and debug your code with just a browser. You can tour the IDE, write code for AWS Lambda and Amazon API Gateway, share your IDE with others in real time, and much more.
Getting started
Create File
Upload Files...
Clone from GitHub
Toolkit for AWS Cloud9
The AWS Toolkit for Cloud9 is an IDE extension that simplifies accessing and interacting with resources from services such as AWS Lambda, AWS CloudFormation, and AWS API Gateway. With the toolkit, developers can also develop, debug, and deploy applications
Terminal
Immediate
Type here to search
72°F
ENG US
11:00 AM 6/23/2023

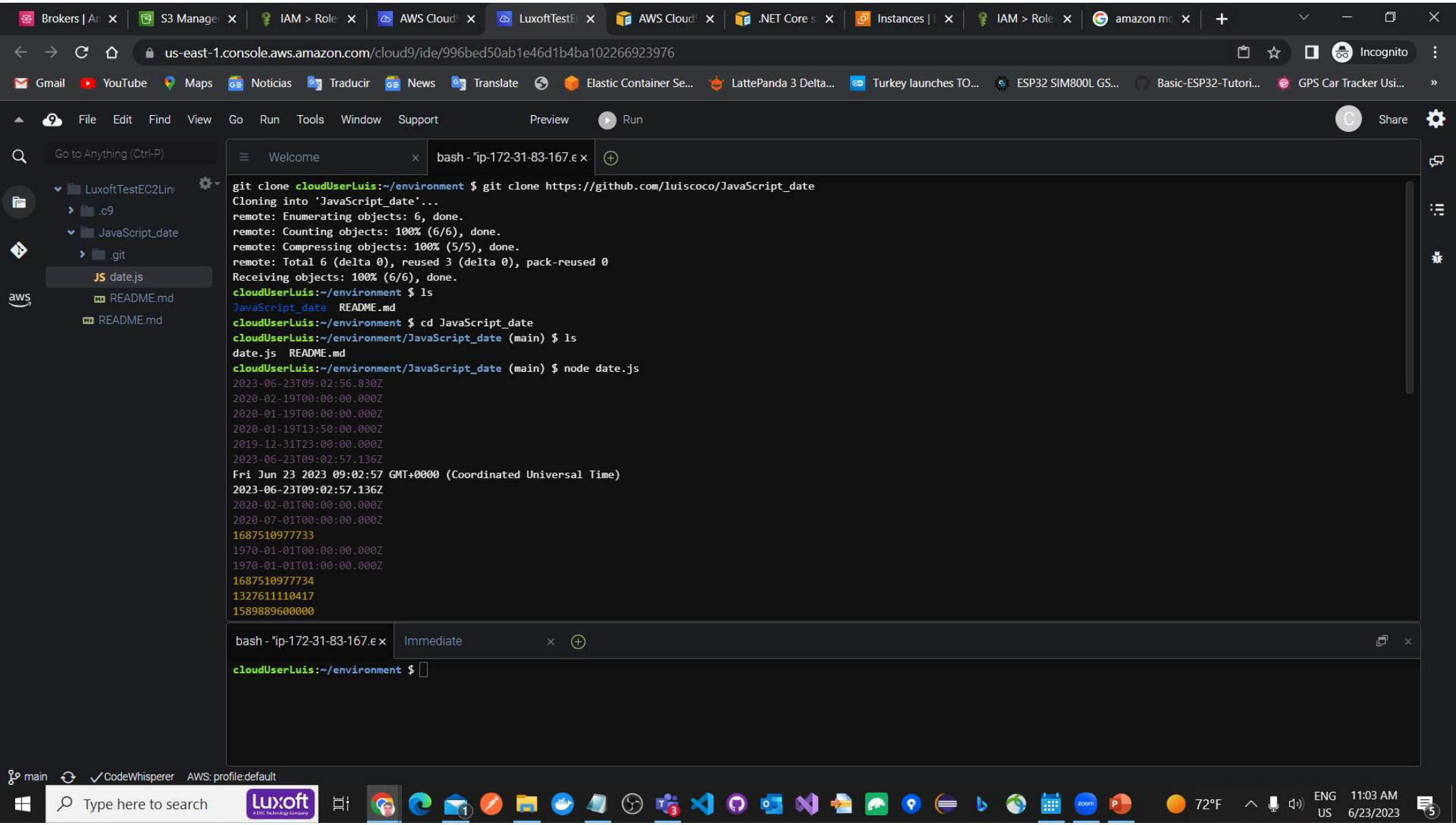

us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
LuxoftTestEC2Lin
.c9
README.md
Welcome
Close Pane Ctrl-W
Close All Tabs In All Panes Alt-Shift-W
Close All But Current Tab Ctrl-Alt-W
Welcome Ctrl-1
bash - "ip-172-31-83-167.ec2.internal" Ctrl-2
Immediate Ctrl-3
Split Pane in Two Rows
Split Pane in Two Columns
to your development environment
write, run, and debug your code with just a browser. You
code for AWS Lambda and Amazon API Gateway , share
your IDE with others in real time, and much more.
Getting started
Create File
Upload Files...
Clone from GitHub
Toolkit for AWS Cloud9
The AWS Toolkit for Cloud9 is an IDE extension that simplifies accessing and interacting with resources from services such as AWS Lambda, AWS CloudFormation, and AWS API Gateway. With the toolkit, developers can also develop, debug, and deploy applications
bash - "ip-172-31-83-167.e
Immediate
cloudUserLuis:~/environment $
CodeWhisperer AWS: profile:default
Type here to search
72°F
ENG US
11:01 AM 6/23/2023

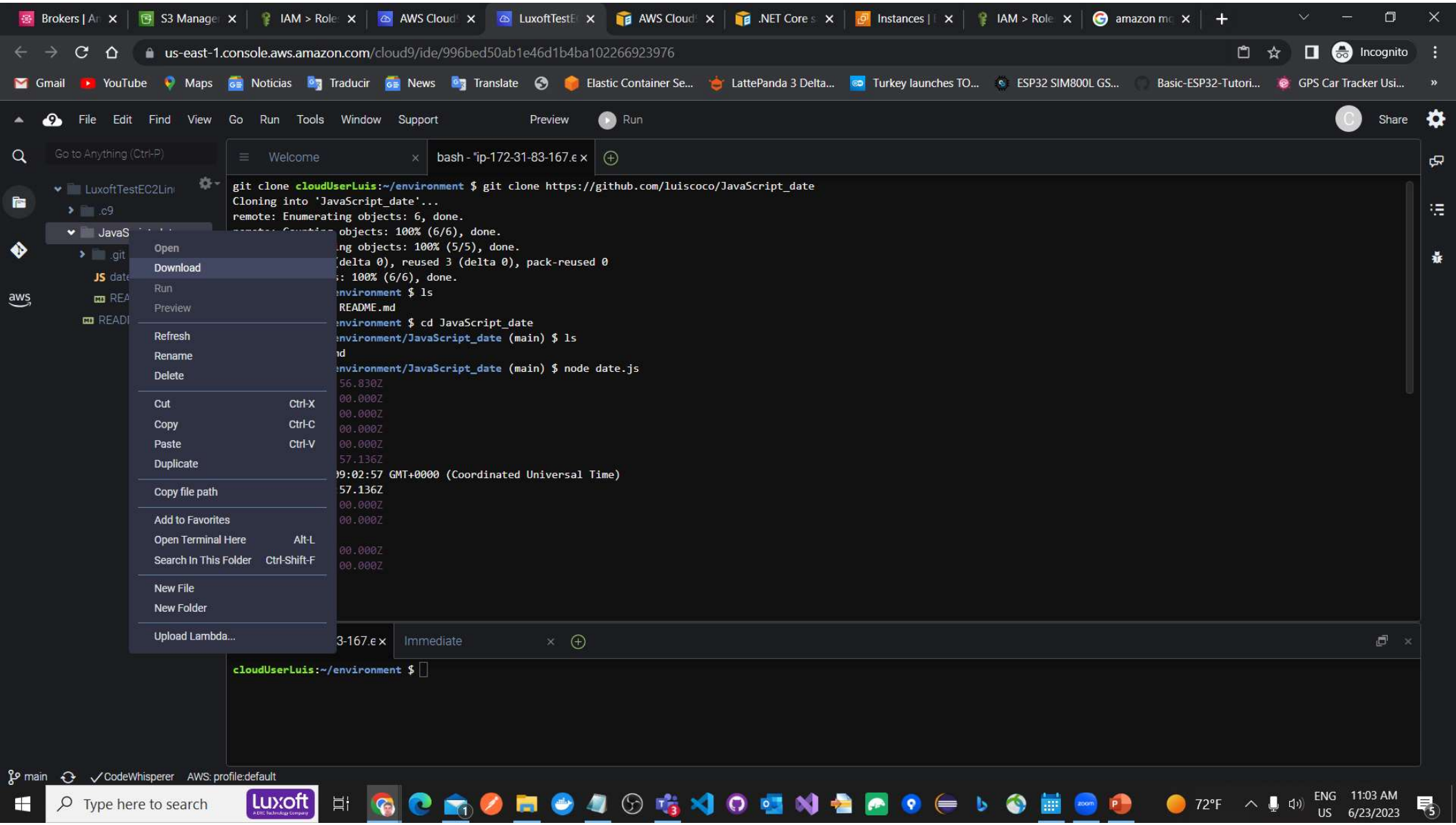

us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
LuxoftTestEC2Lin
.c9
README.md
Welcome
bash - "ip-172-31-83-167.e
git clone cloudUserLuis:~/environment $ git clone https://github.com/luiscoco/JavaScript_date
bash - "ip-172-31-83-167.e
Immediate
cloudUserLuis:~/environment $
CodeWhisperer AWS: profile:default
Type here to search
72°F
ENG US
11:01 AM 6/23/2023


us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Welcome
bash - "ip-172-31-83-167.e
LuxoftTestEC2Lin
.c9
JavaScript_date
.git
date.js
README.md
README.md
git clone cloudUserLuis:~/environment $ git clone https://github.com/luiscoco/JavaScript_date
Cloning into 'JavaScript_date'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 3 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.
cloudUserLuis:~/environment $
bash - "ip-172-31-83-167.e
Immediate
cloudUserLuis:~/environment $
main
CodeWhisperer
AWS: profile:default
Type here to search
72°F
ENG US
11:02 AM 6/23/2023


us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
LuxoftTestEC2Lin
.c9
JavaScript_date
.git
date.js
README.md
README.md
Welcome
bash - "ip-172-31-83-167.e
git clone cloudUserLuis:~/environment $ git clone https://github.com/luiscoco/JavaScript_date
Cloning into 'JavaScript_date'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 3 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.
cloudUserLuis:~/environment $ ls
JavaScript_date README.md
cloudUserLuis:~/environment $
bash - "ip-172-31-83-167.e
Immediate
cloudUserLuis:~/environment $
main
CodeWhisperer
AWS: profile:default
Type here to search
72°F
ENG US
11:02 AM 6/23/2023


us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
LuxoftTestEC2Lin
.c9
JavaScript_date
.git
date.js
README.md
README.md
Welcome
bash - "ip-172-31-83-167.e
git clone cloudUserLuis:~/environment $ git clone https://github.com/luiscoco/JavaScript_date
Cloning into 'JavaScript_date'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 3 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.
cloudUserLuis:~/environment $ ls
JavaScript_date README.md
cloudUserLuis:~/environment $ cd JavaScript_date
bash - "ip-172-31-83-167.e
Immediate
cloudUserLuis:~/environment $
main
CodeWhisperer
AWS: profile:default
Type here to search
72°F
ENG US
11:02 AM 6/23/2023


us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
LuxoftTestEC2Lin
.c9
JavaScript_date
.git
date.js
README.md
README.md
Welcome
bash - "ip-172-31-83-167.e
git clone cloudUserLuis:~/environment $ git clone https://github.com/luiscoco/JavaScript_date
Cloning into 'JavaScript_date'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 6 (delta 0), reused 3 (delta 0), pack-reused 0
Receiving objects: 100% (6/6), done.
cloudUserLuis:~/environment $ ls
JavaScript_date README.md
cloudUserLuis:~/environment $ cd JavaScript_date
cloudUserLuis:~/environment/JavaScript_date (main) $ ls
date.js README.md
cloudUserLuis:~/environment/JavaScript_date (main) $ node date.js
2023-06-23T09:02:56.830Z
2020-02-19T00:00:00.000Z
2020-01-19T00:00:00.000Z
2020-01-19T13:50:00.000Z
2019-12-31T23:00:00.000Z
2023-06-23T09:02:57.136Z
Fri Jun 23 2023 09:02:57 GMT+0000 (Coordinated Universal Time)
2023-06-23T09:02:57.136Z
2020-02-01T00:00:00.000Z
2020-07-01T00:00:00.000Z
1687510977733
1970-01-01T00:00:00.000Z
1970-01-01T01:00:00.000Z
1687510977734
1327611110417
1589889600000
bash - "ip-172-31-83-167.e
Immediate
cloudUserLuis:~/environment $
main
CodeWhisperer
AWS: profile:default
Type here to search
72°F
ENG US
11:03 AM 6/23/2023


us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
Welcome
bash - "ip-172-31-83-167.e
git clone cloudUserLuis:~/environment $ git clone https://github.com/luiscoco/JavaScript_date
Cloning into 'JavaScript_date'...
remote: Enumerating objects: 6, done.
objects: 100% (6/6), done.
ng objects: 100% (5/5), done.
(delta 0), reused 3 (delta 0), pack-reused 0
: 100% (6/6), done.
environment $ ls
README.md
environment $ cd JavaScript_date
environment/JavaScript_date (main) $ ls
environment/JavaScript_date (main) $ node date.js
56.830Z
00.000Z
00.000Z
00.000Z
00.000Z
57.136Z
09:02:57 GMT+0000 (Coordinated Universal Time)
57.136Z
00.000Z
00.000Z
00.000Z
00.000Z
Open
Download
Run
Preview
Refresh
Rename
Delete
Cut Ctrl-X
Copy Ctrl-C
Paste Ctrl-V
Duplicate
Copy file path
Add to Favorites
Open Terminal Here Alt-L
Search In This Folder Ctrl-Shift-F
New File
New Folder
Upload Lambda...
Immediate
cloudUserLuis:~/environment $
main
CodeWhisperer
AWS: profile:default
Type here to search
72°F
ENG US
11:03 AM 6/23/2023

Step 1: Install required tools

In this step, you install the .NET Core SDK into your environment, which is required to run this sample.

1. Confirm whether the latest version of the .NET Core SDK is already installed in your environment. To do this, in a terminal session in the AWS Cloud9 IDE, run the .NET Core command line interface (CLI) with the `--version` option.

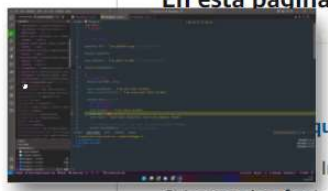
```
dotnet --version
```

   If the .NET Command Line Tools version is displayed, and the version is 2.0 or greater, skip ahead to Step 3: Create a .NET Core console application project. If the version is less than 2.0, or if an error such as `bash: dotnet: command not found` is displayed, continue on to install the .NET Core SDK.

2. For Amazon Linux, in a terminal session in the AWS Cloud9 IDE, run the following commands to help ensure the latest security updates and bug fixes are installed, and to install a `libunwind` package that the .NET Core SDK needs. (To start a new terminal session, on the menu bar, choose **Window, New Terminal**.)

```
sudo yum -y update
sudo yum -y install libunwind
```

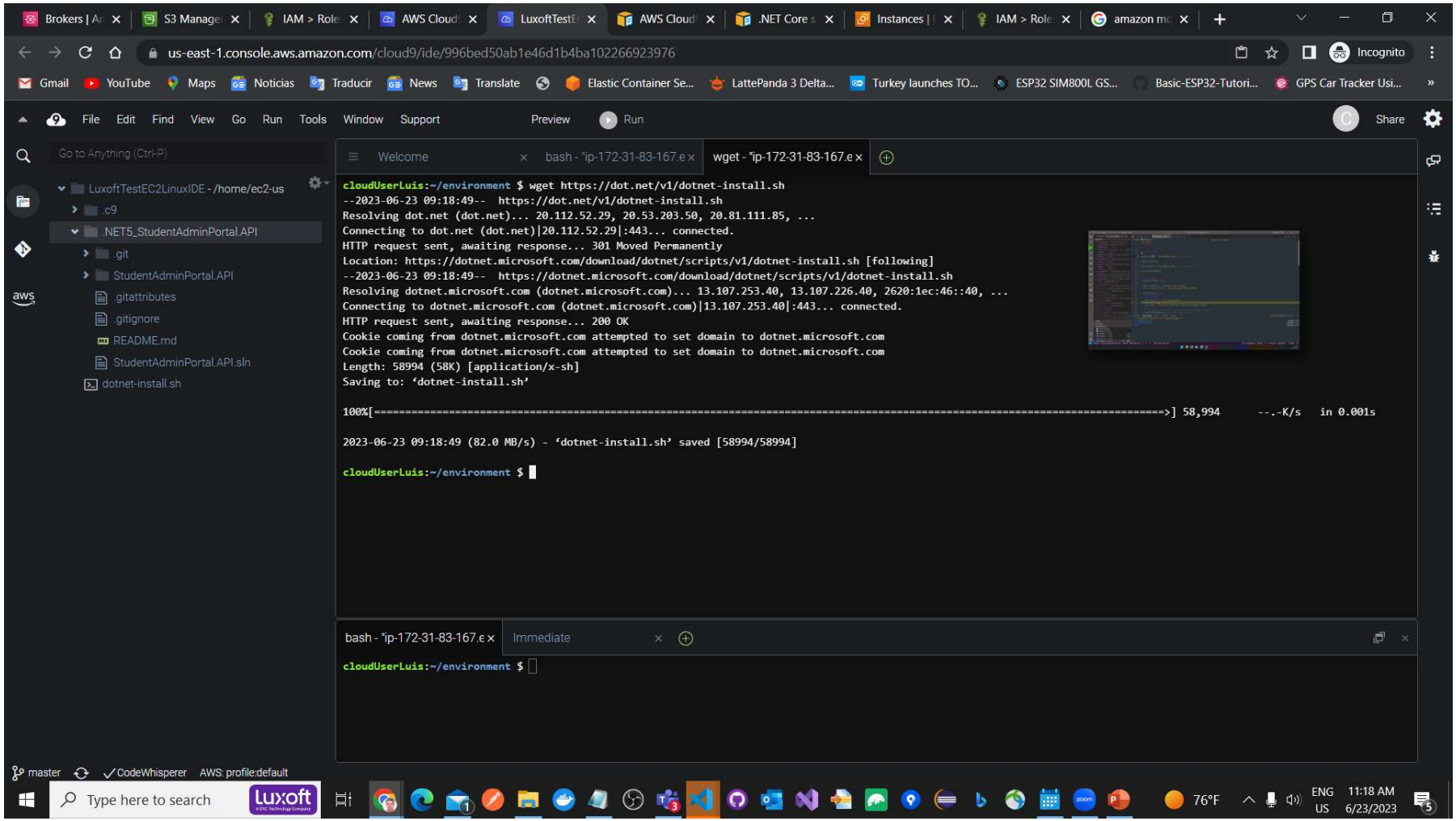
   For Ubuntu Server, in a terminal session in the AWS Cloud9 IDE, run the following command to help ensure the latest security updates and bug fixes are installed. (To start a new terminal session, on the



En esta página

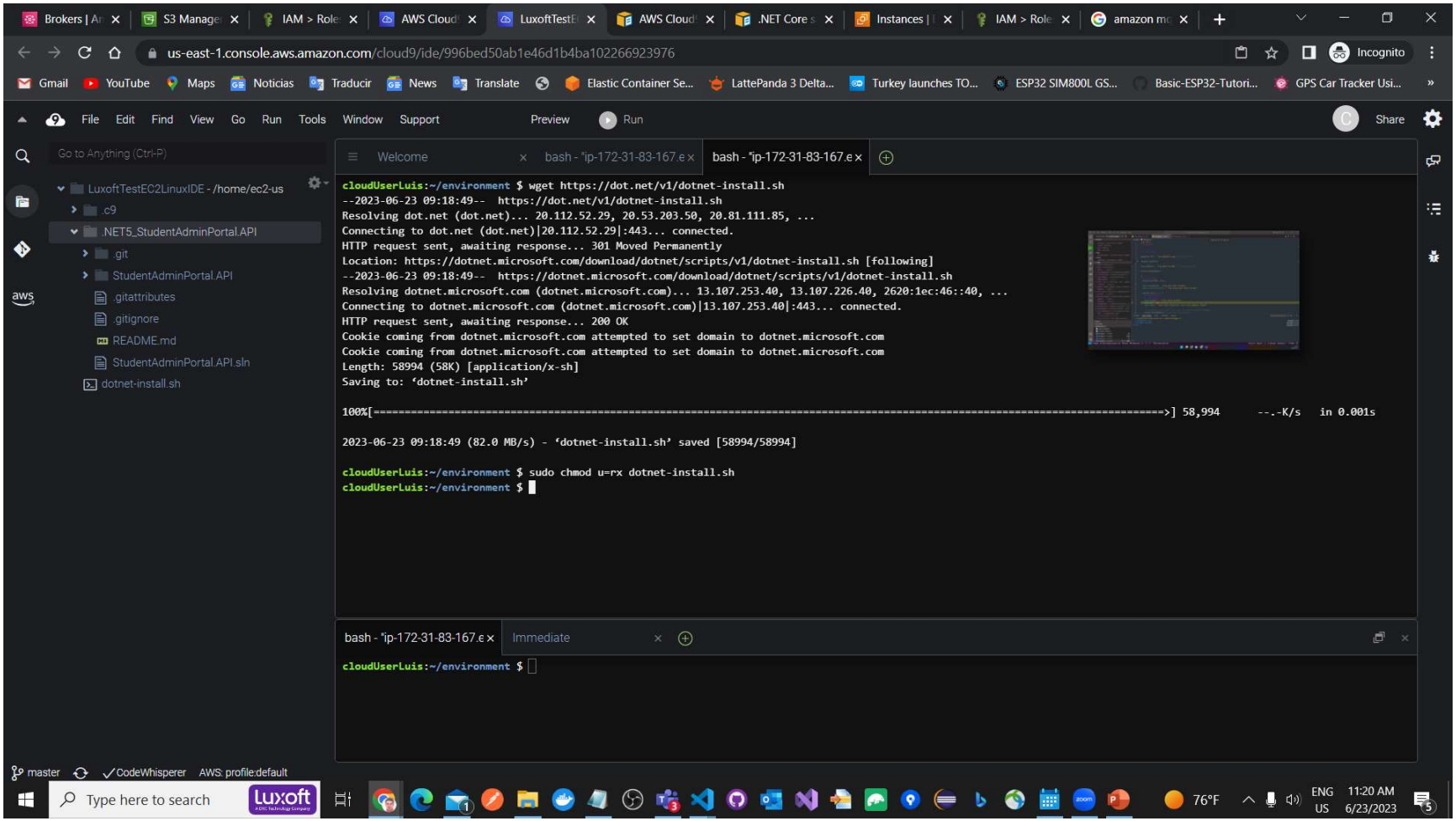
Download the .NET Core SDK installer script into your environment by running the following command.

wget https://dot.net/v1/dotnet-install.sh



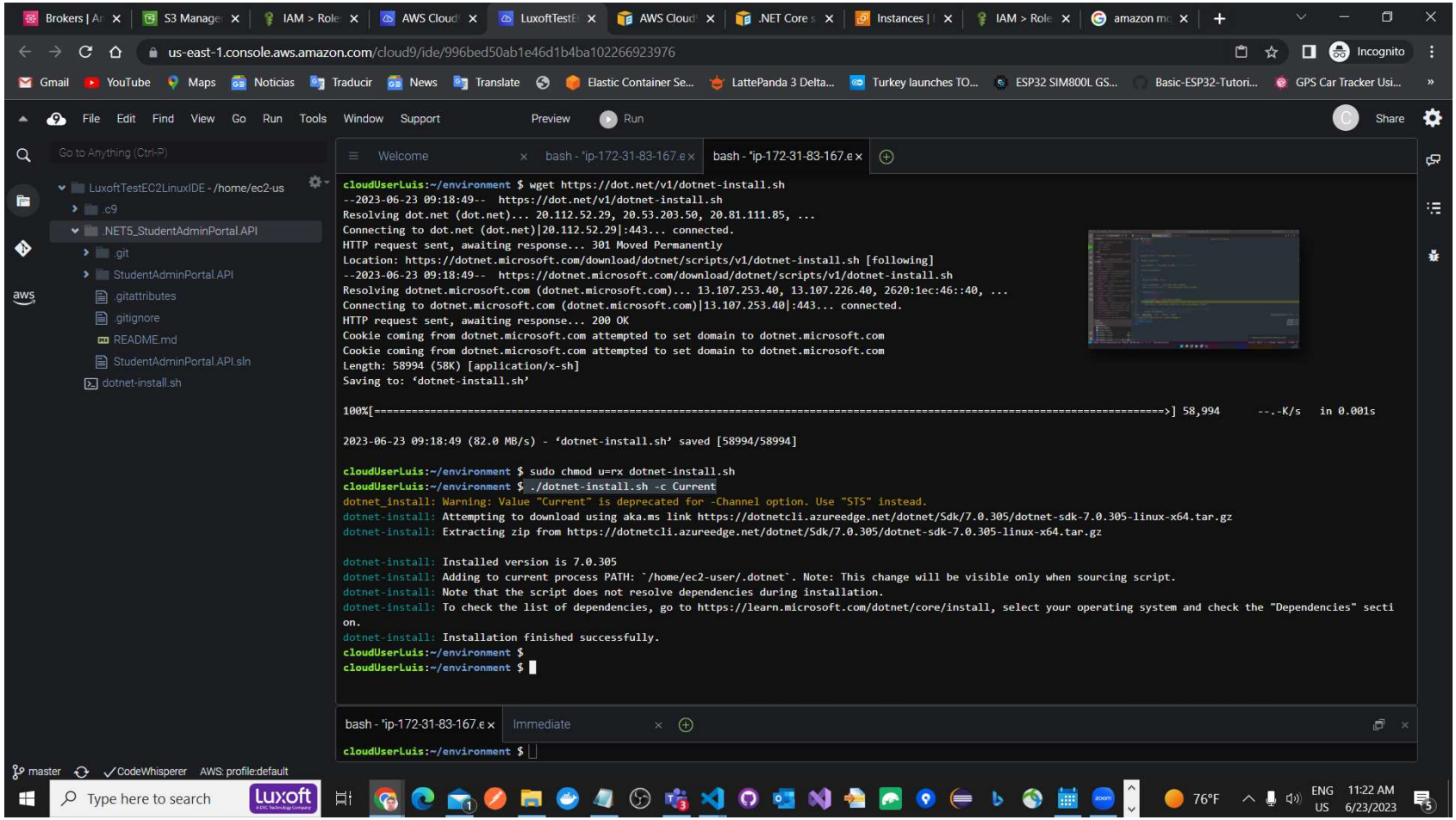
Make the installer script executable by the current user by running the following command.

```
sudo chmod u=rx dotnet-install.sh
```



Run the installer script, which downloads and installs the .NET Core SDK, by running the following command.

./dotnet-install.sh -c Current



Add the .NET Core SDK to your `PATH`. To do this, in the shell profile for the environment (for example, the `.bashrc` file), add the `$HOME/.dotnet` subdirectory to the `PATH` variable for the environment, as follows.

a. Open the `.bashrc` file for editing by using the **`vi`** command.

```
vi ~/.bashrc
```

b. For Amazon Linux, using the down arrow or `j` key, move to the line that starts with `export PATH`.

   For Ubuntu Server, move to the last line of the file by typing `G`.

c. Using the right arrow or `$` key, move to the end of that line.

d. Switch to insert mode by pressing the `i` key. (`-- INSERT ---` will appear at the end of the display.)

e. For Amazon Linux, add the `$HOME/.dotnet` subdirectory to the **`PATH`** variable by typing `:$HOME/.dotnet`. Be sure to include the colon character (`:`). The line should now look similar to the following.

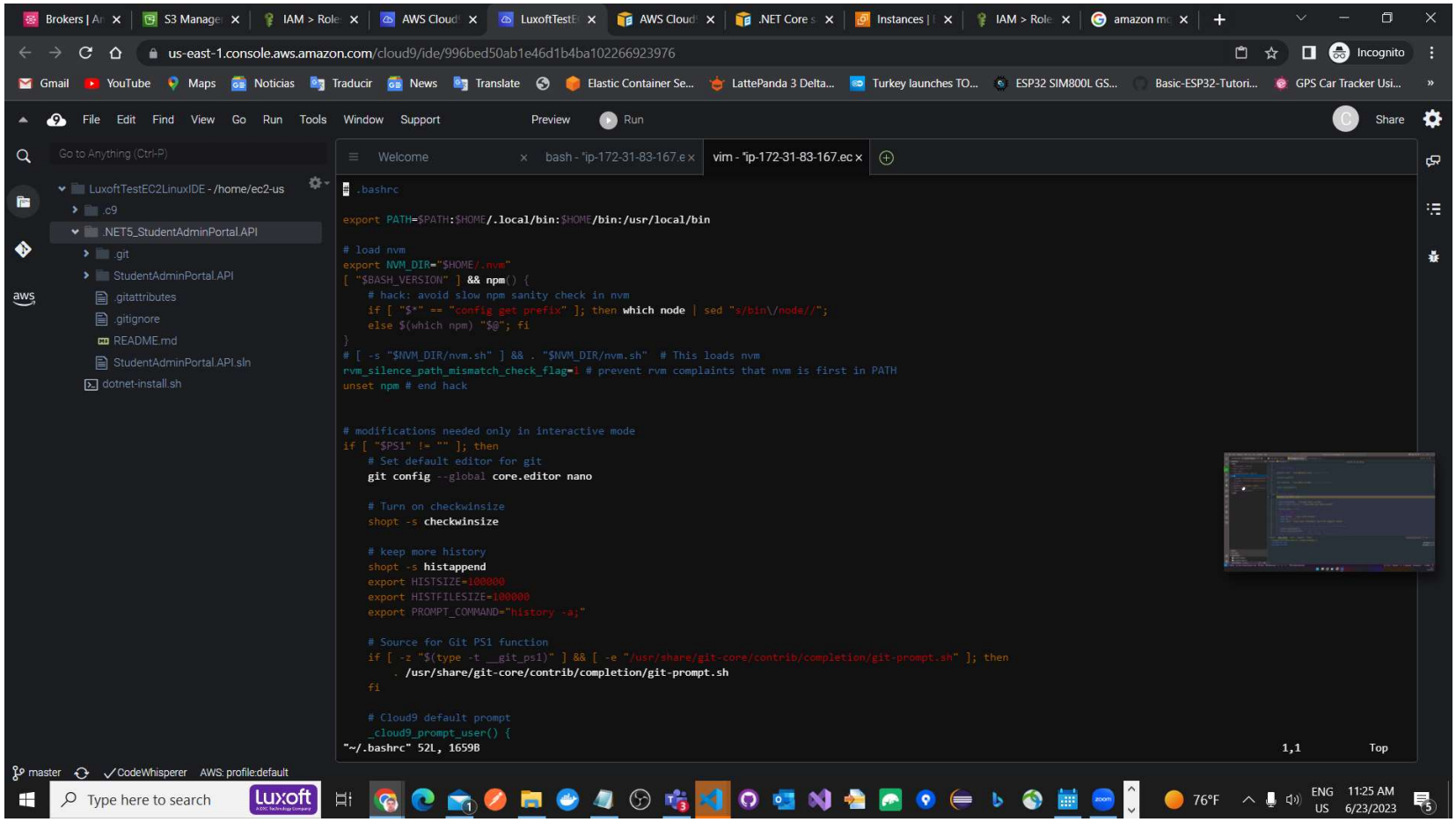
```
export PATH=$PATH:$HOME/.local/bin:$HOME/bin:$HOME/.dotnet
```

   For Ubuntu Server, press the right arrow key and then press `Enter` twice, followed by typing the following line by itself at the end of the file.

```
export PATH=$HOME/.dotnet:$PATH
```

f. Save the file. To do this, press the `Esc` key (`-- INSERT ---` will disappear from the end of the display), type `:wq` (to write to and then quit the file), and then press `Enter`.

Open the .bashrc file for editing by using the vi command:
vi ~/.bashrc



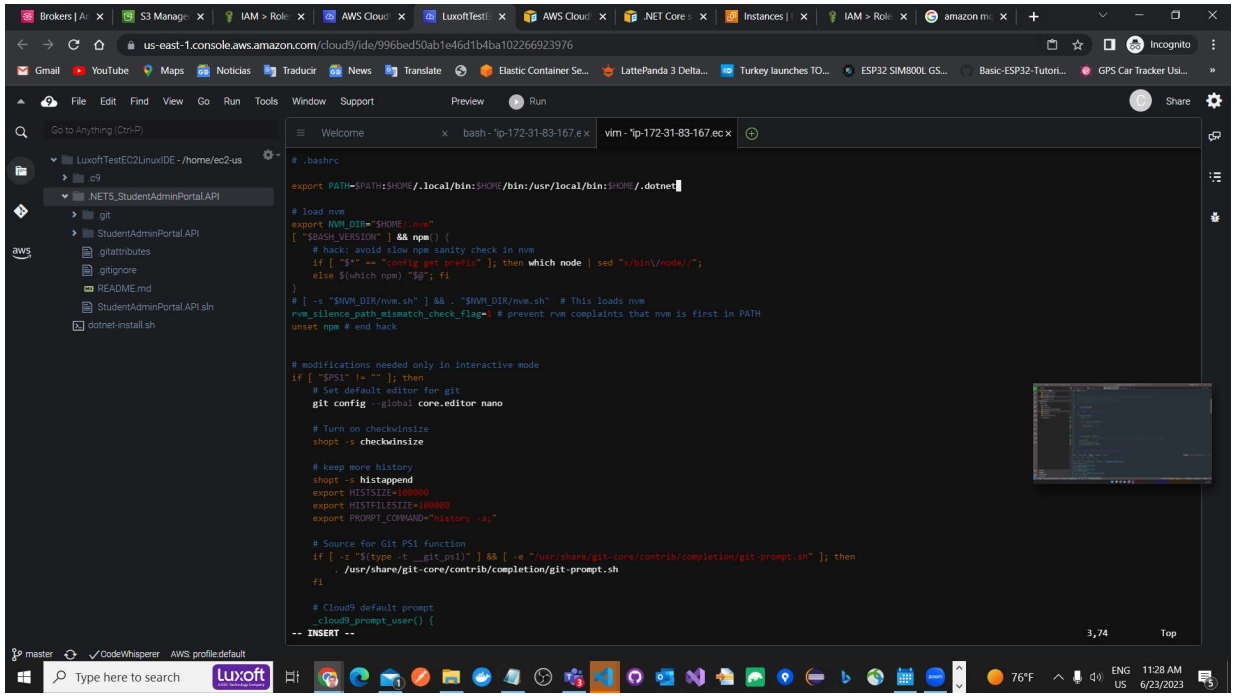
For Amazon Linux, using the down arrow or j key, move to the line that starts with export PATH.

Using the right arrow or $ key, move to the end of that line.

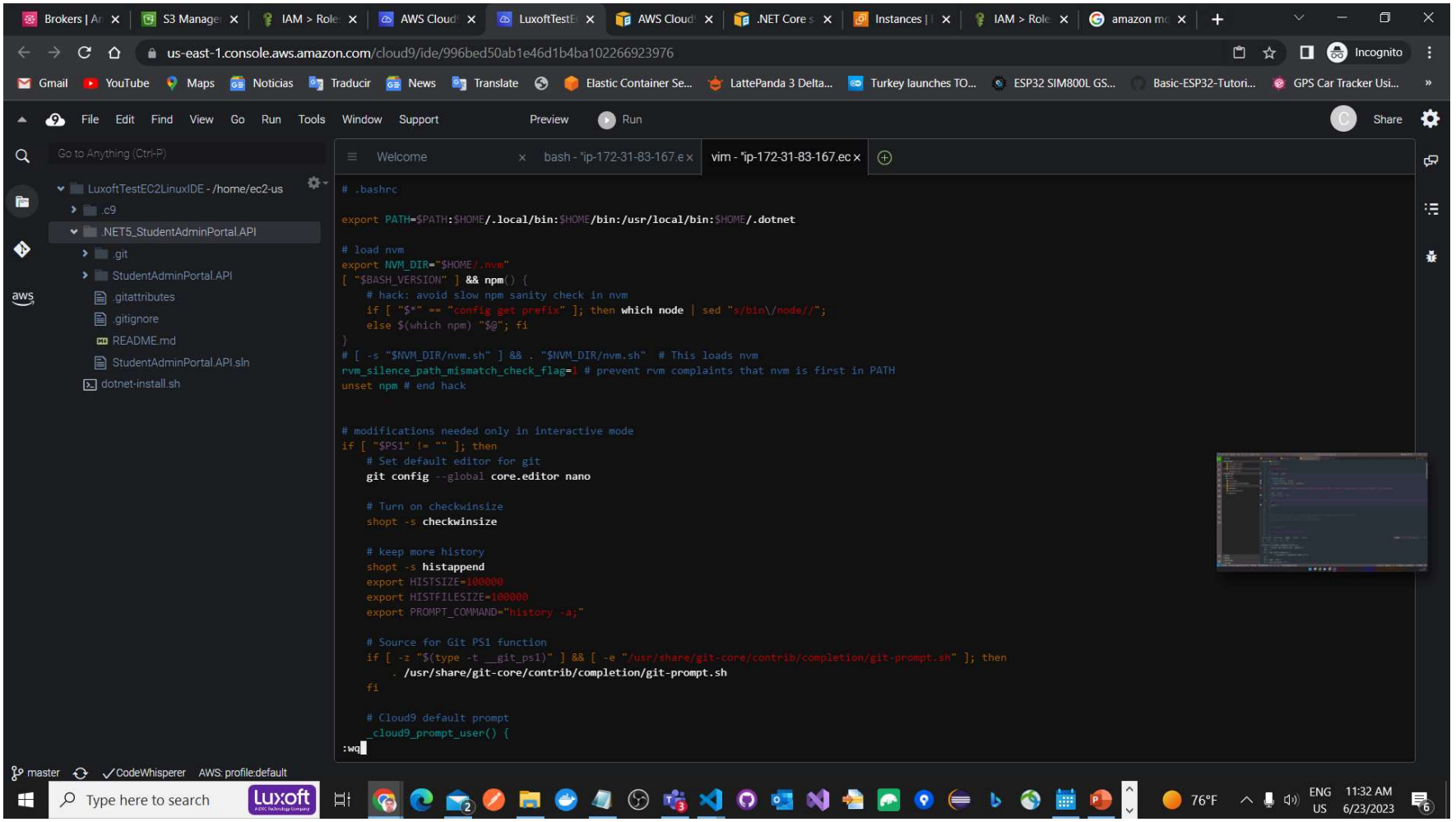
Switch to insert mode by pressing the i key. (-- INSERT -- will appear at the end of the display.)

For Amazon Linux, add the $HOME/.dotnet subdirectory to the PATH variable by typing :$HOME/.dotnet. Be sure to include the colon character (:). The line should now look similar to the following.

export PATH=$PATH:$HOME/.local/bin:$HOME/bin:$HOME/.dotnet



After including the new line press ESC key to exit from the INSERT mode
Type :wq to save the changes



Load the .NET Core SDK by sourcing the .bashrc file.

```
. ~/.bashrc
```

Confirm the .NET Core SDK is loaded by running .NET Core CLI with the --help option.

```
dotnet --help
```

If you no longer want to keep the .NET Core SDK installer script in your environment, you can delete it as follows.

```
rm dotnet-install.sh
```

Confirm whether the latest version of the .NET Core SDK is already installed in your environment. To do this, in a terminal session in the AWS Cloud9 IDE, run the .NET Core command line interface (CLI) with the --version option.

```
dotnet --version
```

Step 2 (Optional): Install the .NET CLI extension for Lambda functions

Although not required for this tutorial, you can deploy AWS Lambda functions and AWS Serverless Application Model applications using the dotnet CLI if you also install the `Amazon.Lambda.Tools` package.

1. To install the package, run the following command:

```
dotnet tool install -g Amazon.Lambda.Tools
```

2. Now set the `PATH` and `DOTNET_ROOT` environment variable to point to the installed Lambda tool. In the `.bashrc` file, find the `export PATH` section and edit it so that it appears similar to the following (see Step 1 for details on editing this file):

```
export PATH=$PATH:$HOME/.local/bin:$HOME/bin:$HOME/.dotnet:$HOME/.dotnet/tools
export DOTNET_ROOT=$HOME/.dotnet
```

To install the package, run the following command: dotnet tool install -g Amazon.Lambda.Tools

Now set the PATH and DOTNET_ROOT environment variable to point to the installed Lambda tool. In the .bashrc file, find the export PATH section and edit it so that it appears similar to the following (see Step 1 for details on editing this file):

export PATH=$PATH:$HOME/.local/bin:$HOME/bin:$HOME/.dotnet:$HOME/.dotnet/tools
export DOTNET_ROOT=$HOME/.dotnet

Brokers | Ar | S3 Manage | IAM > Role | AWS Cloud | LuxoftTestE | AWS Cloud | .NET Core s | Instances | | IAM > Role | amazon mc

us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976

Incognito

Gmail YouTube Maps Noticias Traducir News Translate Elastic Container Se... LattePanda 3 Delta... Turkey launches TO... ESP32 SIM800L GS... Basic-ESP32-Tutori... GPS Car Tracker Usi...

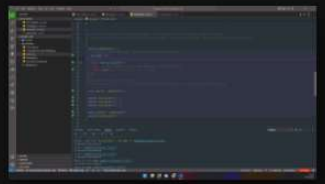
File Edit Find View Go Run Tools Window Support Preview Run Share

Go to Anything (Ctrl-P)

LuxoftTestEC2LinuxIDE - /home/ec2-us
- .c9
- .NET5_StudentAdminPortal.API
  - .git
  - StudentAdminPortal.API
  - .gitattributes
  - .gitignore
  - README.md
  - StudentAdminPortal.API.sln
- dotnet-install.sh

Welcome | bash - "ip-172-31-83-167.e | dotnet - "ip-172-31-83-167

```
cloudUserLuis:~/environment $ dotnet tool install -g Amazon.Lambda.Tools
Tools directory '/home/ec2-user/.dotnet/tools' is not currently on the PATH environment variable.
If you are using bash, you can add it to your profile by running the following command:

cat << \EOF >> ~/.bash_profile
# Add .NET Core SDK tools
export PATH="$PATH:/home/ec2-user/.dotnet/tools"
EOF

You can add it to the current session by running the following command:

export PATH="$PATH:/home/ec2-user/.dotnet/tools"

You can invoke the tool using the following command: dotnet-lambda
Tool 'amazon.lambda.tools' (version '5.7.0') was successfully installed.
cloudUserLuis:~/environment $
```

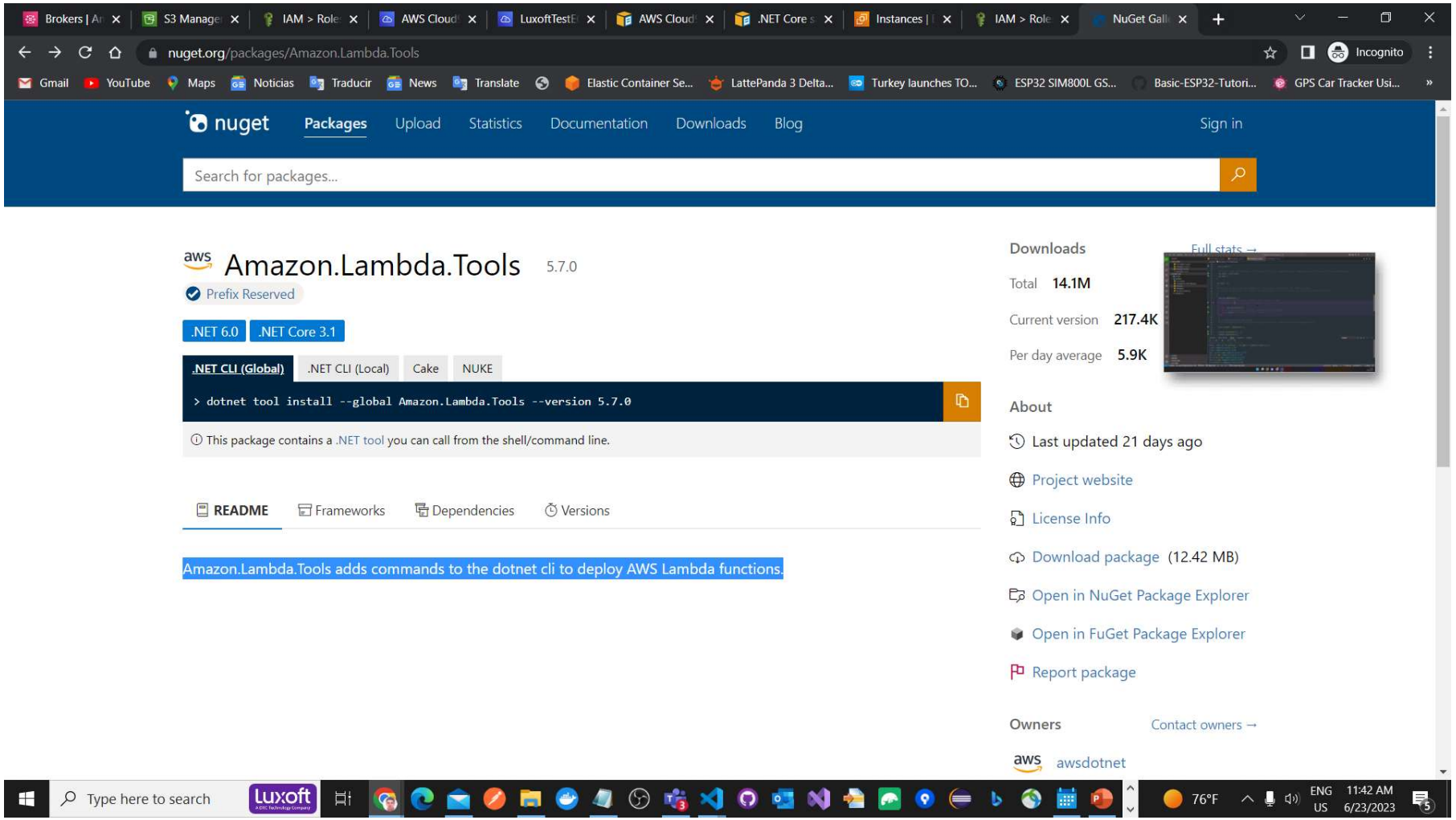


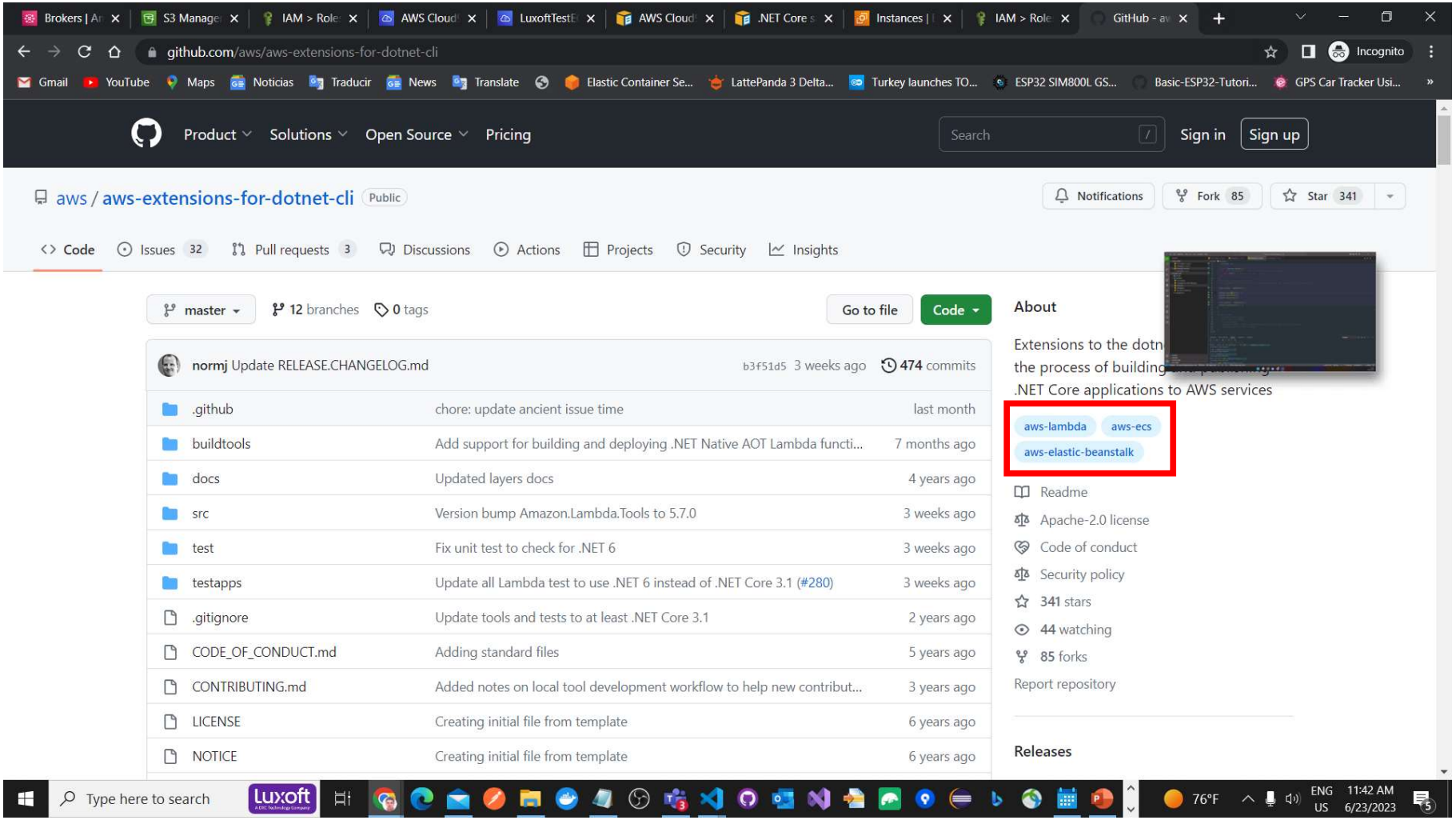
master CodeWhisperer AWS: profile:default

Type here to search

76°F ENG US 11:40 AM 6/23/2023

Amazon.Lambda.Tools adds commands to the dotnet cli to deploy AWS Lambda functions.





https://github.com/aws/aws-extensions-for-dotnet-cli

Step 3: Create a .NET Core console application project

In this step, you use .NET Core to create a project named `hello`. This project contains all of the files that .NET Core needs to run a simple application from the terminal in the IDE. The application's code is written in C#.

Create a .NET Core console application project. To do this, run the .NET Core CLI with the **`new`** command, specifying the console application project template type and the programming language to use (in this sample, C#).

The `-n` option indicates that the project is outputted to a new directory, `hello`. We then navigate to that directory.

```
dotnet new console -lang C# -n hello
cd hello
```

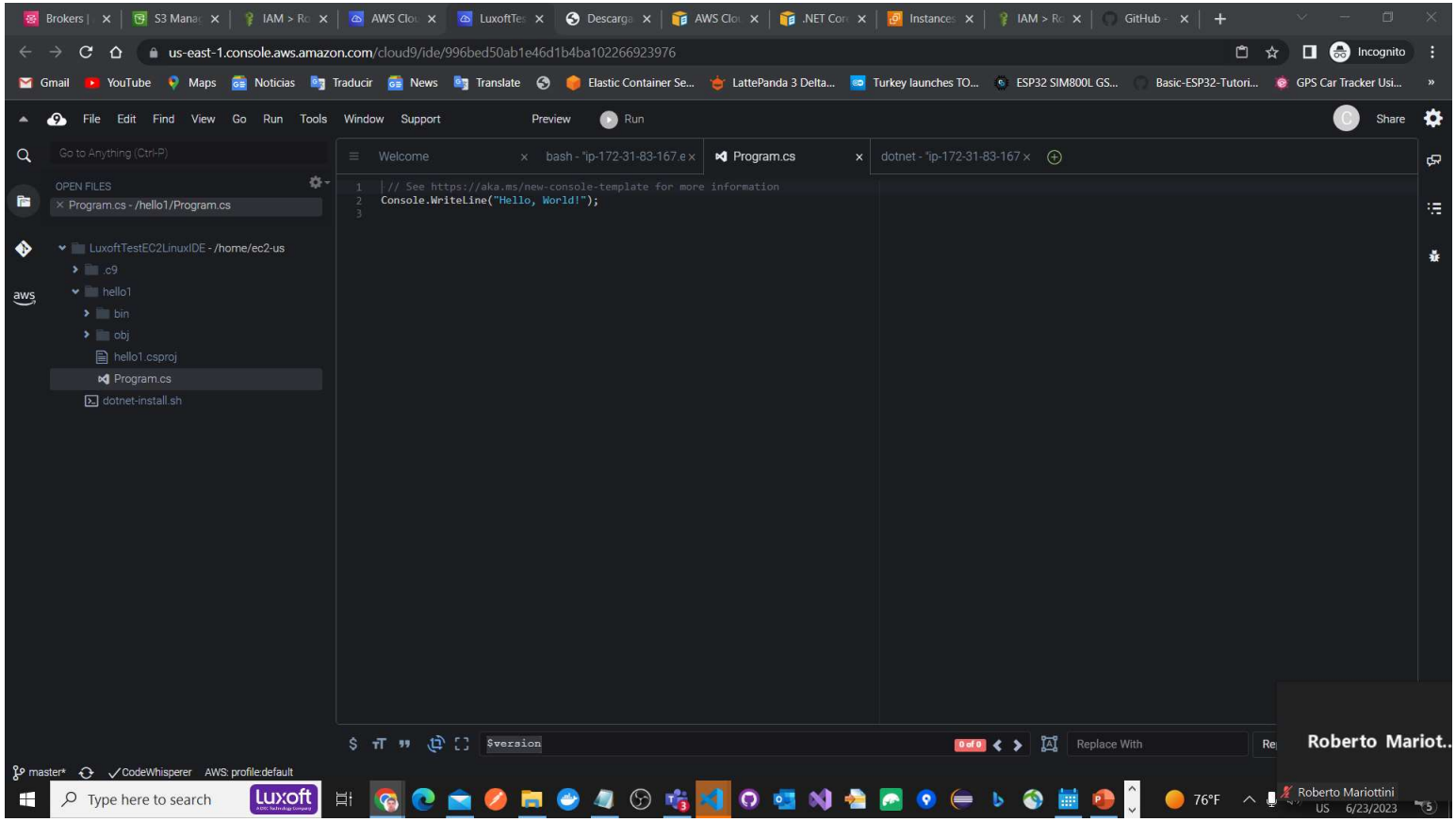
The preceding command adds a subdirectory named `obj` with several files, and some additional standalone files, to the `hello` directory. You should note the following two key files:

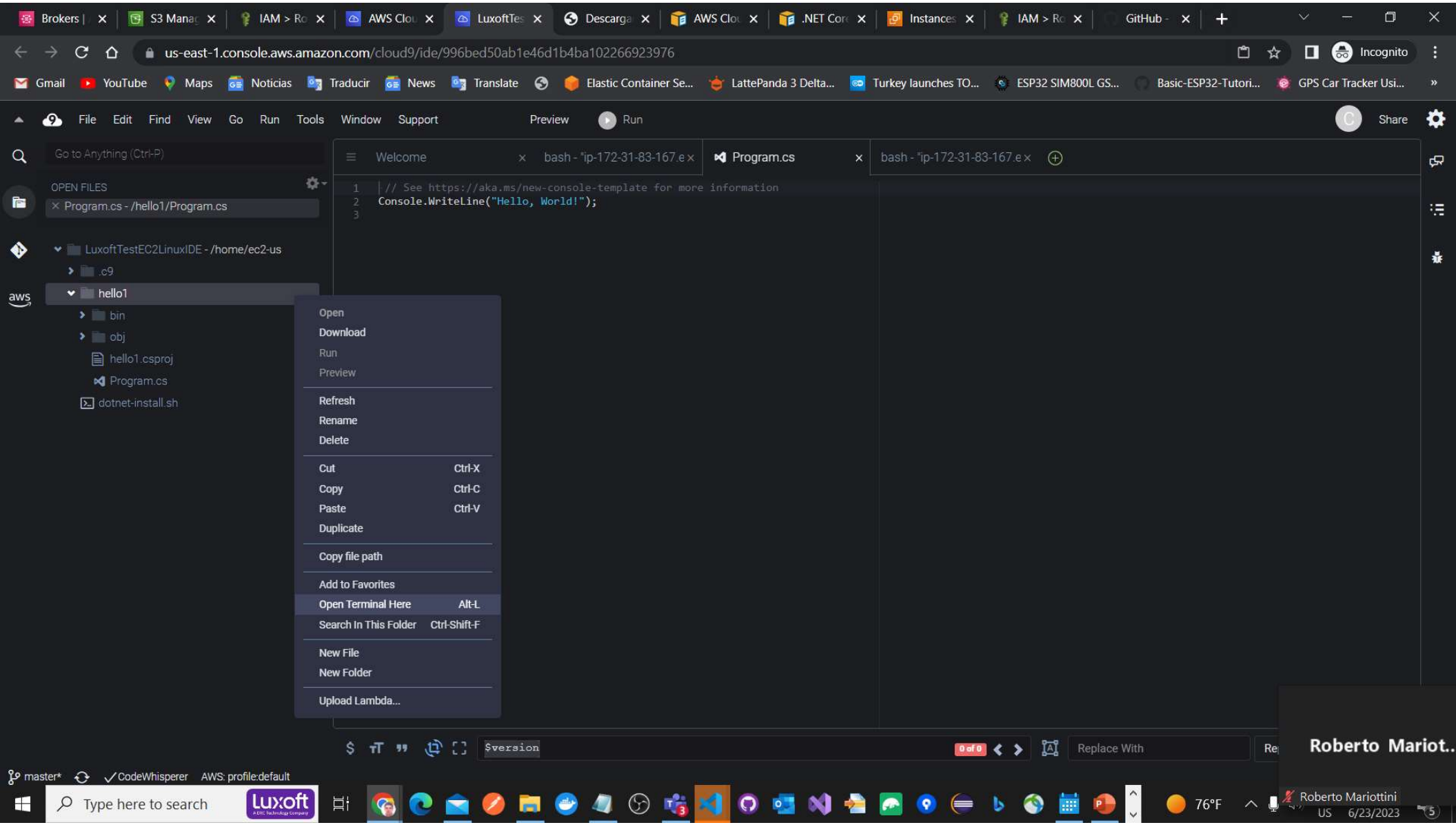
- The `hello/hello.csproj` file contains information about the console application project.
- The `hello/Program.cs` file contains the application's code to run.

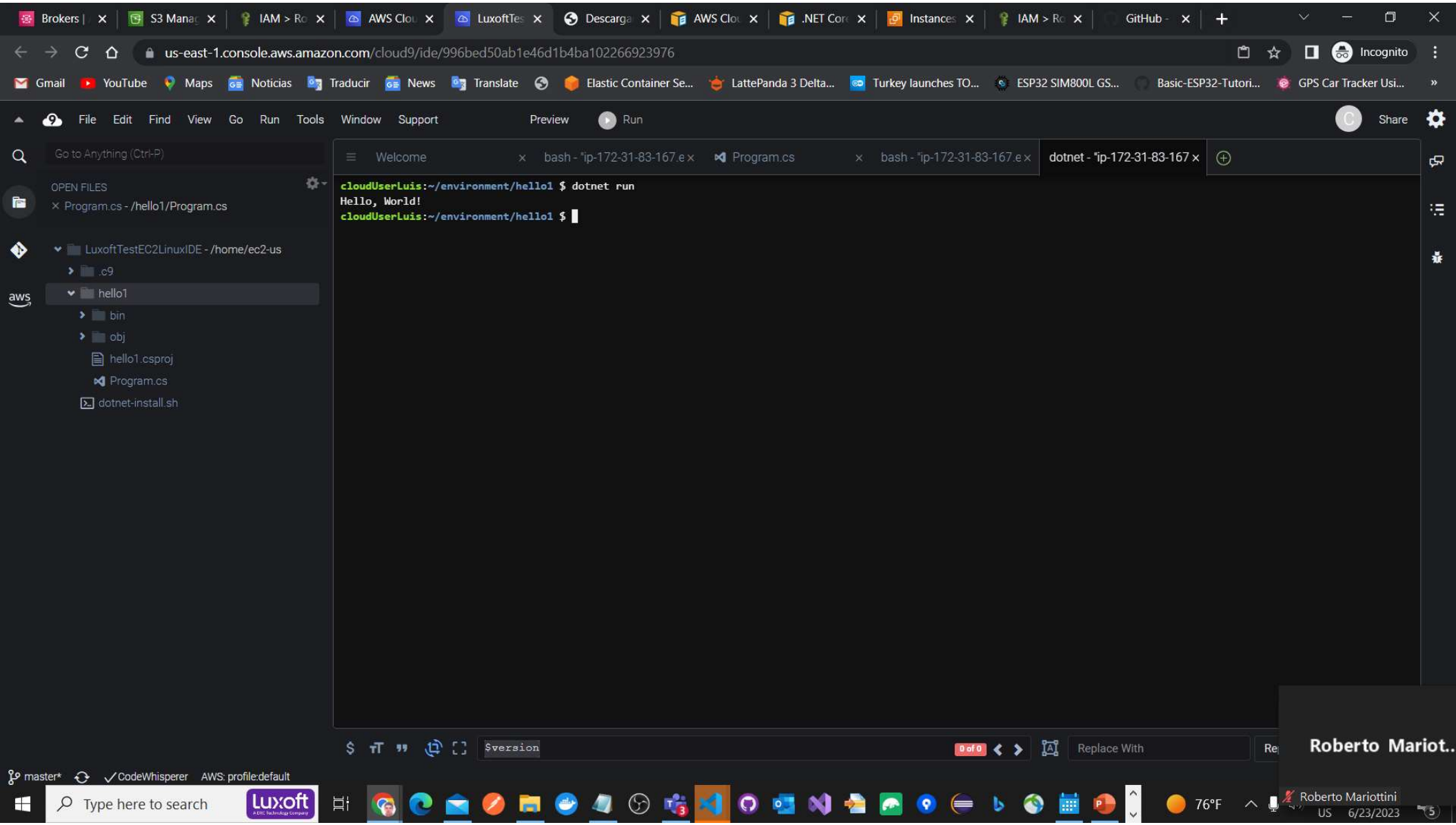
dotnet new console -lang C# -n hello
cd hello

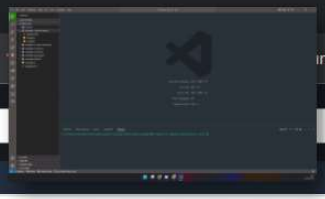
Step 4: Add code

In this step, you add some code to the application.
From the Environment window in the **AWS Cloud9 IDE**, open the hello/Program.cs file.
In the editor, replace the file's current contents with the following code, and then save the Program.cs file.




us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
OPEN FILES
Program.cs - /hello1/Program.cs
LuxoftTestEC2LinuxIDE - /home/ec2-us
.c9
hello1
bin
obj
hello1.csproj
Program.cs
dotnet-install.sh
Welcome
bash - "ip-172-31-83-167.e
Program.cs
bash - "ip-172-31-83-167.e
// See https://aka.ms/new-console-template for more information
Console.WriteLine("Hello, World!");
Open
Download
Run
Preview
Refresh
Rename
Delete
Cut Ctrl-X
Copy Ctrl-C
Paste Ctrl-V
Duplicate
Copy file path
Add to Favorites
Open Terminal Here Alt-L
Search In This Folder Ctrl-Shift-F
New File
New Folder
Upload Lambda...
$version
0 of 0
Replace With
Roberto Mariot..
master*
CodeWhisperer
AWS: profile:default
Type here to search
76°F
Roberto Mariottini
US 6/23/2023


us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
OPEN FILES
Program.cs - /hello1/Program.cs
LuxoftTestEC2LinuxIDE - /home/ec2-us
.c9
hello1
bin
obj
hello1.csproj
Program.cs
dotnet-install.sh
Welcome
bash - "ip-172-31-83-167.e
Program.cs
bash - "ip-172-31-83-167.e
dotnet - "ip-172-31-83-167
cloudUserLuis:~/environment/hello1 $ dotnet run
Hello, World!
cloudUserLuis:~/environment/hello1 $
$version
0 of 0
Replace With
master*
CodeWhisperer
AWS: profile:default
Type here to search
76°F
US
6/23/2023
Roberto Mariot..
Roberto Mariottini



Step 5: Build and run the code

In this step, you build the project and its dependencies into a set of binary files, including a runnable application file. Then you run the application.

1. In the IDE, create a builder for .NET Core as follows.
   a. On the menu bar, choose **Run, Build System, New Build System**.
   b. On the **My Builder.build** tab, replace the tab's contents with the following code.

```
{
  "cmd" : ["dotnet", "build"],
  "info" : "Building..."
}
```

   c. Choose **File, Save As**.
   d. For **Filename**, type `.NET Core.build`.
   e. For **Folder**, type `/.c9/builders`.
   f. Choose **Save**.
2. With the contents of the `Program.cs` file displayed in the editor, choose **Run, Build System, .NET Core**. Then choose **Run, Build**.

   This builder adds a subdirectory named `bin` and adds a subdirectory named `Debug` to the `hello/obj` subdirectory. Note the following three key files.

   - The `hello/bin/Debug/netcoreapp3.1/hello.dll` file is the runnable application file.
   - The `hello/bin/Debug/netcoreapp3.1/hello.deps.json` file lists the application's dependencies.
   - The `hello/bin/Debug/netcoreapp3.1/hello.runtimeconfig.json` file specifies the shared runtime and its version for the application.

   > **Note**
   > The folder name, `netcoreapp3.1`, reflects the version of the .NET Core SDK used in this example. You may see a different number in the folder name depending on the version you've installed.

3. Create a runner for .NET Core as follows.

3. Create a runner for .NET Core as follows.
   a. On the menu bar, choose **Run, Run With, New Runner**.
   b. On the **My Runner.run** tab, replace the tab's contents with the following code.

```
{
  "cmd" : ["dotnet", "run", "$args"],
  "working_dir": "$file_path",
  "info" : "Running..."
}
```

   c. Choose **File, Save As**.
   d. For **Filename**, type `.NET Core.run`.
   e. For **Folder**, type `/.c9/runners`.
   f. Choose **Save**.
4. Run the application with two integers to add (for example, `5` and `9`) as follows.
   a. With the contents of the `Program.cs` file displayed in the editor, choose **Run, Run Configurations, New Run Configuration**.
   b. In the **[New] - Idle** tab, choose **Runner: Auto**, and then choose **.NET Core**.
   c. In the **Command** box, type `hello 5 9`.
   d. Choose **Run**.

   By default, this runner instructs .NET Core to run the `hello.dll` file in the `hello/bin/Debug/netcoreapp3.1` directory.

   Compare your output to the following.

```
Hello, World!
The sum of 2 and 3 is 5.
The sum of 5 and 9 is 14.
```

Step 6: Create and set up a .NET Core console application project that uses the AWS SDK for .NET

You can enhance this sample to use the AWS SDK for .NET to create an Amazon S3 bucket, list your available buckets, and then delete the bucket you just created.

In this new project, you add a reference to the AWS SDK for .NET. The AWS SDK for .NET provides a convenient way to interact with AWS services such as Amazon S3, from your .NET code. You then set up AWS credentials management in your environment. The AWS SDK for .NET needs these credentials to interact with AWS services.

To create the project

1. Create a .NET Core console application project. To do this, run the .NET Core CLI with the `new` command, specifying the console application project template type and the programming language to use.

   The `-n` option indicates that the project is outputted to a new directory, `s3`. We then navigate to that directory.

   ```
   dotnet new console -lang C# -n s3
   cd s3
   ```

2. Add a project reference to the Amazon S3 package in the AWS SDK for .NET. To do this, run the .NET Core CLI with the `add package` command, specifying the name of the Amazon S3 package in NuGet. (NuGet defines how packages for .NET are created, hosted, and consumed, and provides the tools for each of those roles.)

   ```
   dotnet add package AWSSDK.S3
   ```

   When you add a project reference to the Amazon S3 package, NuGet also adds a project reference to the rest of the AWS SDK for .NET.

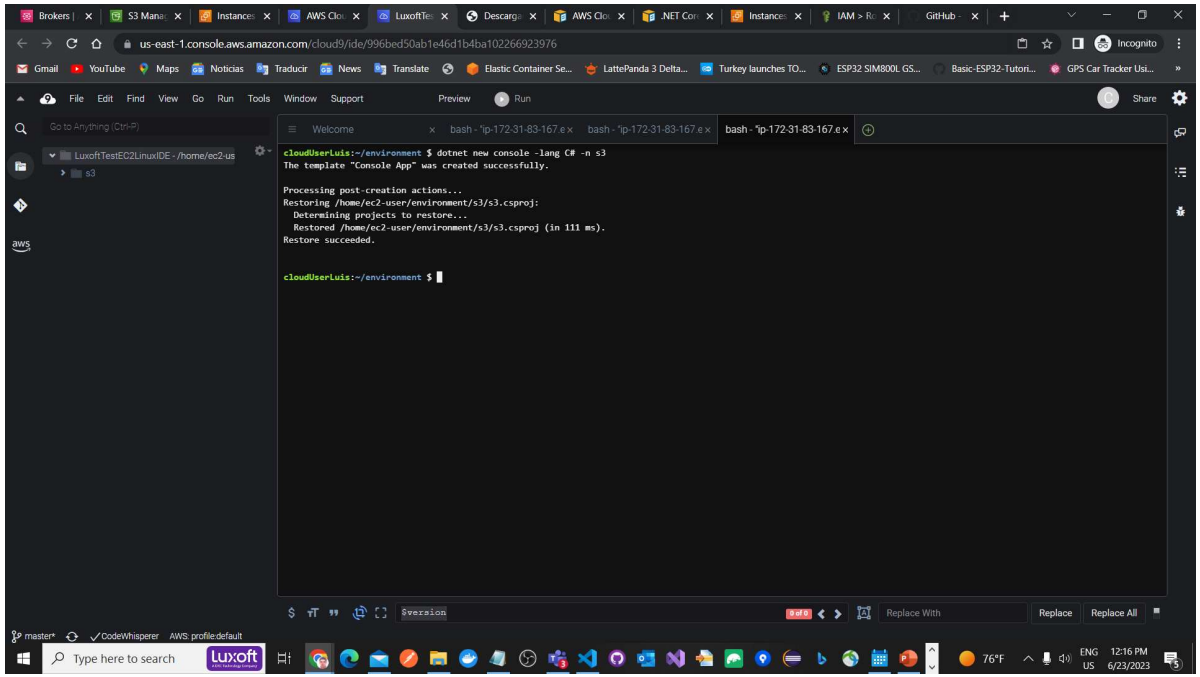
   > **Note**
   > For the names and versions of other AWS related packages in NuGet, see NuGet packages tagged with aws-sdk on the NuGet website.

Step 6: Create and set up a .NET Core console application project that uses the AWS SDK for .NET
Create a .NET Core console application project. To do this, run the .NET Core CLI with the new command, specifying the console application project template type and the programming language to use.
The -n option indicates that the project is outputted to a new directory, s3. We then navigate to that directory.

```
dotnet new console -lang C# -n s3
cd s3
```

Add a project reference to the Amazon S3 package in the AWS SDK for .NET. To do this, run the .NET Core CLI with the add package command, specifying the name of the Amazon S3 package in NuGet. (NuGet defines how packages for .NET are created, hosted, and consumed, and provides the tools for each of those roles.)

```
dotnet add package AWSSDK.S3
```



To set up AWS credentials management

Each time you use the AWS SDK for .NET to call an AWS service, you must provide a set of AWS credentials with the call. These credentials determine whether the AWS SDK for .NET has the appropriate permissions to make that call. If the credentials don't cover the appropriate permissions, the call will fail.

To store your credentials within the environment, follow the instructions in Calling AWS services from an environment in AWS Cloud9, and then return to this topic.

For additional information, see Configuring AWS Credentials in the *AWS SDK for .NET Developer Guide*.

https://docs.aws.amazon.com/cloud9/latest/user-guide/credentials.html

Create an instance profile with the IAM console

Note

If you already have an IAM role that contains an instance profile, skip ahead to Attach an Instance Profile to an Instance with the Amazon EC2 Console.

1. Sign in to the IAM console, at https://console.aws.amazon.com/iam.

   For this step, we recommend you sign in using administrator-level credentials in your AWS account. If you can't do this, check with your AWS account administrator.
2. In the navigation bar, choose **Roles**.

   Note

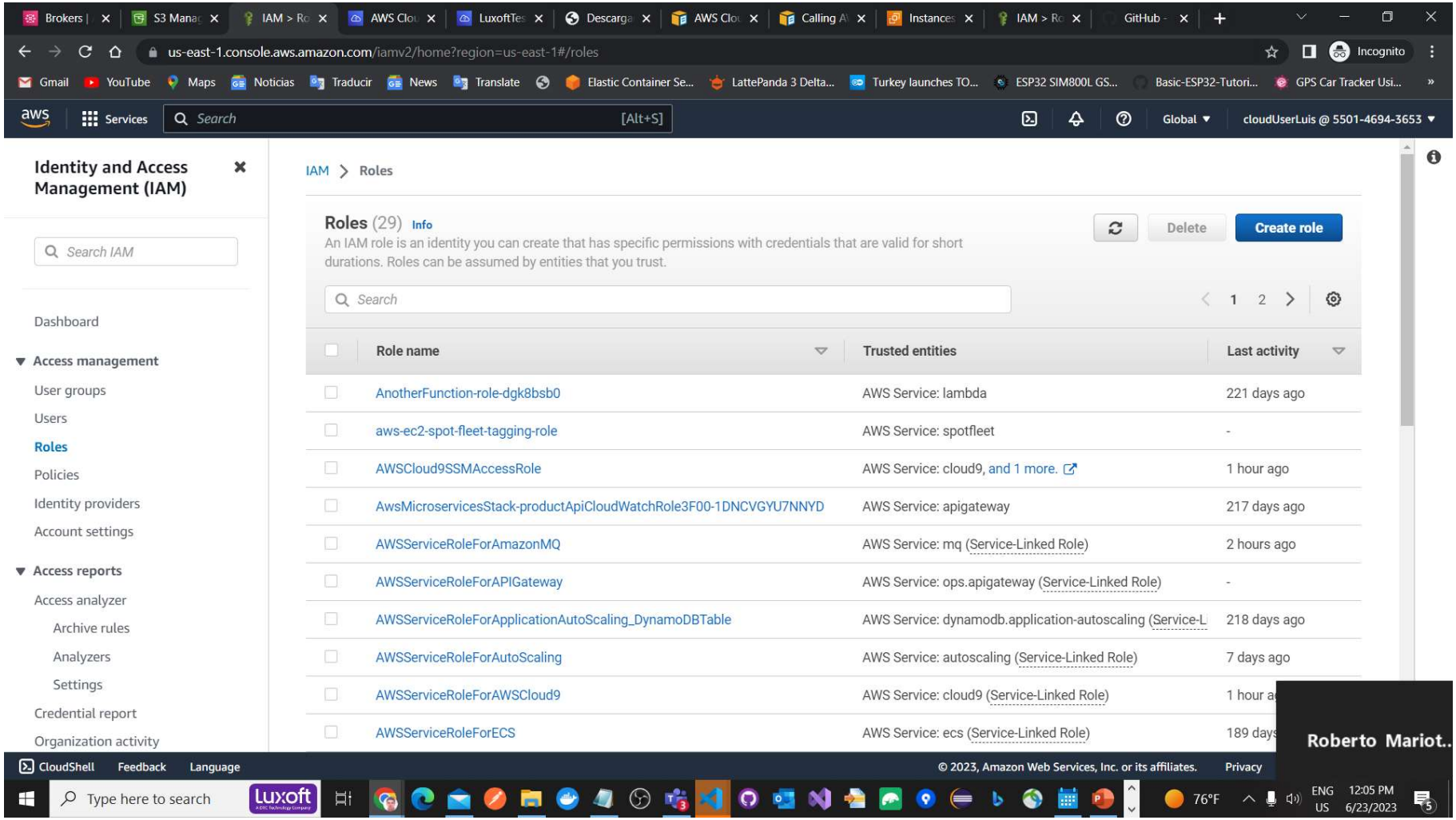
   You cannot use the IAM console to create an instance profile by itself. You must create an IAM role, which contains an instance profile.
3. Choose **Create role**.
4. On the **Select type of trusted entity** page, with **AWS service** already chosen, for **Choose the service that will use this role**, choose **EC2**.
5. For **Select your use case**, choose **EC2**.
6. Choose **Next: Permissions**.
7. On the **Attach permissions policies** page, in the list of policies, select the box next to **AdministratorAccess**, and then choose **Next: Review**.

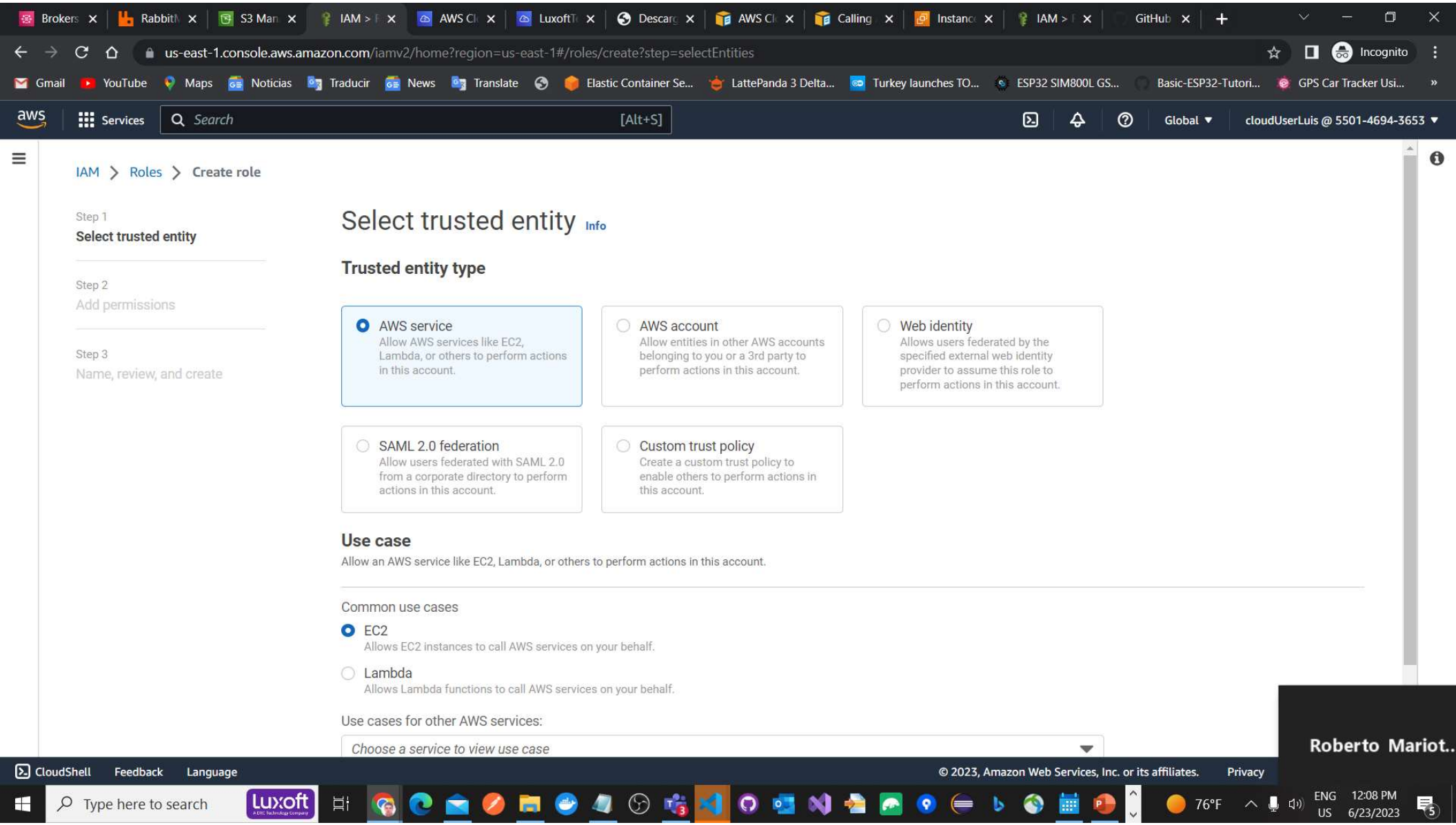
   Note

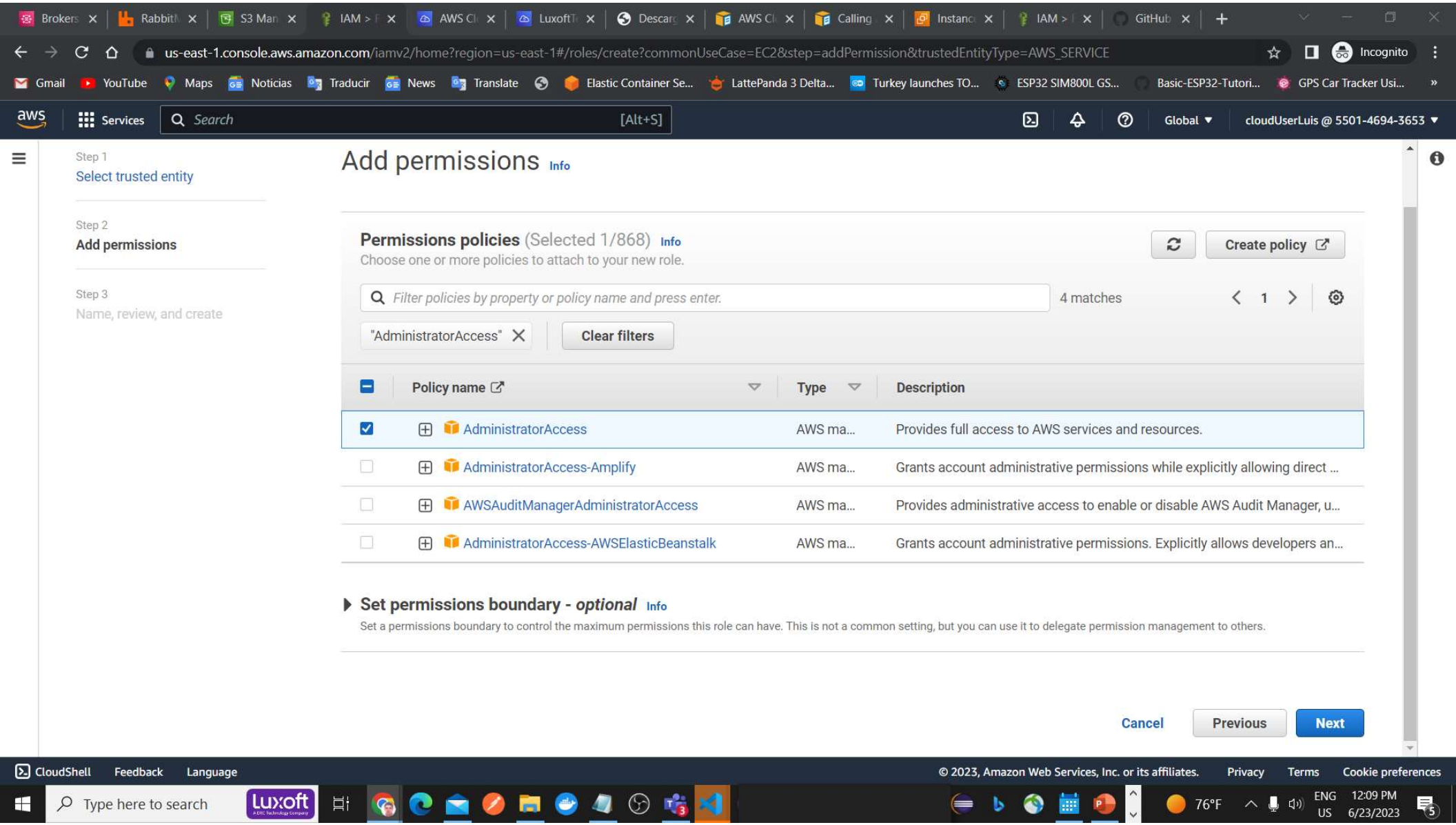
   The **AdministratorAccess** policy allows unrestricted access to all AWS actions and resources across your AWS account. Use it only for experimentation purposes. For more information, see IAM Policies in the *IAM User Guide*.
8. On the **Review** page, for **Role Name**, enter a name for the role (for example, `my-demo-cloud9-instance-profile`).
9. Choose **Create Role**.

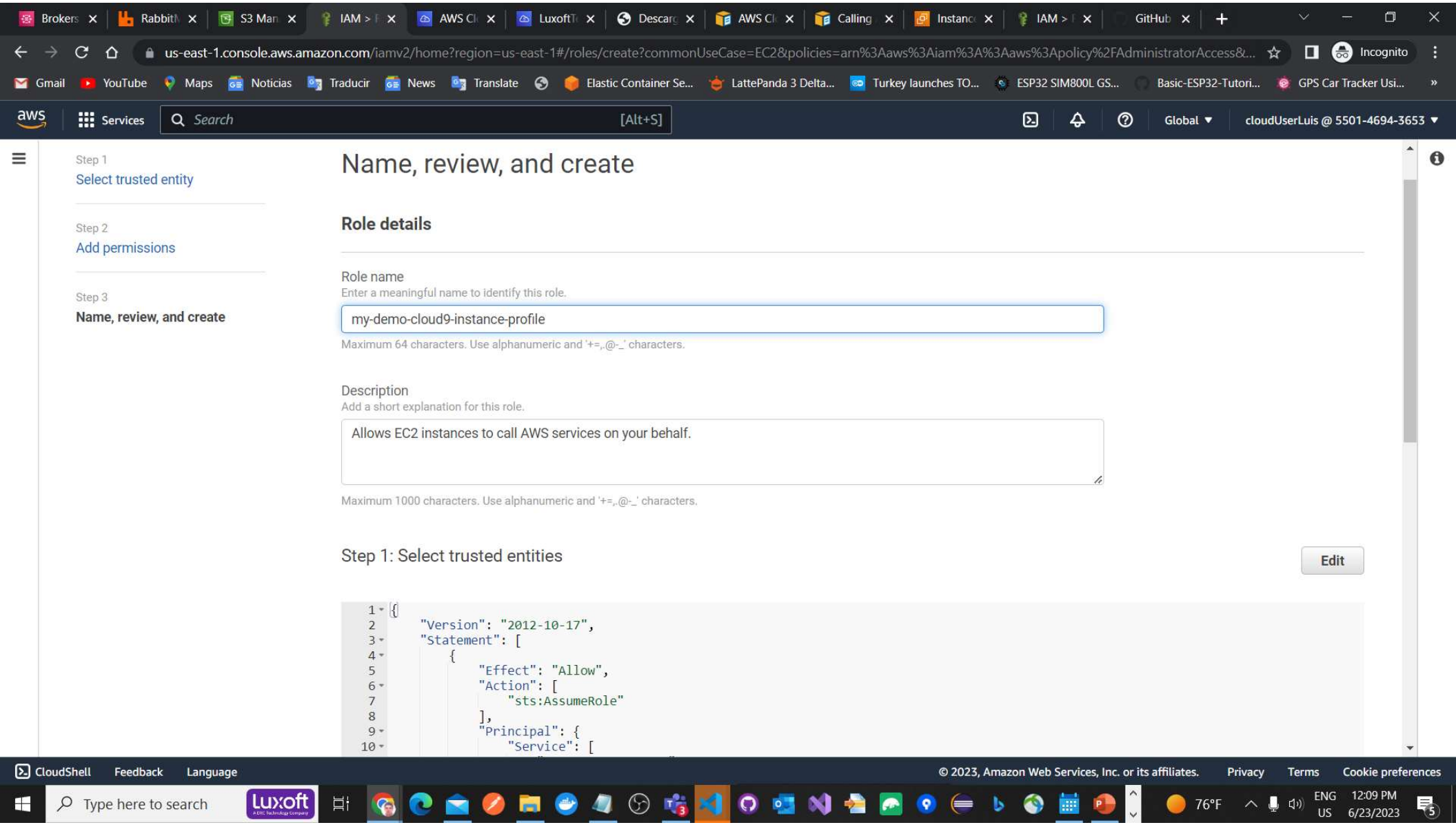
Skip ahead to Attach an Instance Profile to an Instance with the Amazon EC2 Console.

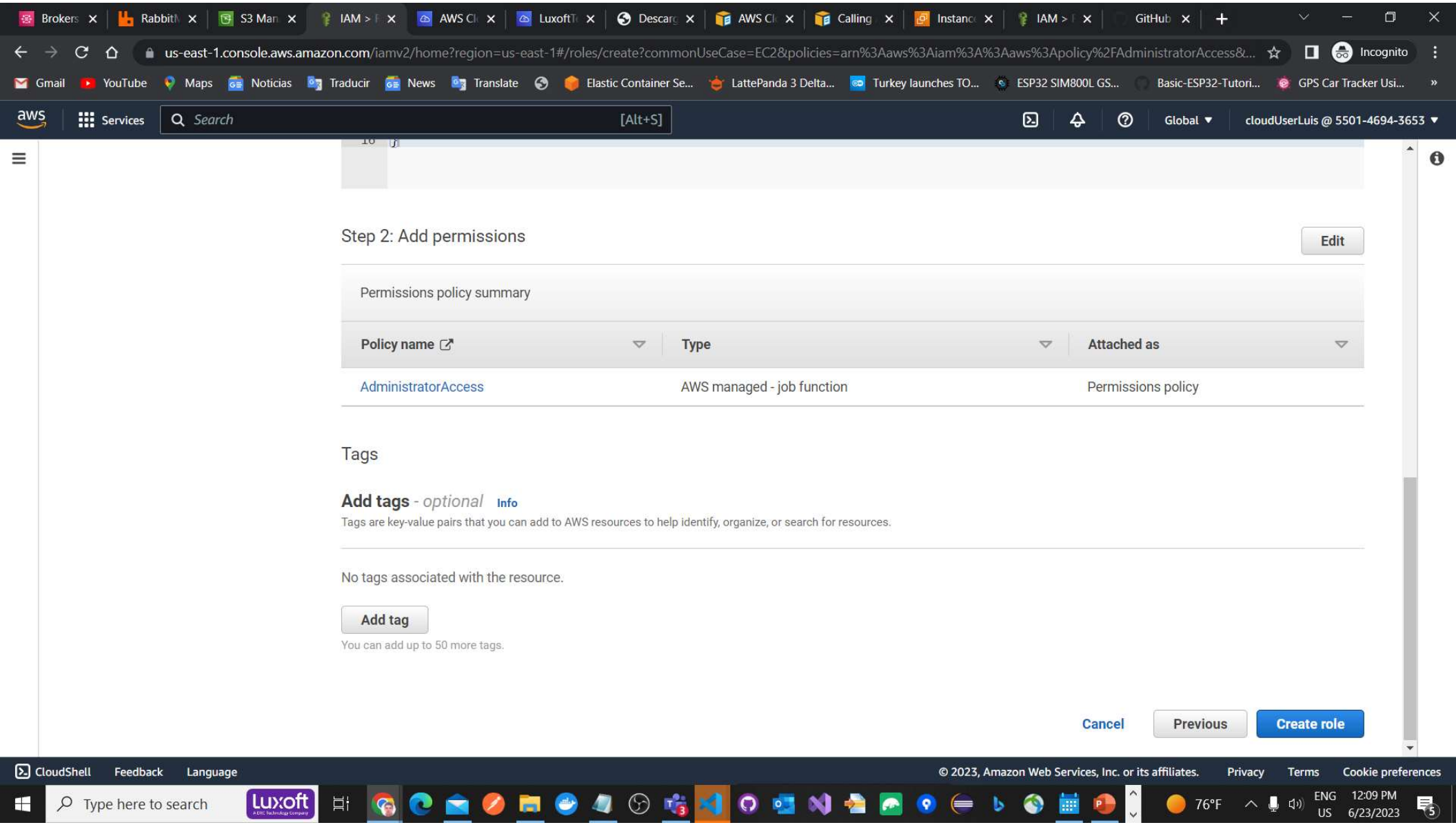
Create a new Role to attach to the EC2 virtual machine where is hosted the AWS Cloud9 IDE

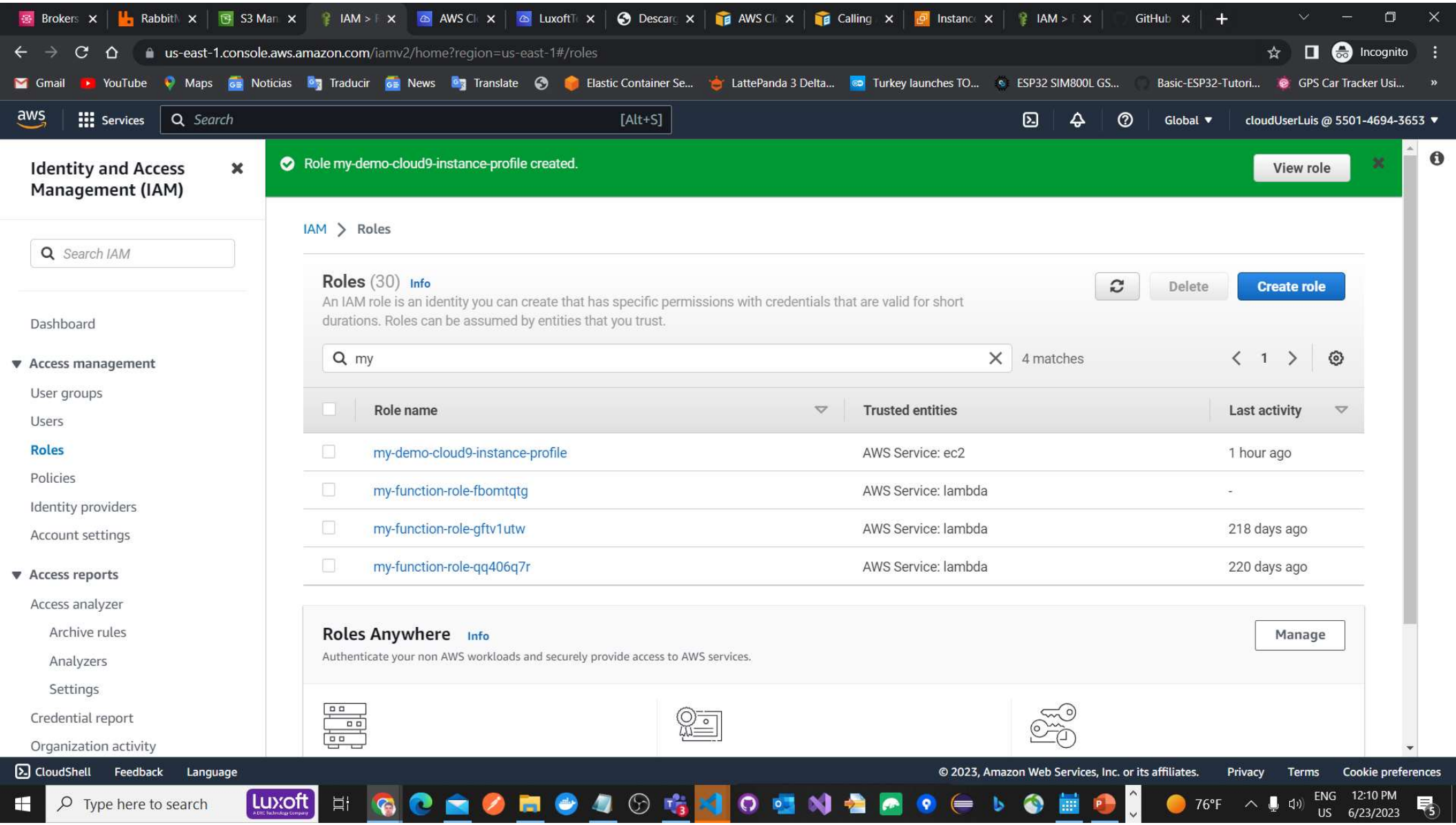



us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/create?step=selectEntities
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
Global
cloudUserLuis @ 5501-4694-3653
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Select trusted entity Info
Trusted entity type
AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.
AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.
Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.
Common use cases
EC2
Allows EC2 instances to call AWS services on your behalf.
Lambda
Allows Lambda functions to call AWS services on your behalf.
Use cases for other AWS services:
Choose a service to view use case
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Roberto Mariot..
Type here to search
76°F
ENG
US
12:08 PM
6/23/2023

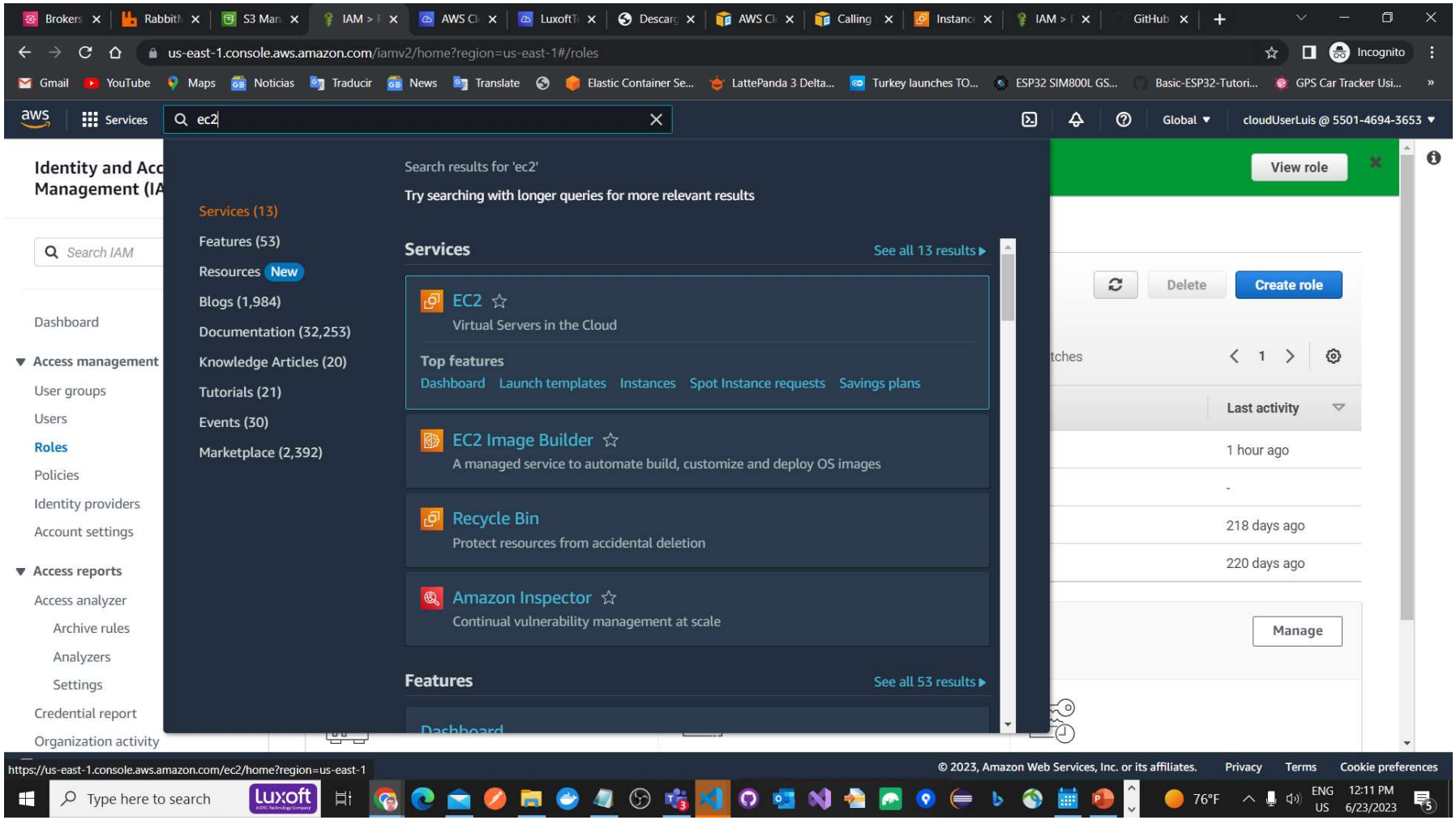

us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/create?commonUseCase=EC2&step=addPermission&trustedEntityType=AWS_SERVICE
Incognito
Services
Search
[Alt+S]
Global
cloudUserLuis @ 5501-4694-3653
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (Selected 1/868) Info
Choose one or more policies to attach to your new role.
Create policy
Filter policies by property or policy name and press enter.
4 matches
1
"AdministratorAccess"
Clear filters
Policy name
Type
Description
AdministratorAccess
AWS ma...
Provides full access to AWS services and resources.
AdministratorAccess-Amplify
AWS ma...
Grants account administrative permissions while explicitly allowing direct ...
AWSAuditManagerAdministratorAccess
AWS ma...
Provides administrative access to enable or disable AWS Audit Manager, u...
AdministratorAccess-AWSElasticBeanstalk
AWS ma...
Grants account administrative permissions. Explicitly allows developers an...
Set permissions boundary - optional Info
Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting, but you can use it to delegate permission management to others.
Cancel
Previous
Next
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
76°F
ENG
US
12:09 PM
6/23/2023

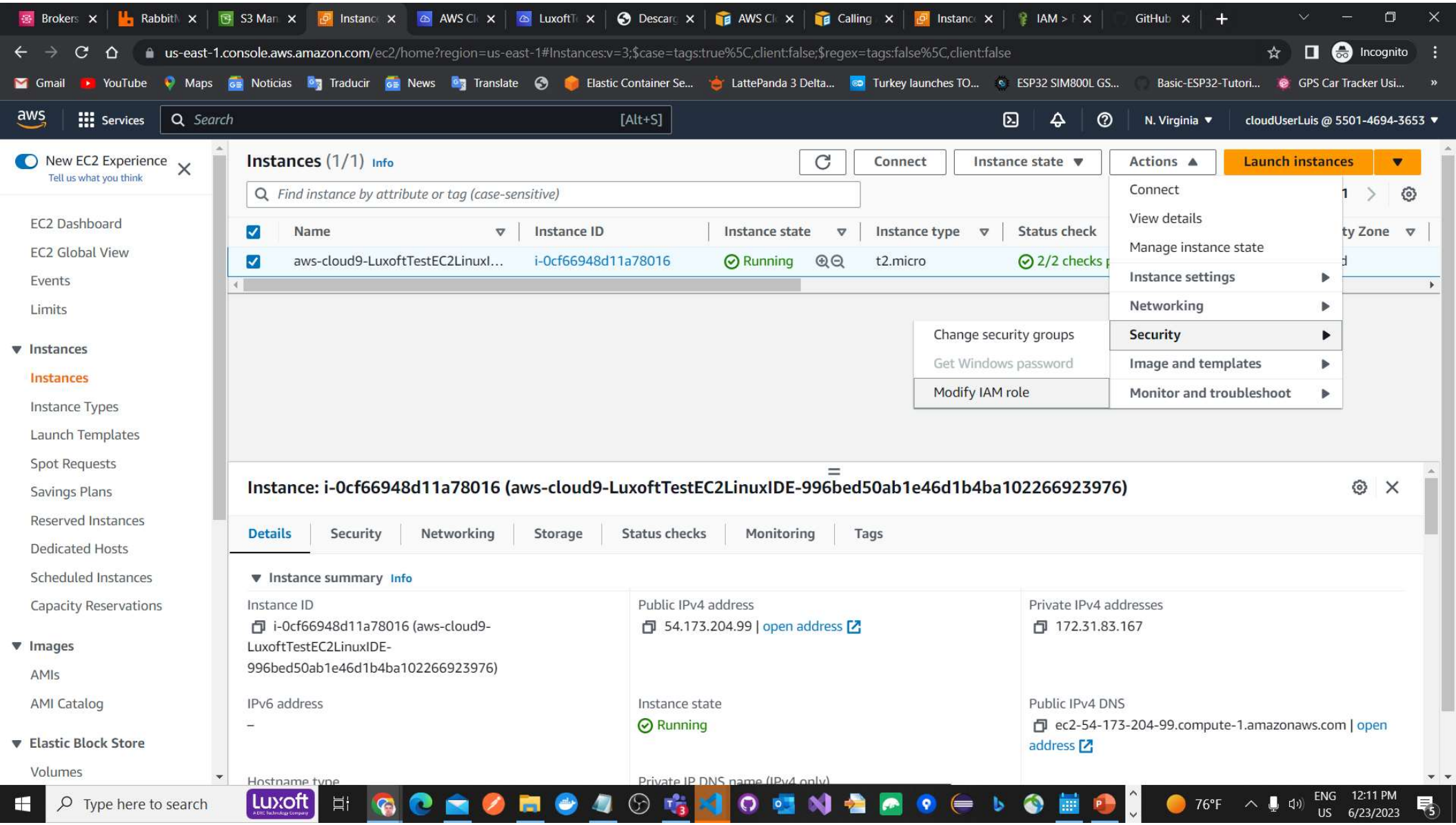

Brokers
RabbitM
S3 Man
IAM > F
AWS Cl
LuxoftT
Descarg
AWS Cl
Calling
Instanc
IAM > F
GitHub
us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/create?commonUseCase=EC2&policies=arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAdministratorAccess&...
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
Global
cloudUserLuis @ 5501-4694-3653
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Name, review, and create
Role details
Role name
Enter a meaningful name to identify this role.
my-demo-cloud9-instance-profile
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allows EC2 instances to call AWS services on your behalf.
Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.
Step 1: Select trusted entities
Edit
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": [
"sts:AssumeRole"
],
"Principal": {
"Service": [
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
76°F
ENG
US
12:09 PM
6/23/2023

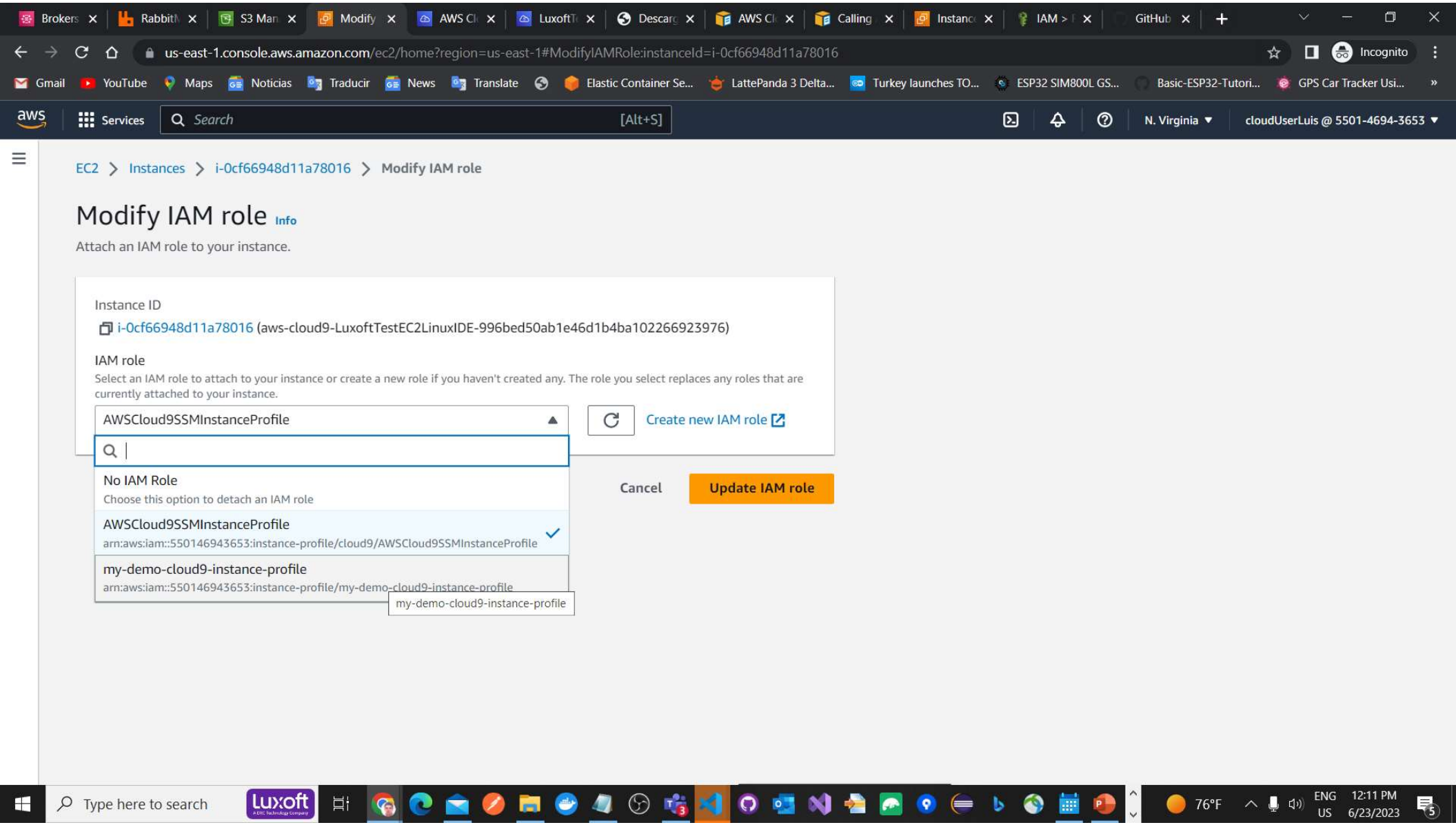

Brokers
RabbitM
S3 Man
IAM > R
AWS Cl
LuxoftT
Descarg
AWS Cl
Calling
Instanc
IAM > R
GitHub
us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles/create?commonUseCase=EC2&policies=arn%3Aaws%3Aiam%3A%3Aaws%3Apolicy%2FAdministratorAccess&...
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
Global
cloudUserLuis @ 5501-4694-3653
Step 2: Add permissions
Edit
Permissions policy summary
Policy name
Type
Attached as
AdministratorAccess
AWS managed - job function
Permissions policy
Tags
Add tags - optional
Info
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.
No tags associated with the resource.
Add tag
You can add up to 50 more tags.
Cancel
Previous
Create role
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
76°F
ENG
US
12:09 PM
6/23/2023

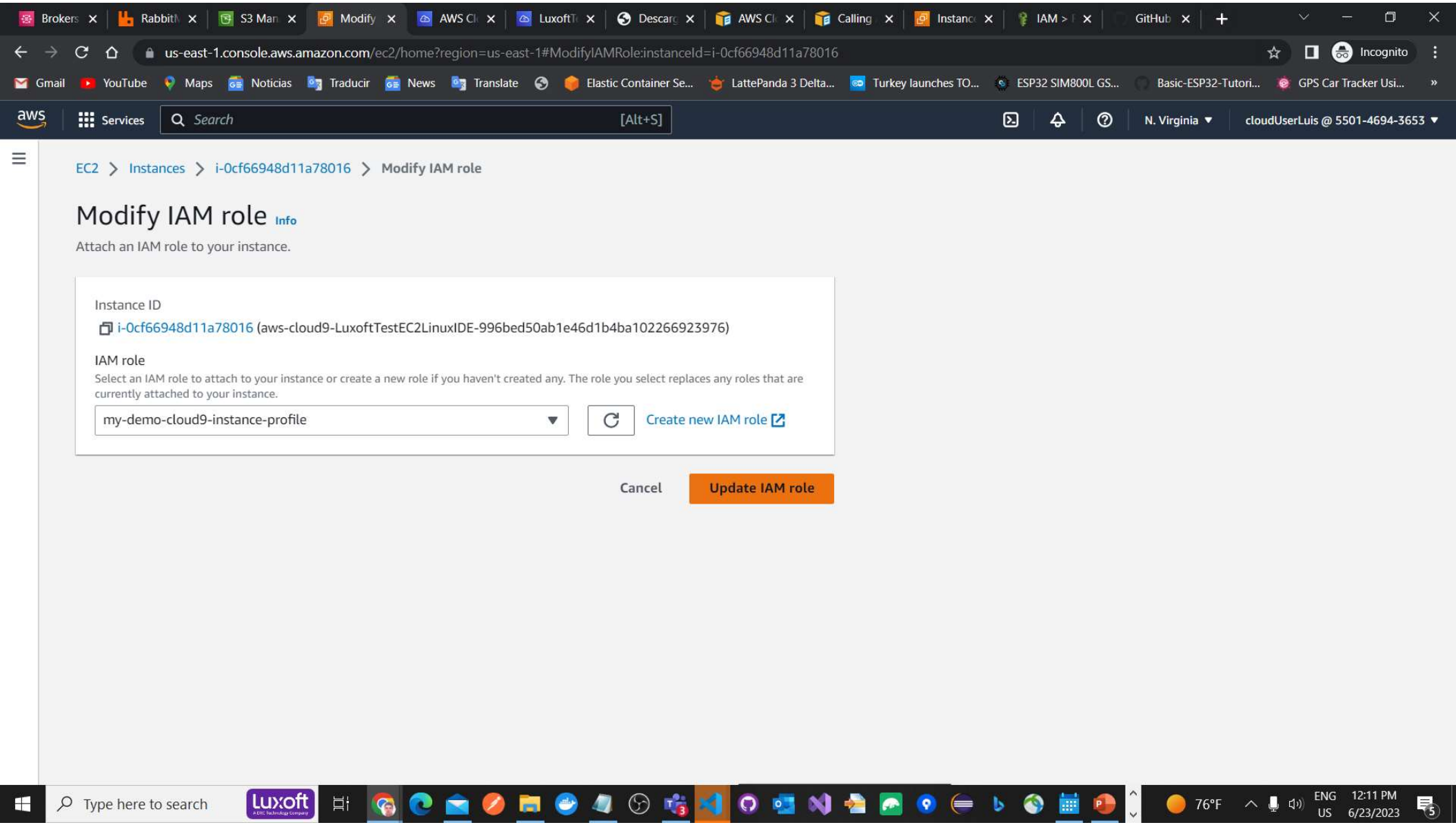

us-east-1.console.aws.amazon.com/iamv2/home?region=us-east-1#/roles
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
Global
cloudUserLuis @ 5501-4694-3653
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Access reports
Access analyzer
Archive rules
Analyzers
Settings
Credential report
Organization activity
Role my-demo-cloud9-instance-profile created.
View role
IAM > Roles
Roles (30) Info
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.
Delete
Create role
my
4 matches
1
Role name
Trusted entities
Last activity
my-demo-cloud9-instance-profile
AWS Service: ec2
1 hour ago
my-function-role-fbomtqtg
AWS Service: lambda
-
my-function-role-gftv1utw
AWS Service: lambda
218 days ago
my-function-role-qq406q7r
AWS Service: lambda
220 days ago
Roles Anywhere Info
Authenticate your non AWS workloads and securely provide access to AWS services.
Manage
CloudShell
Feedback
Language
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
76°F
ENG
US
12:10 PM
6/23/2023

Attach the new Role to the EC2 virtual machine where is hosted the AWS Cloud9 IDE




Brokers
RabbitM
S3 Man
Instanc
AWS Cl
LuxoftT
Descarg
AWS Cl
Calling
Instanc
IAM > F
GitHub
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances:v=3;$case=tags:true%5C,client:false;$regex=tags:false%5C,client:false
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
New EC2 Experience
Tell us what you think
EC2 Dashboard
EC2 Global View
Events
Limits
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Scheduled Instances
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Instances (1/1) Info
Connect
Instance state
Actions
Launch instances
Find instance by attribute or tag (case-sensitive)
Name
Instance ID
Instance state
Instance type
Status check
Zone
aws-cloud9-LuxoftTestEC2LinuxI...
i-0cf66948d11a78016
Running
t2.micro
2/2 checks
Connect
View details
Manage instance state
Instance settings
Networking
Security
Image and templates
Monitor and troubleshoot
Change security groups
Get Windows password
Modify IAM role
Instance: i-0cf66948d11a78016 (aws-cloud9-LuxoftTestEC2LinuxIDE-996bed50ab1e46d1b4ba102266923976)
Details
Security
Networking
Storage
Status checks
Monitoring
Tags
Instance summary Info
Instance ID
i-0cf66948d11a78016 (aws-cloud9-LuxoftTestEC2LinuxIDE-996bed50ab1e46d1b4ba102266923976)
Public IPv4 address
54.173.204.99 | open address
Private IPv4 addresses
172.31.83.167
IPv6 address
–
Instance state
Running
Public IPv4 DNS
ec2-54-173-204-99.compute-1.amazonaws.com | open address
Hostname type
Private IP DNS name (IPv4 only)
Type here to search
76°F
ENG
US
12:11 PM
6/23/2023


us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#ModifyIAMRole:instanceId=i-0cf66948d11a78016
Incognito
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
EC2 > Instances > i-0cf66948d11a78016 > Modify IAM role
Modify IAM role Info
Attach an IAM role to your instance.
Instance ID
i-0cf66948d11a78016 (aws-cloud9-LuxoftTestEC2LinuxIDE-996bed50ab1e46d1b4ba102266923976)
IAM role
Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.
AWSCloud9SSMInstanceProfile
Create new IAM role
No IAM Role
Choose this option to detach an IAM role
AWSCloud9SSMInstanceProfile
arn:aws:iam::550146943653:instance-profile/cloud9/AWSCloud9SSMInstanceProfile
my-demo-cloud9-instance-profile
arn:aws:iam::550146943653:instance-profile/my-demo-cloud9-instance-profile
my-demo-cloud9-instance-profile
Cancel
Update IAM role
Type here to search
76°F
ENG
US
12:11 PM
6/23/2023

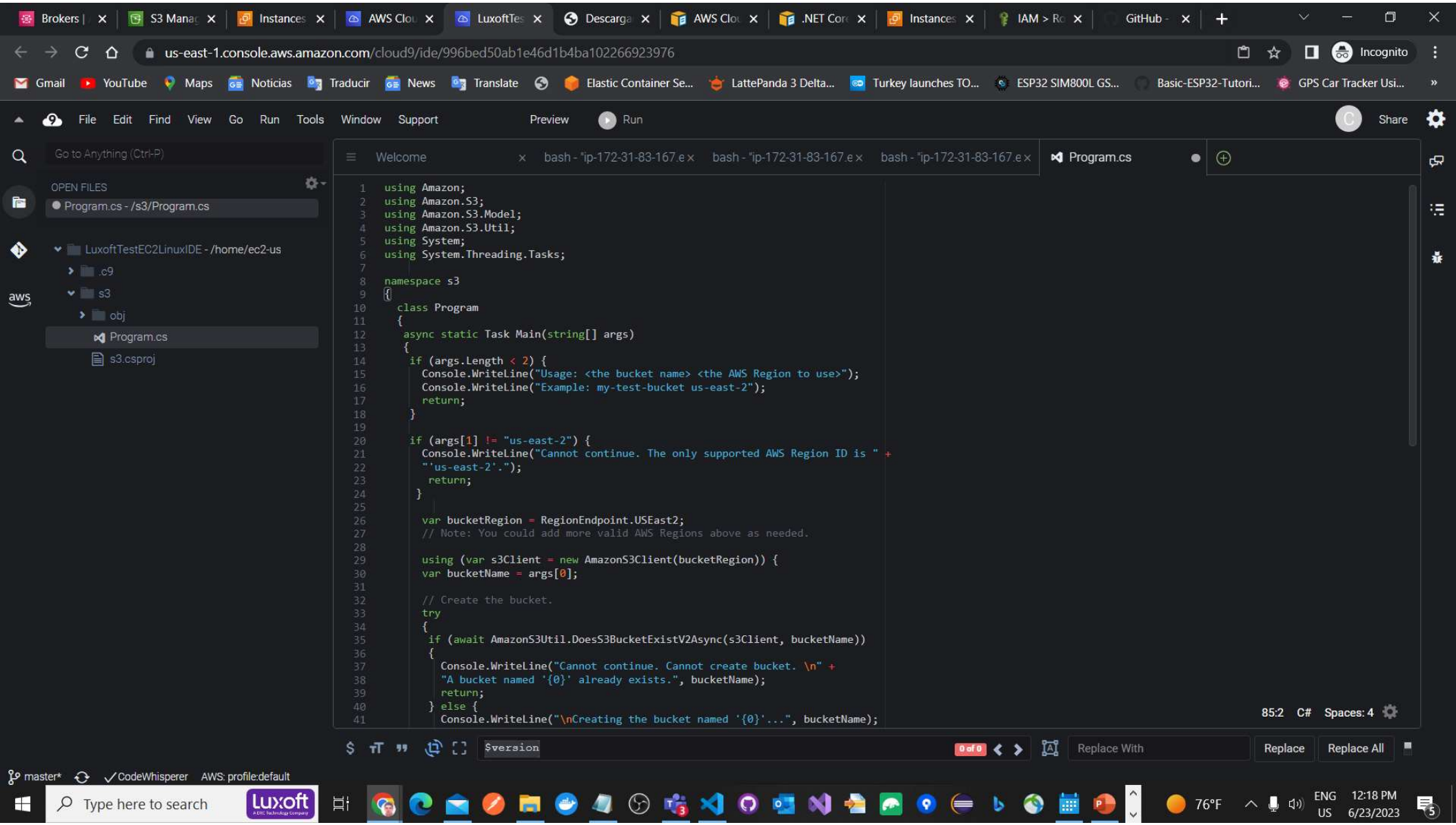

Brokers
RabbitM
S3 Man
Modify
AWS Cl
LuxoftT
Descarg
AWS Cl
Calling
Instanc
IAM > F
GitHub
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#ModifyIAMRole:instanceId=i-0cf66948d11a78016
Incognito
Gmail
YouTube
Maps
Noticias
Traducir
News
Translate
Elastic Container Se...
LattePanda 3 Delta...
Turkey launches TO...
ESP32 SIM800L GS...
Basic-ESP32-Tutori...
GPS Car Tracker Usi...
aws
Services
Search
[Alt+S]
N. Virginia
cloudUserLuis @ 5501-4694-3653
EC2 > Instances > i-0cf66948d11a78016 > Modify IAM role
Modify IAM role Info
Attach an IAM role to your instance.
Instance ID
i-0cf66948d11a78016 (aws-cloud9-LuxoftTestEC2LinuxIDE-996bed50ab1e46d1b4ba102266923976)
IAM role
Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.
my-demo-cloud9-instance-profile
Create new IAM role
Cancel
Update IAM role
Type here to search
76°F
ENG
US
12:11 PM
6/23/2023

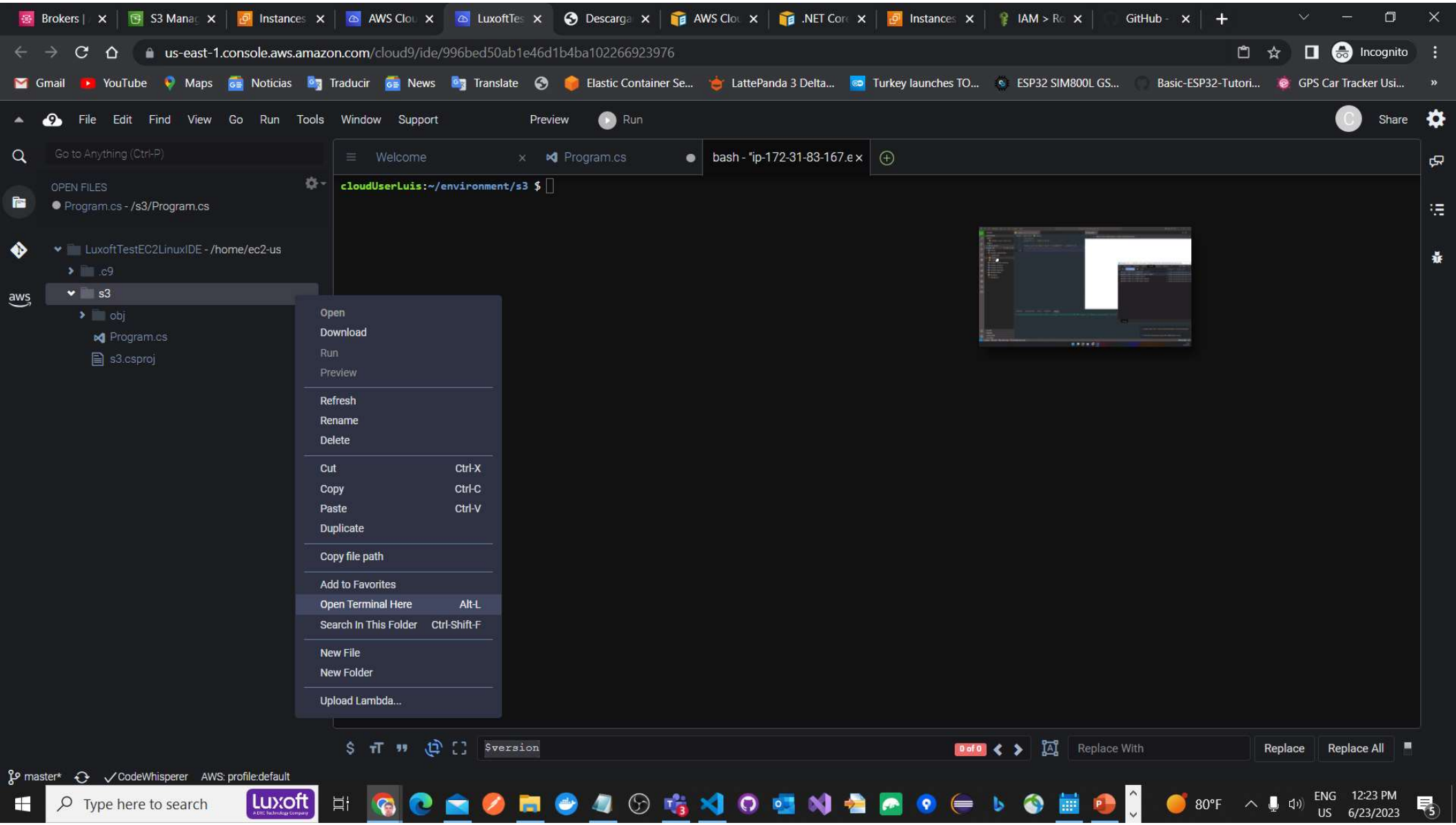
Step 7: Add AWS SDK code

In this step, you add code to interact with Amazon S3 to create a bucket, delete the bucket you just created, and then list your available buckets.

From the Environment window in the AWS Cloud9 IDE, open the s3/Program.cs file. In the editor, replace the file's current contents with the following code, and then save the Program.cs file.

```
using Amazon;
using Amazon.S3;
using Amazon.S3.Model;
using Amazon.S3.Util;
using System;
using System.Threading.Tasks;

namespace s3
{
 class Program
 {
  async static Task Main(string[] args)
  {
   if (args.Length < 2) {
    Console.WriteLine("Usage: <the bucket name> <the AWS Region to use>");
    Console.WriteLine("Example: my-test-bucket us-east-2");
    return;
   }

   if (args[1] != "us-east-2") {
    Console.WriteLine("Cannot continue. The only supported AWS Region ID is " +
    "'us-east-2'.");
     return;
   }

    var bucketRegion = RegionEndpoint.USEast2;
    // Note: You could add more valid AWS Regions above as needed.

    using (var s3Client = new AmazonS3Client(bucketRegion)) {
    var bucketName = args[0];

    // Create the bucket.
    try
    {
```


us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
OPEN FILES
Program.cs - /s3/Program.cs
LuxoftTestEC2LinuxIDE - /home/ec2-us
.c9
s3
obj
Program.cs
s3.csproj
Welcome
bash - "ip-172-31-83-167.e
bash - "ip-172-31-83-167.e
bash - "ip-172-31-83-167.e
Program.cs
using Amazon;
using Amazon.S3;
using Amazon.S3.Model;
using Amazon.S3.Util;
using System;
using System.Threading.Tasks;
namespace s3
{
class Program
{
async static Task Main(string[] args)
{
if (args.Length < 2) {
Console.WriteLine("Usage: <the bucket name> <the AWS Region to use>");
Console.WriteLine("Example: my-test-bucket us-east-2");
return;
}
if (args[1] != "us-east-2") {
Console.WriteLine("Cannot continue. The only supported AWS Region ID is " +
"'us-east-2'.");
return;
}
var bucketRegion = RegionEndpoint.USEast2;
// Note: You could add more valid AWS Regions above as needed.
using (var s3Client = new AmazonS3Client(bucketRegion)) {
var bucketName = args[0];
// Create the bucket.
try
{
if (await AmazonS3Util.DoesS3BucketExistV2Async(s3Client, bucketName))
{
Console.WriteLine("Cannot continue. Cannot create bucket. \n" +
"A bucket named '{0}' already exists.", bucketName);
return;
} else {
Console.WriteLine("\nCreating the bucket named '{0}'...", bucketName);
85:2 C# Spaces: 4
$version
0 of 0
Replace With
Replace
Replace All
master*
CodeWhisperer
AWS: profile:default
Type here to search
76°F
ENG US
12:18 PM 6/23/2023


us-east-1.console.aws.amazon.com/cloud9/ide/996bed50ab1e46d1b4ba102266923976
File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
OPEN FILES
Program.cs - /s3/Program.cs
LuxoftTestEC2LinuxIDE - /home/ec2-us
.c9
s3
obj
Program.cs
s3.csproj
Welcome
Program.cs
bash - "ip-172-31-83-167.e
cloudUserLuis:~/environment/s3 $
Open
Download
Run
Preview
Refresh
Rename
Delete
Cut Ctrl-X
Copy Ctrl-C
Paste Ctrl-V
Duplicate
Copy file path
Add to Favorites
Open Terminal Here Alt-L
Search In This Folder Ctrl-Shift-F
New File
New Folder
Upload Lambda...
$version
0 of 0
Replace With
Replace
Replace All
master*
CodeWhisperer
AWS: profile:default
Type here to search
80°F
ENG US
12:23 PM 6/23/2023

For running the application run the command:
dotnet run my-bucket-luiscocoenriquez us-east-2