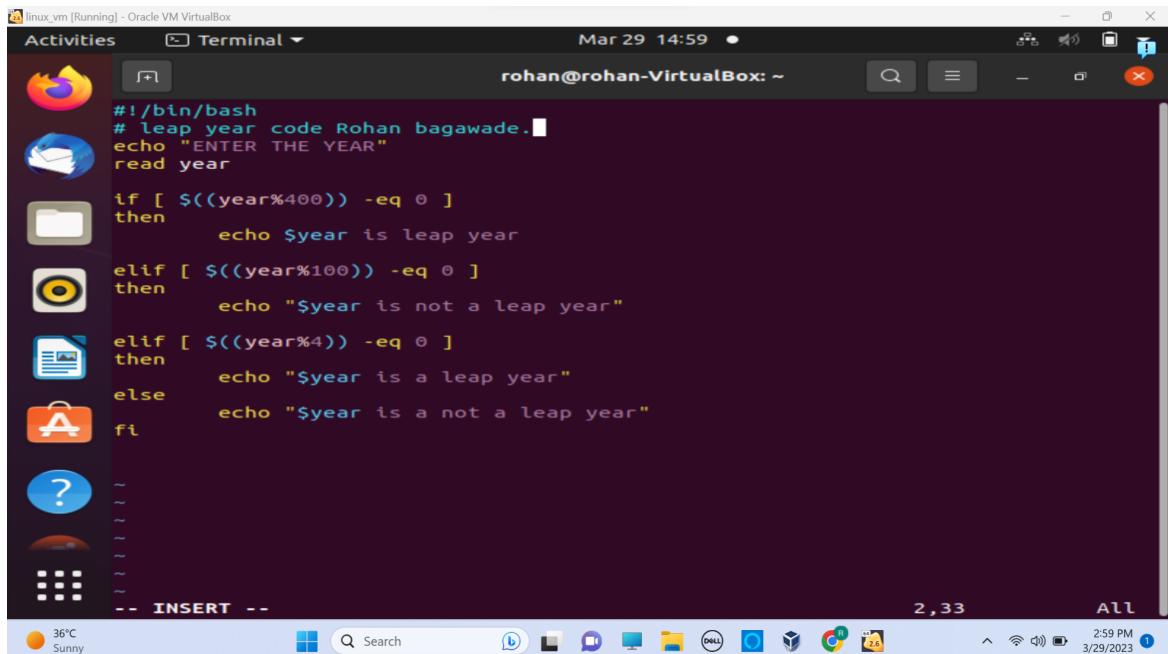


1) Write a shell program to find if the inputted year is leap or not.

ans=

```
#!/bin/bash
echo "ENTER THE YEAR"
read year
if [ $((year%400)) -eq 0 ]
then
    echo "$year is a leap year"
elif [ $((year%100)) -eq 0 ]
then
    echo "year is not a leap year"
elif [ $((year%4)) -eq 0 ]
then
    echo "$year is a leap year"
else
    echo "$year is not a leap year"
fi
```

CODE=



The screenshot shows a terminal window titled "Terminal" with the command "rohan@rohan-VirtualBox: ~". The window displays the following code:

```
#!/bin/bash
# leap year code Rohan bagawade.
echo "ENTER THE YEAR"
read year
if [ $((year%400)) -eq 0 ]
then
    echo $year is leap year
elif [ $((year%100)) -eq 0 ]
then
    echo "$year is not a leap year"
elif [ $((year%4)) -eq 0 ]
then
    echo "$year is a leap year"
else
    echo "$year is not a leap year"
fi
```

The terminal shows the output of the script being run, with the cursor at the end of the final "fi" statement. The desktop environment includes icons for various applications like a browser, file manager, and system settings, along with a weather widget showing "36°C Sunny".

OUTPUT=

The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "rohan@rohan-VirtualBox: ~". The terminal content displays the execution of a shell script named "leap_year.sh". The script prompts the user to enter a year and then outputs whether it is a leap year or not. The user enters several years, and the script correctly identifies 2024 as a leap year and others as not being leap years.

```
rohan@rohan-VirtualBox:~$ ./leap_year.sh
ENTER THE YEAR
2022
2022 is a not a leap year
rohan@rohan-VirtualBox:~$ ./leap_year.sh
ENTER THE YEAR
2023
2023 is a not a leap year
rohan@rohan-VirtualBox:~$ ./leap_year.sh
ENTER THE YEAR
2024
2024 is a leap year
rohan@rohan-VirtualBox:~$ ./leap_year.sh
ENTER THE YEAR
2032
2032 is a leap year
rohan@rohan-VirtualBox:~$ ./leap_year.sh
ENTER THE YEAR
2020
2020 is a leap year
rohan@rohan-VirtualBox:~$ ./leap_year.sh
ENTER THE YEAR
2019
2019 is a not a leap year
rohan@rohan-VirtualBox:~$ ./leap_year.sh
ENTER THE YEAR
2030
2030 is a not a leap year
rohan@rohan-VirtualBox:~$
```

Q.2 create a user and add user into group

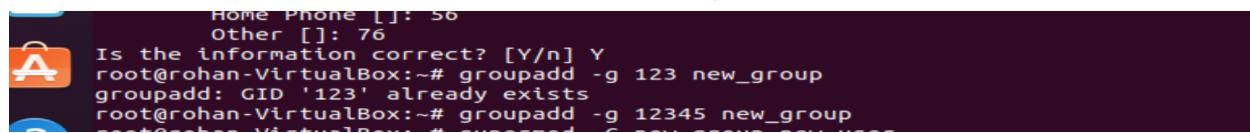
ans=

- Adding a user with name = new_user
- Command = Adduser new_user

The screenshot shows a Linux desktop environment with a terminal window open, run as root. The terminal window title is "root@rohan-VirtualBox: ~". The root user runs the "adduser" command to create a new user account named "new_user". The process involves setting a password, entering user details like full name, room number, work phone, home phone, and other information, and confirming the setup.

```
rohan@rohan-VirtualBox:~/root$ sudo -i
root@rohan-VirtualBox:~# adduser new_user
Adding user `new_user' ...
Adding new group `new_user' (1004) ...
Adding new user `new_user' (1004) with group `new_user' ...
Creating home directory `/home/new_user' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for new_user
Enter the new value, or press ENTER for the default
    Full Name []: new user
    Room Number []: 23
    Work Phone []: 34
    Home Phone []: 56
    Other []: 76
Is the information correct? [Y/n] Y
root@rohan-VirtualBox:~#
```

- ADDING THE GROUP WITH NAME = new_group
- Command = Groupadd -g 12345 new_group

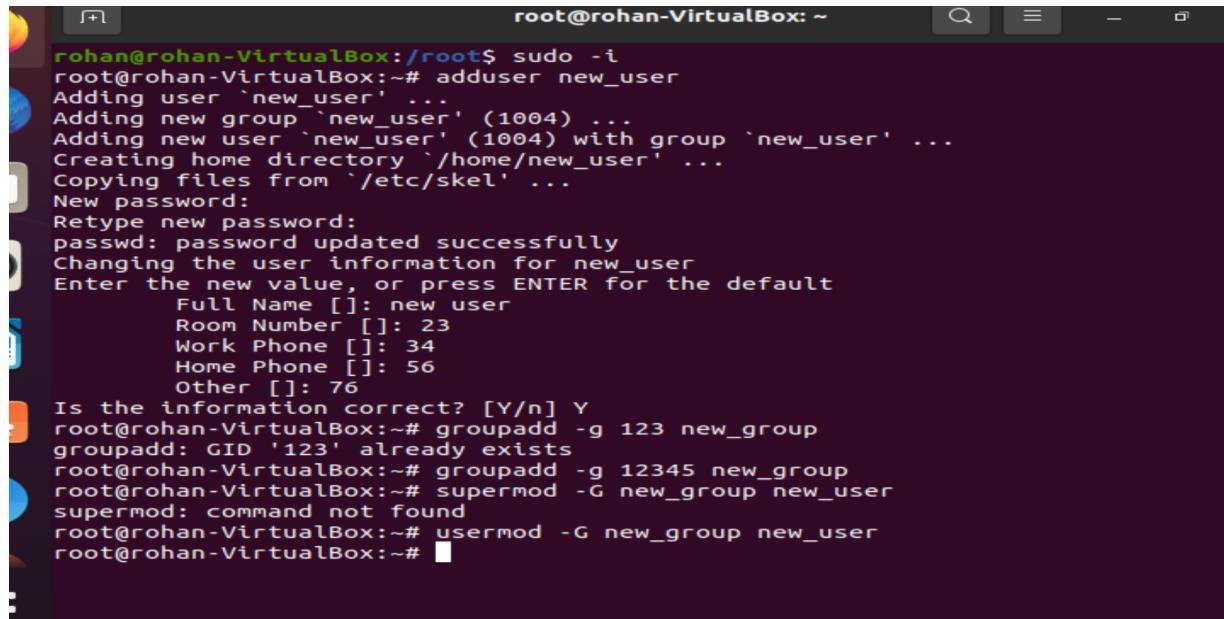


```

        HOME Phone []: 56
        Other []: 76
A Is the information correct? [Y/n] Y
root@rohan-VirtualBox:~# groupadd -g 123 new_group
groupadd: GID '123' already exists
root@rohan-VirtualBox:~# groupadd -g 12345 new_group
root@rohan-VirtualBox:~#

```

- Adding new user to a new group
- Command = usermod -G new_group new_user

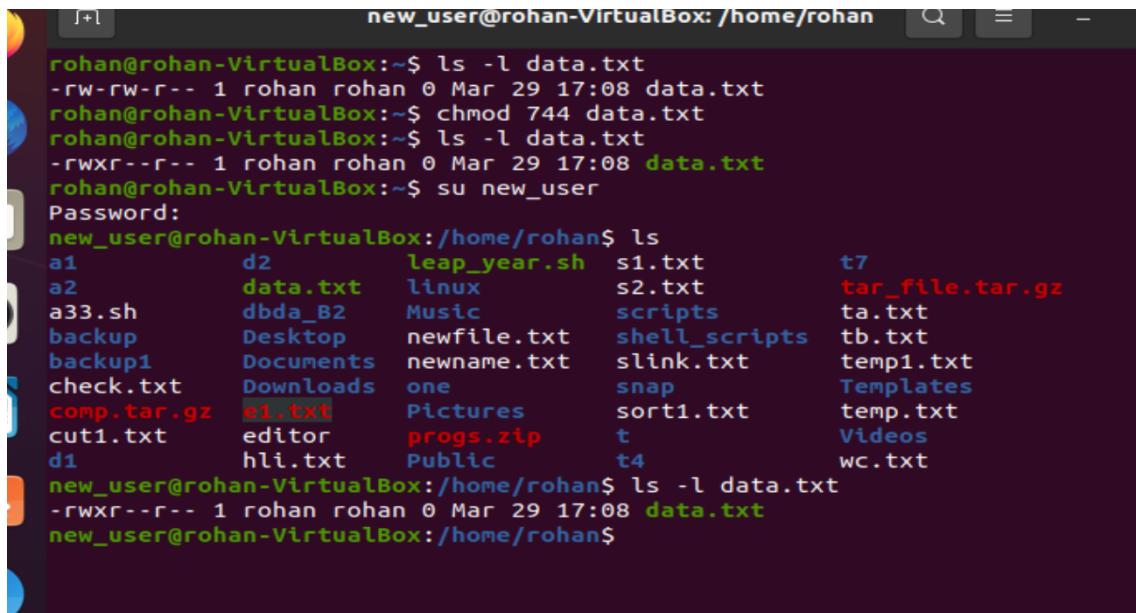


```

root@rohan-VirtualBox:~# sudo -i
root@rohan-VirtualBox:~# adduser new_user
Adding user `new_user' ...
Adding new group `new_user' (1004) ...
Adding new user `new_user' (1004) with group `new_user' ...
Creating home directory `/home/new_user' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for new_user
Enter the new value, or press ENTER for the default
      Full Name []: new user
      Room Number []: 23
      Work Phone []: 34
      Home Phone []: 56
      Other []: 76
Is the information correct? [Y/n] Y
root@rohan-VirtualBox:~# groupadd -g 123 new_group
groupadd: GID '123' already exists
root@rohan-VirtualBox:~# groupadd -g 12345 new_group
root@rohan-VirtualBox:~# supermod -G new_group new_user
supermod: command not found
root@rohan-VirtualBox:~# usermod -G new_group new_user
root@rohan-VirtualBox:~#

```

- Creating a file named data.txt
- Changing the permissions to
- group/owner = rwx
- User = r
- Others = r



```
rohan@rohan-VirtualBox:~$ ls -l data.txt
-rw-rw-r-- 1 rohan rohan 0 Mar 29 17:08 data.txt
rohan@rohan-VirtualBox:~$ chmod 744 data.txt
rohan@rohan-VirtualBox:~$ ls -l data.txt
-rwxr--r-- 1 rohan rohan 0 Mar 29 17:08 data.txt
rohan@rohan-VirtualBox:~$ su new_user
Password:
new_user@rohan-VirtualBox:/home/rohan$ ls
a1          d2      leap_year.sh  s1.txt      t7
a2          data.txt  linux        s2.txt      tar_file.tar.gz
a33.sh      dbda_B2  Music        scripts    ta.txt
backup      Desktop   newfile.txt  shell_scripts tb.txt
backup1     Documents  newname.txt slink.txt  temp1.txt
check.txt   Downloads  one         snap       Templates
comp.tar.gz el.txt    Pictures    sort1.txt  temp.txt
cut1.txt    editor     progs.zip   t          Videos
d1          hli.txt   Public      t4        wc.txt
new_user@rohan-VirtualBox:/home/rohan$ ls -l data.txt
-rwxr--r-- 1 rohan rohan 0 Mar 29 17:08 data.txt
new_user@rohan-VirtualBox:/home/rohan$
```

Q.3) create a directory with name dbda

Ans =

Creation of directory= Mkdir dbda

Creation of file = Cd dbda && touch Sep.txt

Rename of file = Mv Sep.txt Batch2.txt

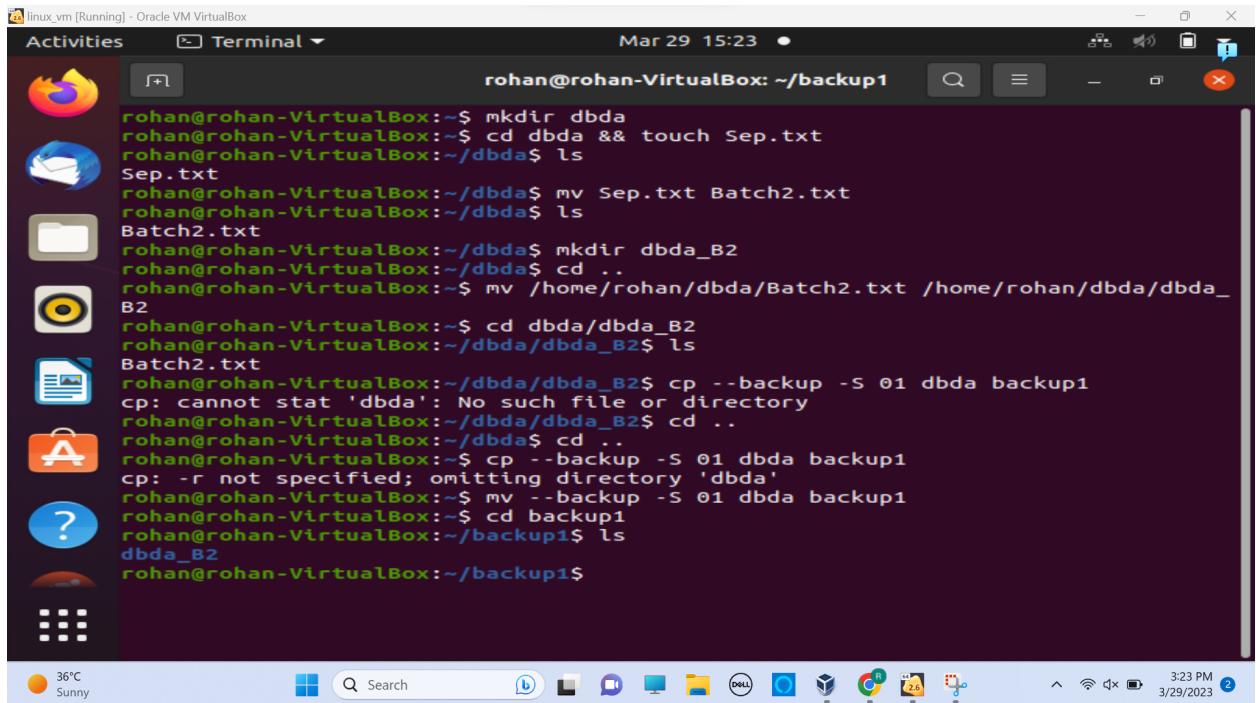
Creation of directory= Mkdir dbda_B2

Cd ..

Moving file to anot. = Move /home/rohan/dbda/Batch2.txt /home/rohan/dbda/dbda_B2

Cd dbda/dbda_B2

Creation of backup = Mv –backup -S 01 dbda backup1



The screenshot shows a Linux desktop environment with a terminal window open. The terminal window title is "Terminal" and the date and time are "Mar 29 15:23". The terminal content shows the following command sequence:

```
rohan@rohan-VirtualBox:~/backup1$ mkdir dbda
rohan@rohan-VirtualBox:~$ cd dbda && touch Sep.txt
rohan@rohan-VirtualBox:~/dbda$ ls
Sep.txt
rohan@rohan-VirtualBox:~/dbda$ mv Sep.txt Batch2.txt
rohan@rohan-VirtualBox:~/dbda$ ls
Batch2.txt
rohan@rohan-VirtualBox:~/dbda$ mkdir dbda_B2
rohan@rohan-VirtualBox:~/dbda$ cd ..
rohan@rohan-VirtualBox:~$ mv /home/rohan/dbda/Batch2.txt /home/rohan/dbda/dbda_B2
rohan@rohan-VirtualBox:~/dbda$ cd dbda_B2
rohan@rohan-VirtualBox:~/dbda_B2$ ls
Batch2.txt
rohan@rohan-VirtualBox:~/dbda_B2$ cp --backup -S 01 dbda backup1
cp: cannot stat 'dbda': No such file or directory
rohan@rohan-VirtualBox:~/dbda_B2$ cd ..
rohan@rohan-VirtualBox:~$ cp --backup -S 01 dbda backup1
cp: -r not specified; omitting directory 'dbda'
rohan@rohan-VirtualBox:~$ mv --backup -S 01 dbda backup1
rohan@rohan-VirtualBox:~$ cd backup1
rohan@rohan-VirtualBox:~/backup1$ ls
dbda_B2
rohan@rohan-VirtualBox:~/backup1$
```

The desktop interface includes a dock with various icons like a browser, file manager, and system tools. The status bar at the bottom shows the date and time as "3/29/2023 3:23 PM".

QUE. 4) CREATE A LINUX EC2

Ans = public ip address = <http://65.2.175.243/>

- 1) Creating an ec2 instance

The screenshot shows the AWS EC2 Instances "Launch an instance" page. At the top, there is a success message: "Successfully initiated launch of instance (i-0fa5c8bde51705909)". Below this, there is a "Next Steps" section with several options:

- Create billing and free tier usage alerts
- Connect to your instance
- Connect an RDS database
- Create EBS snapshot policy

At the bottom of the page, there is a feedback link, language selection, and a footer with copyright information and links to Privacy, Terms, and Cookie preferences.

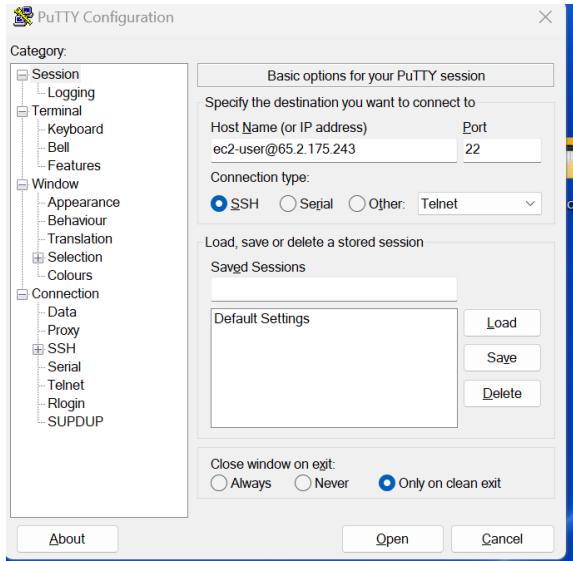
2) Instance status

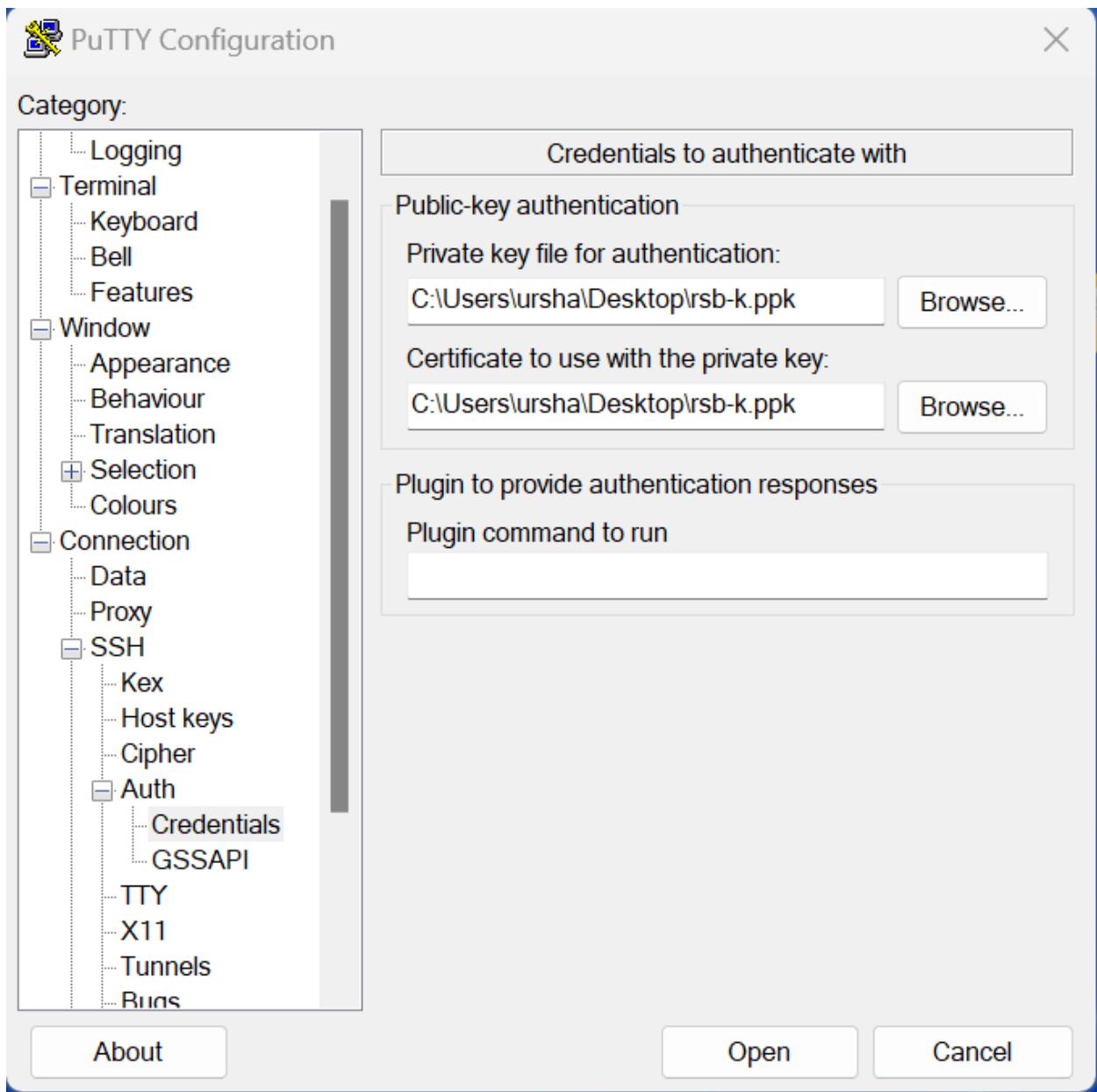
The screenshot shows the AWS EC2 Instances "Instance details" page for the instance i-0fa5c8bde51705909. The page title is "Instance summary for i-0fa5c8bde51705909 (Rohan_EC2_Instance) Info". The instance state is shown as "Running". Other details include:

Attribute	Value
Instance ID	i-0fa5c8bde51705909 (Rohan_EC2_Instance)
Public IPv4 address	65.2.175.243 open address
Private IPv4 addresses	172.31.40.170
Public IPv4 DNS	ec2-65-2-175-243.ap-south-1.compute.amazonaws.com open address
IPv6 address	-
Instance state	Running
Hostname type	
IP name: ip-172-31-40-170.ap-south-1.compute.internal	
Private IP DNS name (IPv4 only)	ip-172-31-40-170.ap-south-1.compute.internal
Answer private resource DNS name	
IPv4 (A)	
Instance type	t2.micro
Elastic IP addresses	-
Auto-assigned IP address	65.2.175.243 [Public IP]
VPC ID	vpc-06af5d3a1c604d161

At the bottom of the page, there is a feedback link, language selection, and a footer with copyright information and links to Privacy, Terms, and Cookie preferences.

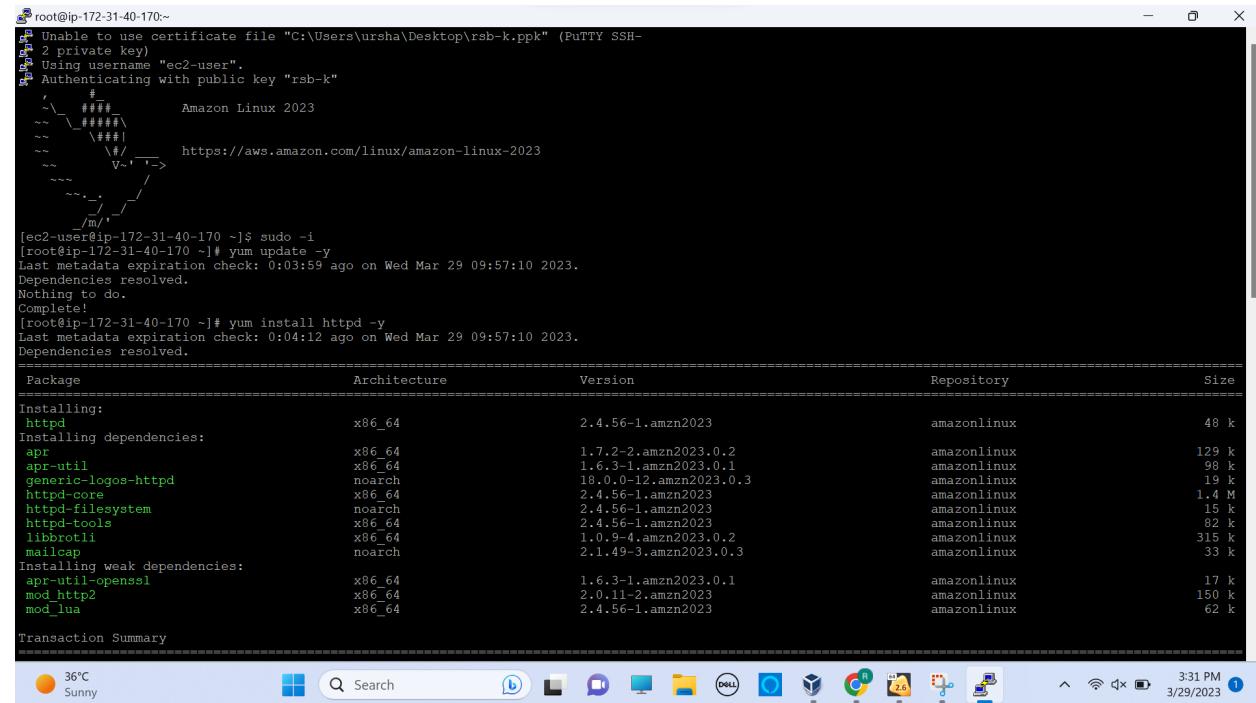
3) Inserting public ip into putty





```
=sudo -i
```

```
=yum update -y
```

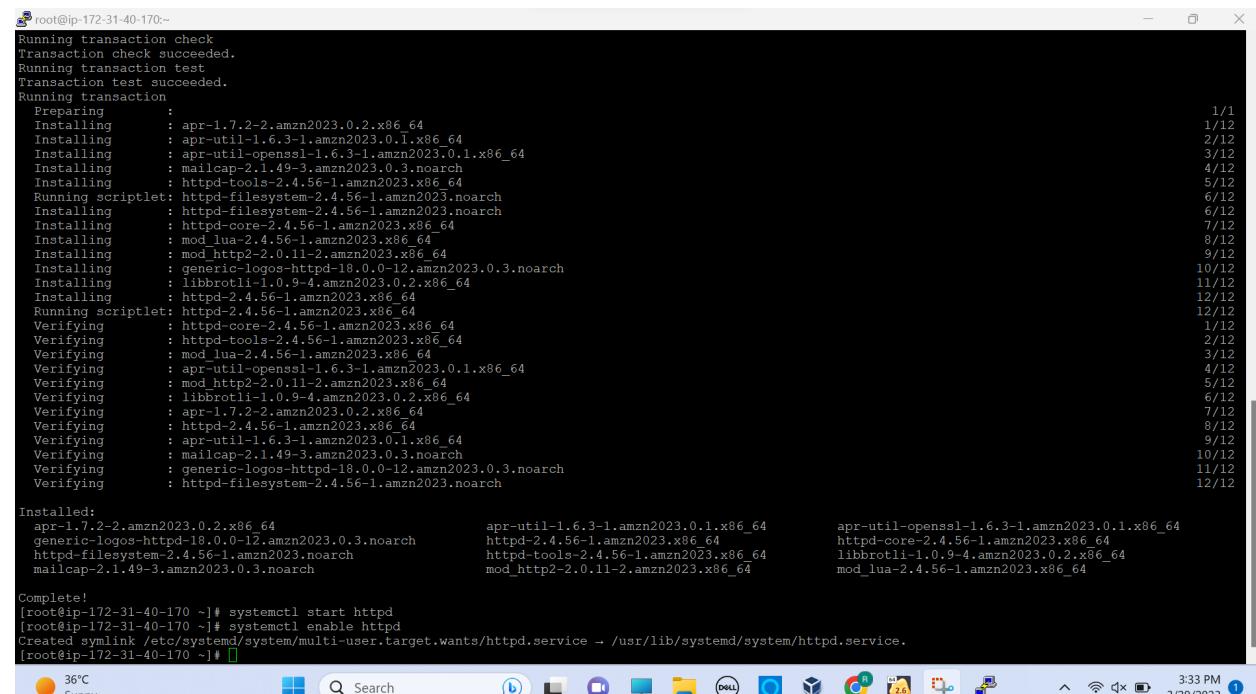


```
[root@ip-172-31-40-170 ~]# Unable to use certificate file "C:\Users\ursha\Desktop\rsb-k.ppk" (PuTTY SSH-2 private key)
[root@ip-172-31-40-170 ~]# Using username "ec2-user".
[root@ip-172-31-40-170 ~]# Authenticating with public key "rsb-k"
[ec2-user@ip-172-31-40-170 ~]$ sudo -i
[root@ip-172-31-40-170 ~]# yum update -y
Last metadata expiration check: 0:03:59 ago on Wed Mar 29 09:57:10 2023.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-40-170 ~]# yum install httpd -y
Last metadata expiration check: 0:04:12 ago on Wed Mar 29 09:57:10 2023.
Dependencies resolved.

Transaction Summary
-----
```

Package	Architecture	Version	Repository	Size
Installing:				
httpd	x86_64	2.4.56-1.amzn2023	amazonlinux	48 k
Installing dependencies:				
apr	x86_64	1.7.2-2.amzn2023.0.2	amazonlinux	129 k
apr-util	x86_64	1.6.3-1.amzn2023.0.1	amazonlinux	98 k
generic-logos-httd	noarch	18.0.0-12.amzn2023.0.3	amazonlinux	19 k
httpd-core	x86_64	2.4.56-1.amzn2023	amazonlinux	1.4 M
httpd-filesystem	noarch	2.4.56-1.amzn2023	amazonlinux	15 k
httpd-tools	x86_64	2.4.56-1.amzn2023	amazonlinux	82 k
libbrotli	x86_64	1.0.9-4.amzn2023.0.2	amazonlinux	315 k
mailcap	noarch	2.1.49-3.amzn2023.0.3	amazonlinux	33 k
Installing weak dependencies:				
apr-util-openssl	x86_64	1.6.3-1.amzn2023.0.1	amazonlinux	17 k
mod_http2	x86_64	2.0.11-2.amzn2023	amazonlinux	150 k
mod_lua	x86_64	2.4.56-1.amzn2023	amazonlinux	62 k

```
Transaction Summary
-----
```



```
[root@ip-172-31-40-170 ~]# Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing : 1/12
Installing : apr-1.7.2-2.amzn2023.0.2.x86_64 1/12
Installing : apr-util-1.6.3-1.amzn2023.0.1.x86_64 2/12
Installing : generic-logos-httd-1.6.3-1.amzn2023.0.1.x86_64 3/12
Installing : httpd-filesystem-2.4.56-1.amzn2023.noarch 4/12
Installing : httpd-tools-2.4.56-1.amzn2023.x86_64 5/12
Running scriptlet: httpd-filesystem-2.4.56-1.amzn2023.noarch 6/12
Installing : httpd-core-2.4.56-1.amzn2023.x86_64 6/12
Installing : mod_lua-2.4.56-1.amzn2023.x86_64 7/12
Installing : mod_http2-2.0.11-2.amzn2023.x86_64 8/12
Installing : generic-logos-httd-18.0.0-12.amzn2023.0.3.noarch 9/12
Installing : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 10/12
Installing : httpd-2.4.56-1.amzn2023.x86_64 11/12
Running scriptlet: httpd-2.4.56-1.amzn2023.x86_64 12/12
Verifying : httpd-core-2.4.56-1.amzn2023.x86_64 1/12
Verifying : httpd-tools-2.4.56-1.amzn2023.x86_64 2/12
Verifying : mod_lua-2.4.56-1.amzn2023.x86_64 3/12
Verifying : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64 4/12
Verifying : mod_http2-2.0.11-2.amzn2023.x86_64 5/12
Verifying : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 6/12
Verifying : apr-1.7.2-2.amzn2023.0.2.x86_64 7/12
Verifying : httpd-2.4.56-1.amzn2023.x86_64 8/12
Verifying : apr-util-1.6.3-1.amzn2023.0.1.x86_64 9/12
Verifying : generic-logos-httd-18.0.0-12.amzn2023.0.3.noarch 10/12
Verifying : httpd-filesystem-2.4.56-1.amzn2023.noarch 11/12
Verifying : mod_lua-2.4.56-1.amzn2023.x86_64 12/12

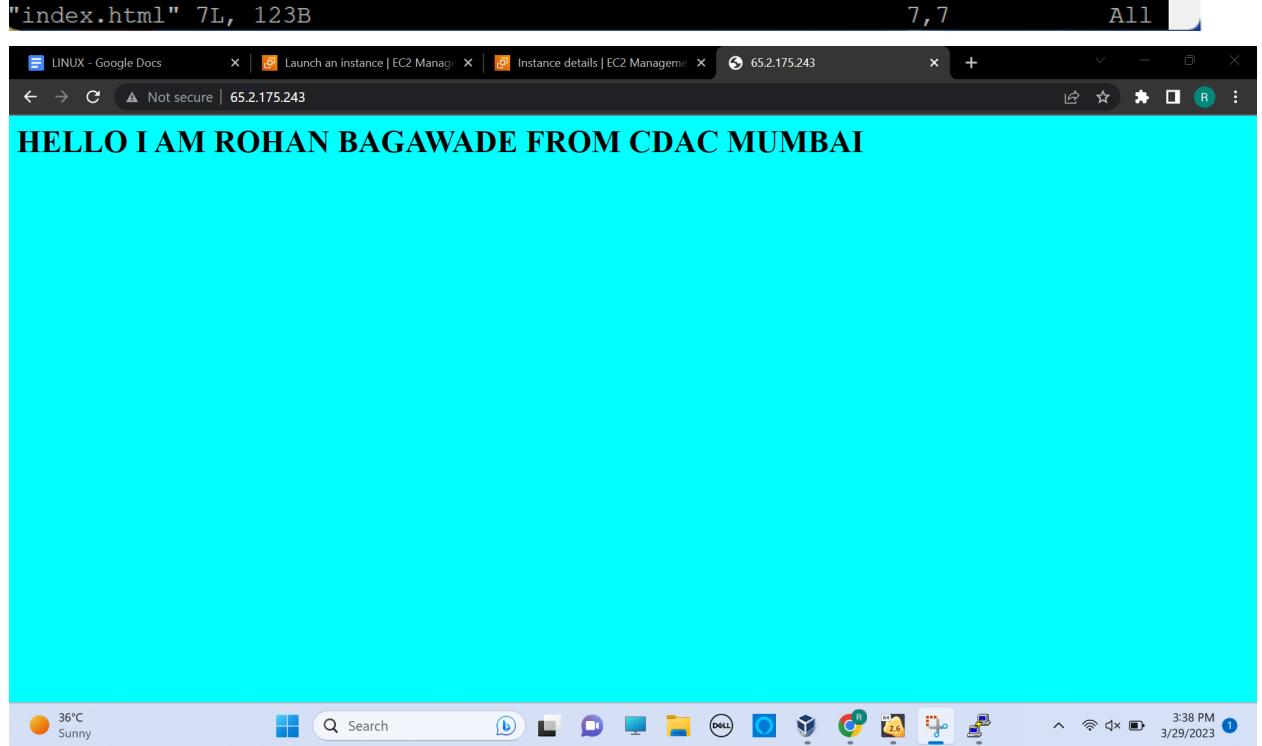
Installed:
  apr-1.7.2-2.amzn2023.0.2.x86_64
  generic-logos-httd-18.0.0-12.amzn2023.0.3.noarch
  httpd-filesystem-2.4.56-1.amzn2023.noarch
  mailcap-2.1.49-3.amzn2023.0.3.noarch

april-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
httpd-core-2.4.56-1.amzn2023.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mod_lua-2.4.56-1.amzn2023.x86_64

[root@ip-172-31-40-170 ~]# systemctl start httpd
[root@ip-172-31-40-170 ~]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-40-170 ~]#
```

 root@ip-172-31-40-170:/var/www/html
[root@ip-172-31-40-170 ~]# cd /var/www/html
[root@ip-172-31-40-170 html]# vi index.html
[root@ip-172-31-40-170 html]# █

```
root@ip-172-31-40-170:/var/www/html
<html>
    <head>
        <body bgcolor="cyan">
            <h1> HELLO I AM ROHAN BAGAWADE FROM CDAC MUMBAI </h1>
        </body>
    </head>
</html>
~
~
~
```



Que. 5) TO CREATE A NEW IAM USER

ANS =

The screenshot shows two consecutive steps in the AWS IAM console.

Step 1: User Creation

The top half of the screenshot shows the "Users" page after creating a new user named "jane". A green success message at the top states: "User created successfully. You can view and download the user's password and email instructions for signing in to the AWS Management Console." Below the message, the "Users (1)" section displays the newly created user "jane". The user details show: User name "jane", Groups "None", Last activity "Never", MFA "None", and Password "None".

Step 2: Adding a User to a Group

The bottom half of the screenshot shows the "Name the group" step in the "Add users to the group" wizard. The user "jane" is selected for addition to a new group named "devops". The "User group name" field contains "devops". The "Add users to the group - Optional" section indicates "Selected 1/1" and shows the user "jane" listed with 0 groups, last activity "Never", and creation time "1 minute ago".

EC2

Identity and Access Management (IAM)

devops user group created.

View group

IAM > User groups

User groups (1) Info

A user group is a collection of IAM users. Use groups to specify permissions for a collection of users.

Filter User groups by property or group name and press enter

Create group

Group name	Users	Permissions	Creation time
devops	1	Defined	Now

Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

Access reports

Summary

Edit

User group name: devops

Creation time: March 29, 2023, 15:44 (UTC+05:30)

ARN: arn:aws:iam::216524373056:group/devops

Users Permissions Access Advisor

Permissions policies (1) Info

You can attach up to 10 managed policies.

Filter policies by property or policy name and press enter.

Simulate Remove Add permissions

Policy name	Type	Description
AmazonEC2FullAccess	AWS managed	Provides full access to Amazon EC2 via the AWS Manageme...

Access AMI

LINUX - Google Docs | Launch an instance | EC2 | IAM Management Console | IAM Policy Simulator | 65.2.175.243

pollicysim.aws.amazon.com/home/index.jsp#users/jane

IAM Policy Simulator

Mode : Existing Policies | rohansb

Policies

Selected user: jane

IAM Policies

Filter: AmazonEC2FullAccess

Custom IAM Policies

There are no policies to display!

Permissions Boundary Policy

You can simulate a maximum of one permissions boundary policy per user or role.

There are no policies to display!

Policy Simulator

Amazon EC2 | 3 Action(s) sele... | Select All | Deselect All | Reset Contexts | Clear Results | Run Simulation

Global Settings

Action Settings and Results [3 actions selected. 0 actions not simulated. 3 actions allowed. 0 actions denied.]

Service	Action	Resource Type	Simulation Resource	Permission
Amazon EC2	RunInstances	volume,image,su...	*	allowed 1 matching statements.
Amazon EC2	CreateInstanceEventWin...	instance-event-wi...	*	allowed 1 matching statements.
Amazon EC2	CreateInstanceExportTask	export-instance-t...	*	allowed 1 matching statements.

36°C Sunny | Search |                              | 3:50 PM | 3/29/2023