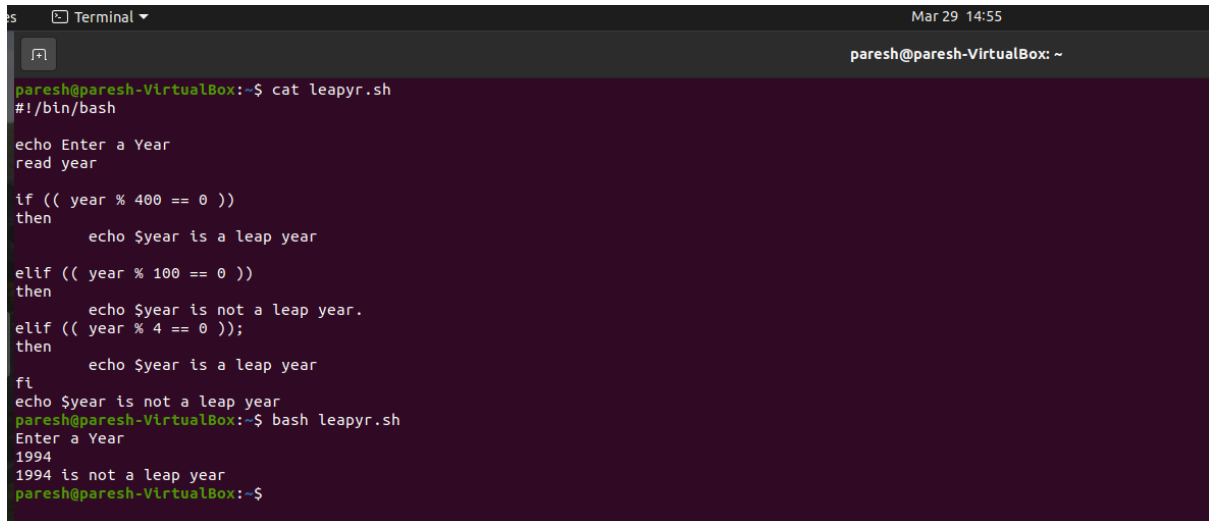


Shell

PRN 230340325030 Paresh Kisan Kadali

Q1.

1. Created one .sh file named leapyr.sh using vim editor.
2. The code is written as below in screenshot.
3. The script is giving accurate output.



```
paresh@paresh-VirtualBox:~$ cat leapyr.sh
#!/bin/bash

echo Enter a Year
read year

if (( year % 400 == 0 ))
then
    echo $year is a leap year
elif (( year % 100 == 0 ))
then
    echo $year is not a leap year.
elif (( year % 4 == 0 ));
then
    echo $year is a leap year
fi
echo $year is not a leap year
paresh@paresh-VirtualBox:~$ bash leapyr.sh
Enter a Year
1994
1994 is not a leap year
paresh@paresh-VirtualBox:~$
```

Q2. is on next page

Q2.

1. In root, using command addgroup and adduser, group developer and user parag have been created successfully (screenshot 1)
2. User parag has been added in group developer (Screenshot 2)
3. using touch command, sample.txt file is created which has the permissions as seen below in screenshot(Screenshot 3)
4. Using command chmod permissions are edited as asked.

```
Terminal Mar 29 15:00
root@paresh-VirtualBox:~# addgroup developer
Adding group 'developer' (GID 1007) ...
Done.
root@paresh-VirtualBox:~# adduser parag
Adding user 'parag' ...
Adding new group 'parag' (1008) ...
Adding new user 'parag' (1004) with group 'parag' ...
Creating home directory '/home/parag' ...
Copying files from '/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for parag
Enter the new value, or press ENTER for the default
  Full Name []: parag
    Room Number []:
    Work Phone []:
    Home Phone []:
      Other []:
Is the information correct? [Y/n] y
root@paresh-VirtualBox:~#
```

```
Terminal
root@paresh-VirtualBox:~# gpasswd -a parag developer
Adding user parag to group developer
root@paresh-VirtualBox:~#
```

```
parag@paresh-VirtualBox:/home/parag$ ls
sample.txt
parag@paresh-VirtualBox:/home/parag$ ls -ltr sample.txt
-rw-r--r-- 1 root root 0 Mar 29 15:09 sample.txt
parag@paresh-VirtualBox:/home/parag$ sudo chmod u-w sample.txt
parag@paresh-VirtualBox:/home/parag$ ls -ltr
total 0
-r--r--r-- 1 root root 0 Mar 29 15:09 sample.txt
parag@paresh-VirtualBox:/home/parag$
```

Q3. on next page

Q3.

1. Using mkdir command directory named dbda and dbda_B2 are created.
2. Using touch command text file is created.
3. Using mv command textfile is renamed.
4. Again using mv command it is moved to another directory.
5. Using cp -r command one copy of the same directory has been made recursively as a backup.

```
paresh@paresh-VirtualBox:~$ mkdir dbda
paresh@paresh-VirtualBox:~$ mkdir dbda_B2
paresh@paresh-VirtualBox:~$ cd dbda
paresh@paresh-VirtualBox:~/dbda$ touch Sep.txt
paresh@paresh-VirtualBox:~/dbda$ ls
Sep.txt
paresh@paresh-VirtualBox:~/dbda$ mv Sep.txt Batch2.txt
paresh@paresh-VirtualBox:~/dbda$ ls
Batch2.txt
```

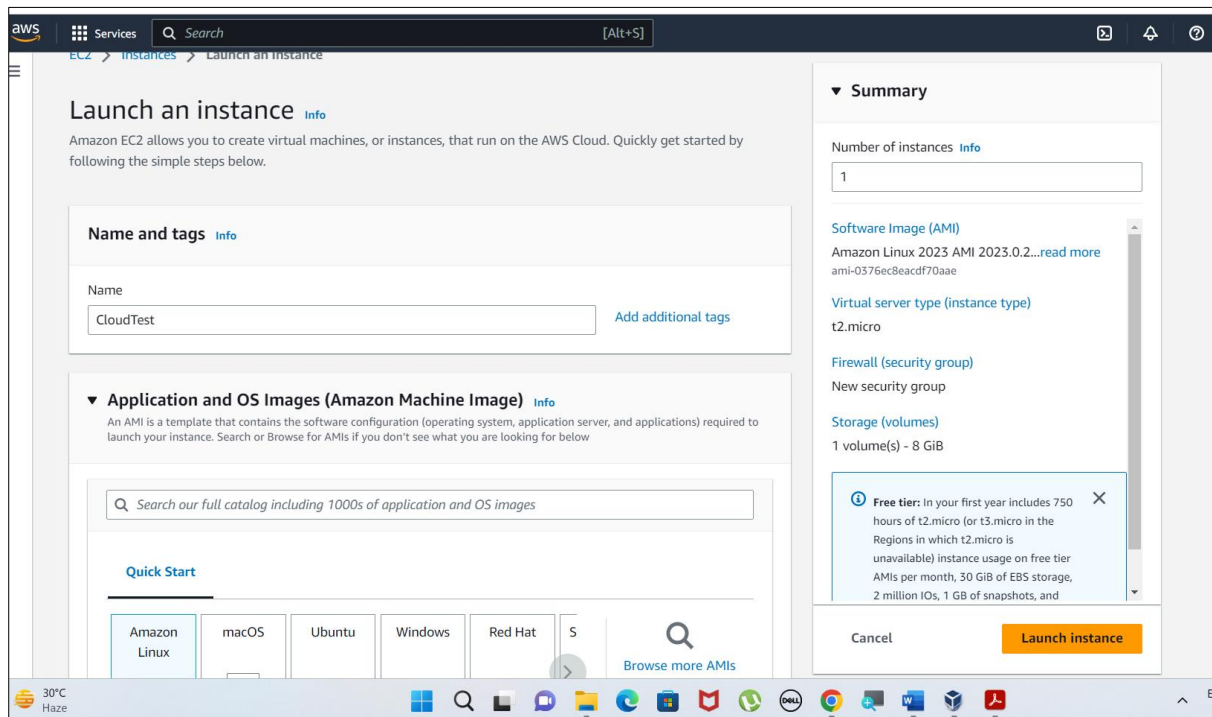
```
paresh@paresh-VirtualBox:~/dbda$ ls
dbda_B2
paresh@paresh-VirtualBox:~/dbda$ cd dbda_B2
paresh@paresh-VirtualBox:~/dbda/dbda_B2$ ls
Batch2.txt
paresh@paresh-VirtualBox:~/dbda/dbda_B2$
```

```
paresh@paresh-VirtualBox:~$ cp -r dbda dbdabu
paresh@paresh-VirtualBox:~$ ls
1.      1.txt      dbda      dbdabu    Documents  dpkg.txt  leapyr.sh  Music    pattern
1.sh    classwork  dbda_B2  Desktop  Downloads  h2.txt    ls.sh      paresh   patter..
```

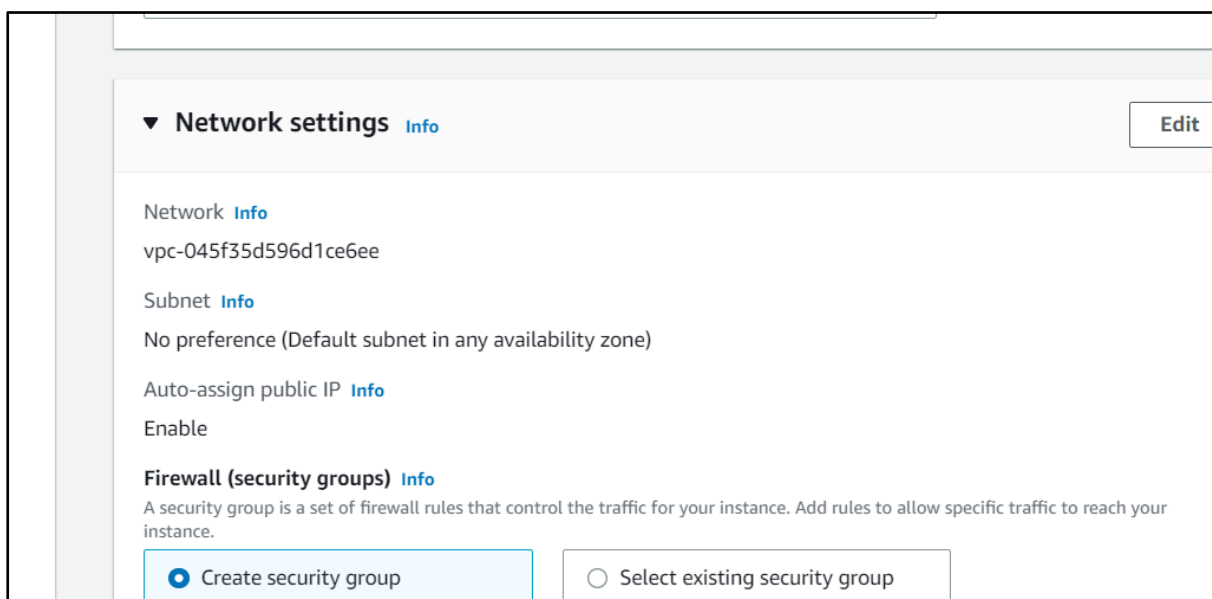
Cloud Questions on next page:

Q1.

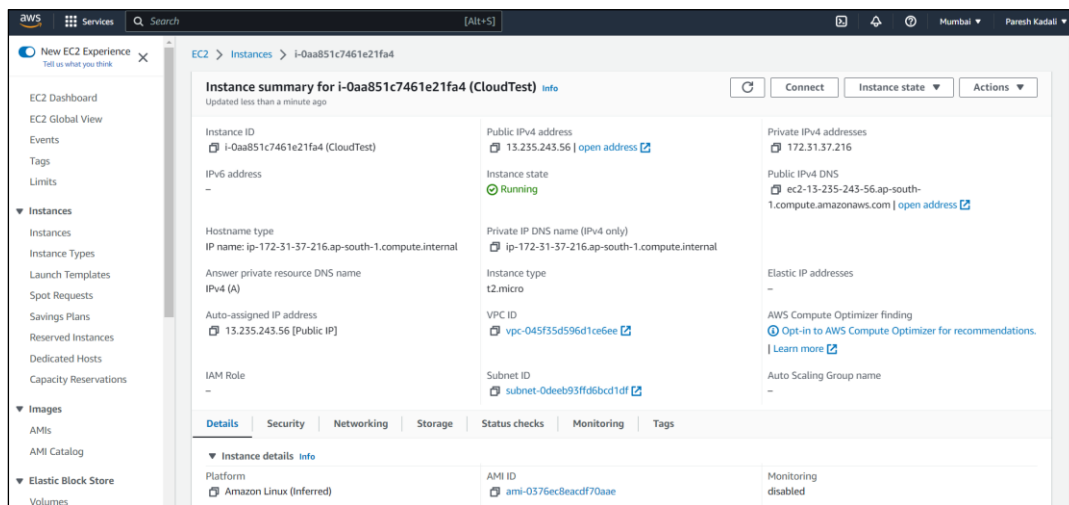
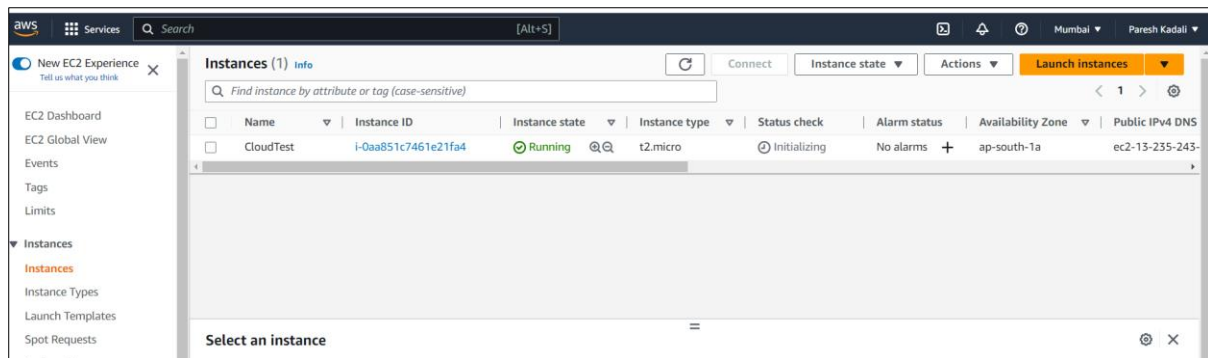
1. From AWS management console. after logging in to my account, one instance is created on EC2 platform as below screenshots.



2. Key pair is required to launch/create an instance. As the key was created is earlier, we can use the available one without creating a new one.

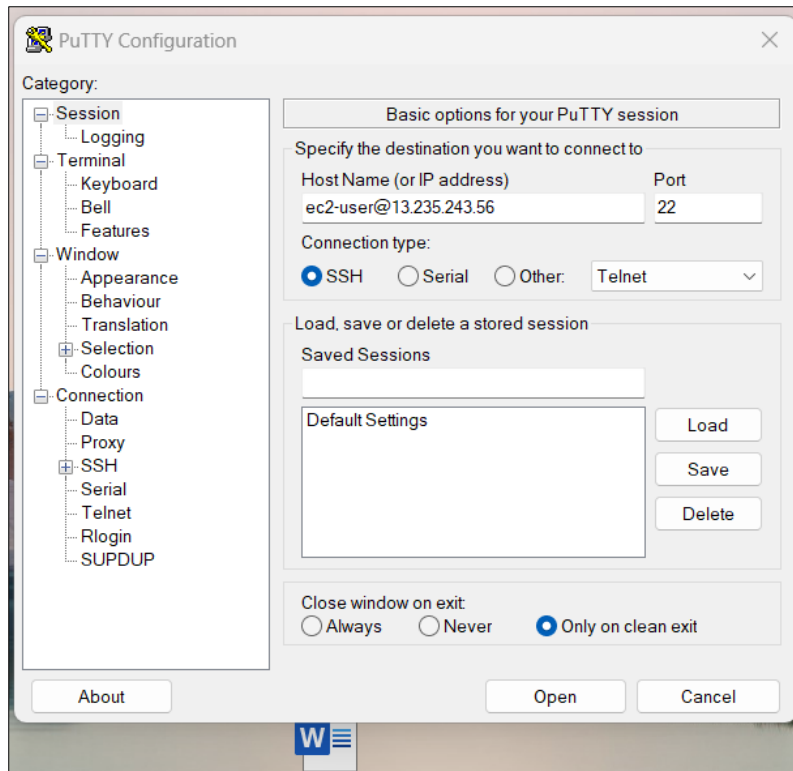


3. As it is seen here, instance has been created successfully and it is running.

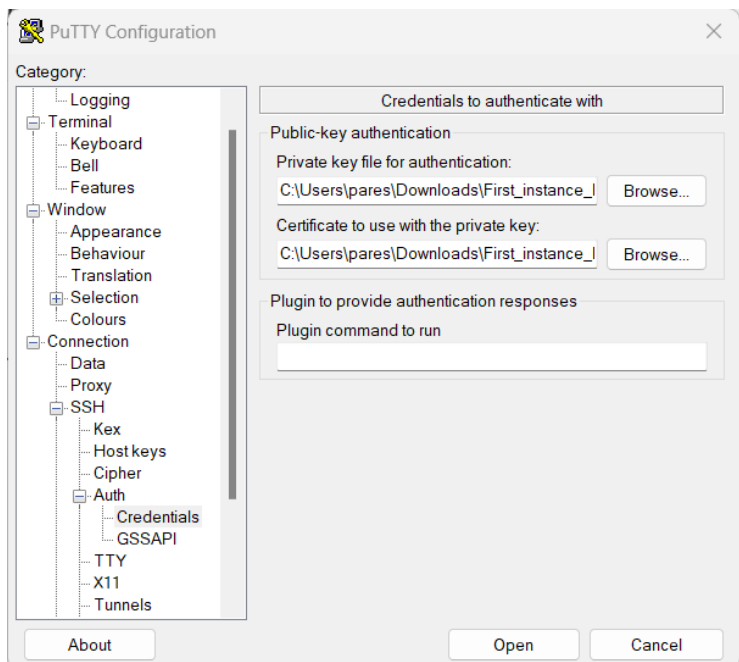


Continued in next page:

4. Putty application is started and the public IP address has been but as below:



5. .ppk file is added in Auth section of Putty.



6. Terminal is working successfully for the instance.

```
ec2-user@ip-172-31-37-216:~  
Unable to use certificate file "C:\Users\pires\Downloads\First_instance_key.p  
pk" (PuTTY SSH-2 private key)  
Using username "ec2-user".  
Authenticating with public key "First_instance_key"  
  
#_#####  
~\_#####\_ Amazon Linux 2023  
~~\_#####\  
~~\_###|  
~~\_#/ https://aws.amazon.com/linux/amazon-linux-2023  
~~V~'-'>  
~~~~  
~-.-.  
~/m/'-_-/_
```

[ec2-user@ip-172-31-37-216 ~]\$

7. Necessary files are installed as below:
Commands used: `yum update httpd -y`
`yum install httpd`

```

root@wp-1f4c-s1-sf-x86-~
(3/12): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm
(4/12): httpd-core-2.4.56-1.amzn2023.x86_64.rpm
(5/12): mod_http2-2.0.11-2.amzn2023.x86_64.rpm
(6/12): libbrotli-1.0.9-4.amzn2023.0.2.x86_64.rpm
(7/12): httpd-2.4.56-1.amzn2023.x86_64.rpm
(8/12): apr-1.7.2-2.amzn2023.0.2.x86_64.rpm
(9/12): mailcap-2.1.49-3.amzn2023.0.3.noarch.rpm
(10/12): apr-util-1.6.3-1.amzn2023.0.1.x86_64.rpm
(11/12): generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch.rpm
(12/12): httpd-filesystem-2.4.56-1.amzn2023.noarch.rpm
-----
Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing      :
Installing     : apr-1.7.2-2.amzn2023.0.2.x86_64
Installing     : apr-util-1.6.3-1.amzn2023.0.1.x86_64
Installing     : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
Installing     : mailcap-2.1.49-3.amzn2023.0.3.noarch
Installing     : httpd-tools-2.4.56-1.amzn2023.x86_64
Running scriptlet: httpd-filesystem-2.4.56-1.amzn2023.noarch
Installing     : httpd-filesystem-2.4.56-1.amzn2023.noarch
Installing     : httpd-core-2.4.56-1.amzn2023.x86_64
Installing     : mod_lua-2.4.56-1.amzn2023.x86_64
Installing     : mod_http2-2.0.11-2.amzn2023.x86_64
Installing     : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
Installing     : libbrotli-1.0.9-4.amzn2023.0.2.x86_64
Installing     : httpd-2.4.56-1.amzn2023.x86_64
Running scriptlet: httpd-2.4.56-1.amzn2023.x86_64
Verifying      : httpd-core-2.4.56-1.amzn2023.x86_64
Verifying      : httpd-tools-2.4.56-1.amzn2023.x86_64
Verifying      : mod_lua-2.4.56-1.amzn2023.x86_64
Verifying      : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
Verifying      : mod_http2-2.0.11-2.amzn2023.x86_64
Verifying      : libbrotli-1.0.9-4.amzn2023.0.2.x86_64
Verifying      : apr-1.7.2-2.amzn2023.0.2.x86_64
Verifying      : httpd-2.4.56-1.amzn2023.x86_64
Verifying      : apr-util-1.6.3-1.amzn2023.0.1.x86_64
Verifying      : mailcap-2.1.49-3.amzn2023.0.3.noarch
Verifying      : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
Verifying      : httpd-filesystem-2.4.56-1.amzn2023.noarch

Installed:
apr-1.7.2-2.amzn2023.0.2.x86_64      apr-util-1.6.3-1.amzn2023.0.1.x86_64      apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64      generic-logos
httpd-2.4.56-1.amzn2023.x86_64      httpd-core-2.4.56-1.amzn2023.x86_64      httpd-filesystem-2.4.56-1.amzn2023.noarch      httpd-tools-2
libbrotli-1.0.9-4.amzn2023.0.2.x86_64      mailcap-2.1.49-3.amzn2023.0.3.noarch      mod_http2-2.0.11-2.amzn2023.x86_64      mod_lua-2.4.5

```

8. In root documentary, necessary files are loaded and started using following commands:
- systemctl start httpd
 - systemctl enable httpd
- command systemctl status httpd is used to check if the system is active or not.

```
root@ip-172-31-37-216:~
[root@ip-172-31-37-216 ~]# systemctl start httpd
[root@ip-172-31-37-216 ~]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-37-216 ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Wed 2023-03-29 10:34:01 UTC; 57s ago
     Docs: man:httpd.service(8)
  Main PID: 25313 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 177 (limit: 1112)
   Memory: 12.8M
      CPU: 103ms
   CGroup: /system.slice/httpd.service
           └─25313 /usr/sbin/httpd -DFOREGROUND
             └─25325 /usr/sbin/httpd -DFOREGROUND
               └─25326 /usr/sbin/httpd -DFOREGROUND
                 └─25327 /usr/sbin/httpd -DFOREGROUND
                   └─25328 /usr/sbin/httpd -DFOREGROUND

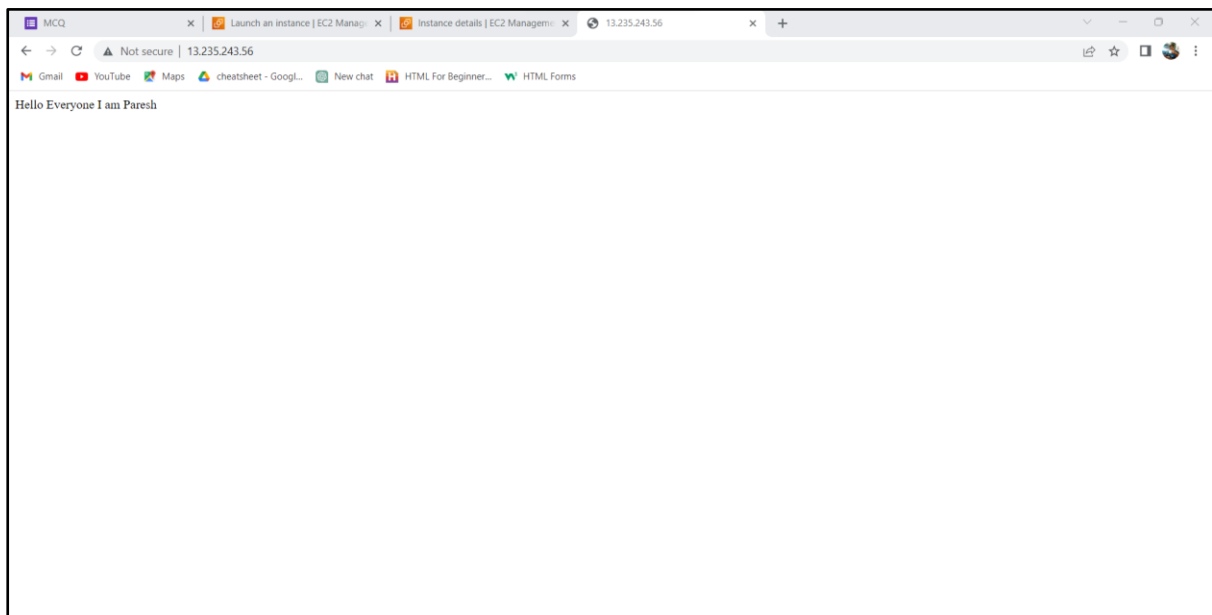
Mar 29 10:34:01 ip-172-31-37-216.ap-south-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Mar 29 10:34:01 ip-172-31-37-216.ap-south-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Mar 29 10:34:01 ip-172-31-37-216.ap-south-1.compute.internal httpd[25313]: Server configured, listening on: port 80
[root@ip-172-31-37-216 ~]#
```

9. Entering in root/var/www/html path, one html file name index.html is created using vi editor as below:

```
root@ip-172-31-37-216:/var/www/html
[
<html>
    <head>
        Hello Everyone
    </head>
    <body>
        I am Paresh
    </body>
</html>
~
~
~
~
~
~
```

continued on next page

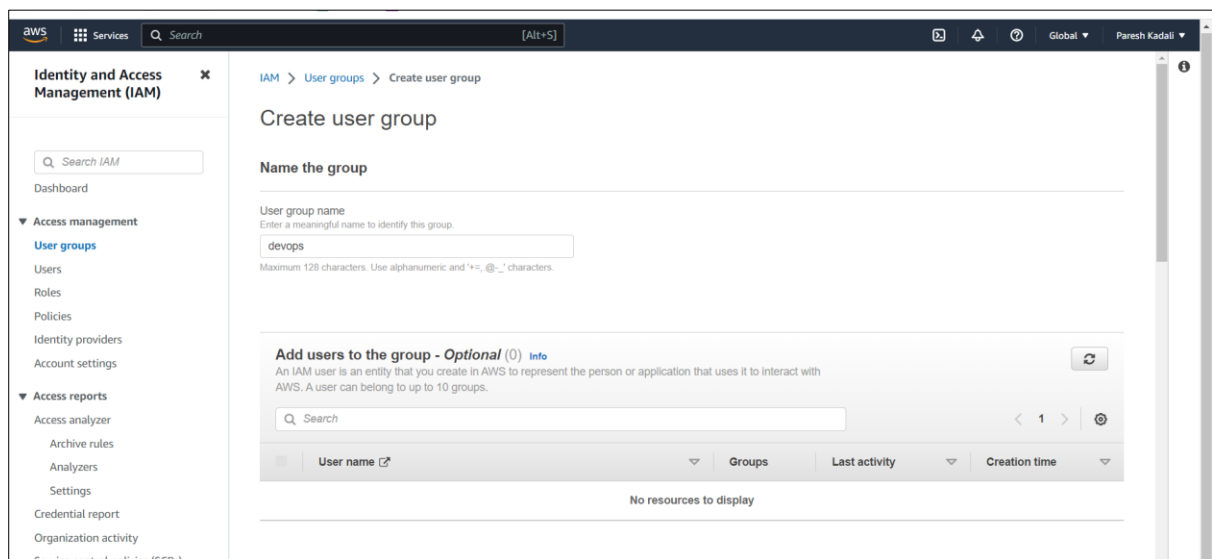
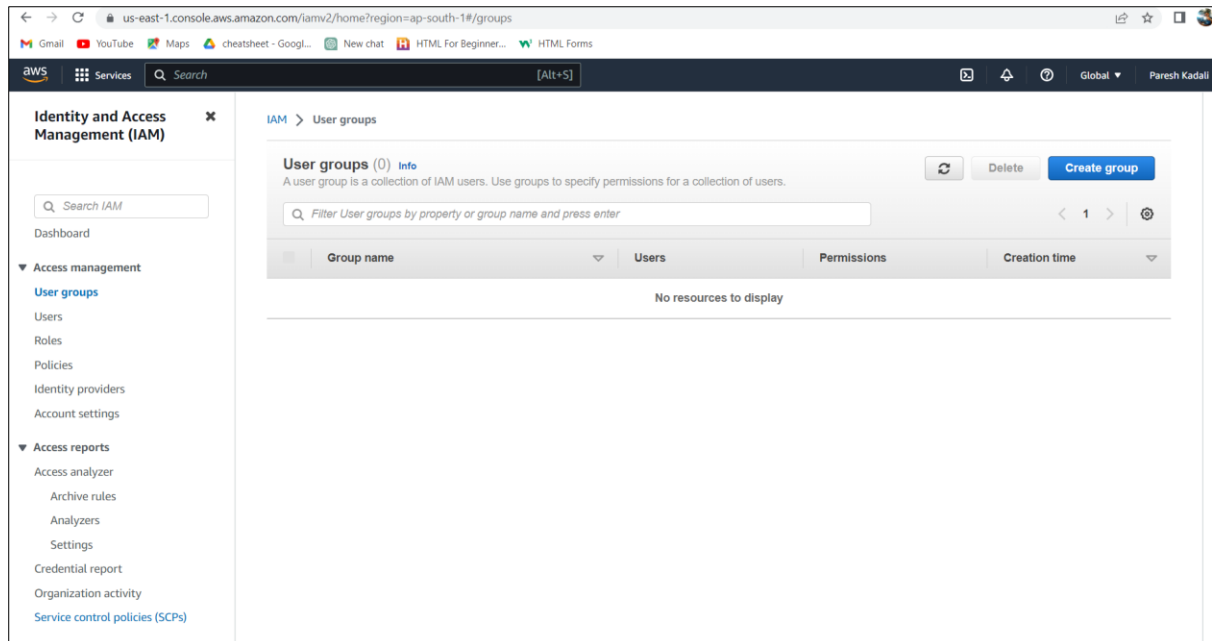
10. After successfully creating one html file we have been able to host/run the same file on our instance public IP as below:



Q2 is on next page

Q2.

1. After entering into IAM, user group named devops has been created as below:



continued in next page:

2. Group devops has been given the full access to use EC2 as below from system policies.

Ily

The screenshot shows the AWS IAM console interface. On the left is a navigation sidebar with 'Identity and Access Management (IAM)' selected. The main content area is titled 'Attach permissions policies - Optional (Selected 1/824)'. Below this, there's a search bar with the filter 'EC2' applied, showing 28 matches. A table lists various AWS managed policies. The 'AmazonEC2FullAccess' policy is selected with a checkbox.

	Policy name	Type	Description
<input checked="" type="checkbox"/>	AmazonEC2FullAccess	AWS managed	Provides full access to Ar
<input type="checkbox"/>	AmazonEC2RoleforSSM	AWS managed	This policy will soon be d
<input type="checkbox"/>	AmazonEC2RoleforAWSCodeDeploy	AWS managed	Provides EC2 access to :
<input type="checkbox"/>	AmazonEC2ContainerRegistryFullAccess	AWS managed	Provides administrative a
<input type="checkbox"/>	AmazonEC2ContainerRegistryReadOnly	AWS managed	Provides read-only acces
<input type="checkbox"/>	AmazonElasticMapReduceforEC2Role	AWS managed	Default policy for the Am
<input type="checkbox"/>	AmazonEC2ReadOnlyAccess	AWS managed	Provides read only acces
<input type="checkbox"/>	AmazonEC2SpotFleetAutoscaleRole	AWS managed	Policy to enable Autoscal
<input type="checkbox"/>	CloudWatchActionsEC2Access	AWS managed	Provides read-only acces
<input type="checkbox"/>	AmazonEC2ContainerServiceEventsRole	AWS managed	Policy to enable CloudW

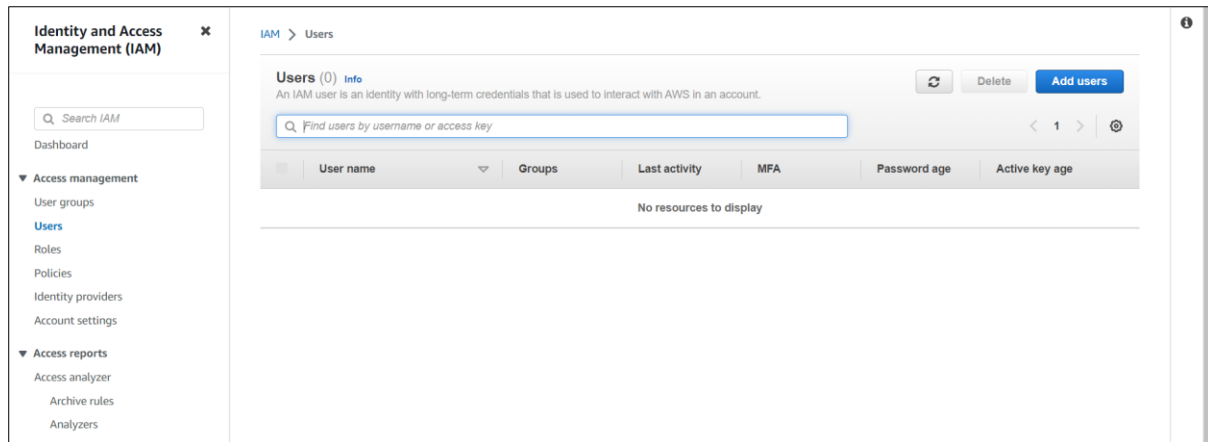
3. devops user group is created successfully.

The screenshot shows the AWS IAM console with a green notification banner at the top stating 'devops user group created.' with a 'View group' button. Below the notification, the 'User groups' page is displayed. It shows a table with one user group named 'devops'.

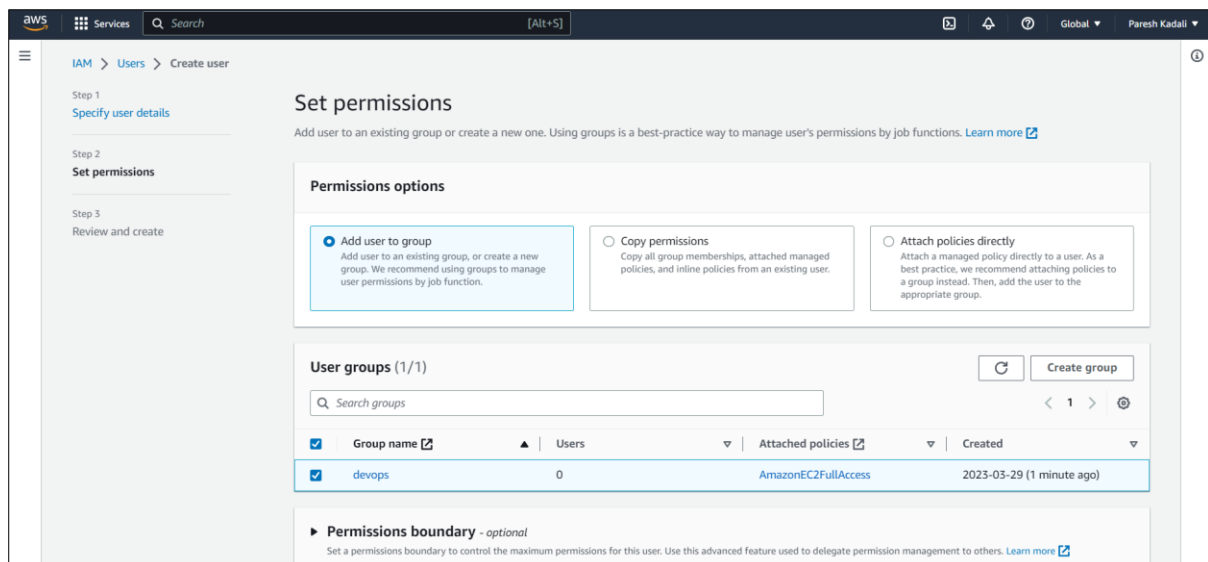
Group name	Users	Permissions	Creation time
devops	Loading	Defined	Now

continued on next page:

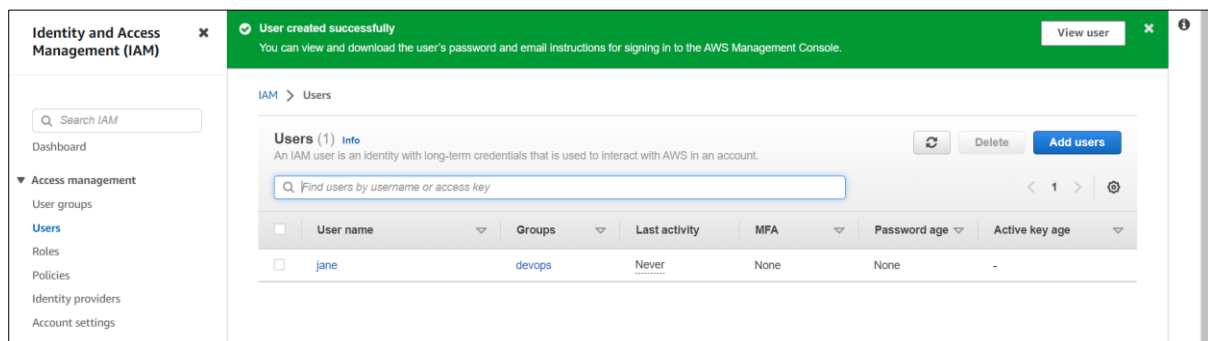
4. Also one user of name jane is created as below:



5. While creating only, user jane is added in group devops with the same permissions as the group.



6. User jane has been created.



continued on next page..

7. We can verify the user's permission in IAM policy simulator provides by AWS as below:
8. Here we have given the user full access to use EC2 machine, it should show the same related permissions in the simulator.

As seen below as we have not given any access for the API or backend linking to the user, it is showing denied. For EC2, it is showing every permission is allowed as expected.

The screenshot displays the AWS IAM Policy Simulator interface. On the left, the 'Policies' pane shows the selected policy 'AmazonEC2FullAccess' and its JSON statement. The main 'Policy Simulator' pane shows the simulation results for the 'Amazon EC2' policy. The 'Action Settings and Results' table indicates that 4 actions were selected, 0 were not simulated, 3 were allowed, and 1 was denied.

Service	Action	Resource Type	Simulation Resource	Permission
Amazon API Gateway	InvalidateCache	execute-api-general	*	denied
Amazon EC2	AcceptAddressTransfer	elastic-ip	*	allowed
Amazon EC2	AcceptTransitGatewayVpcAtta...	transit-gateway-attach...	*	allowed
Amazon EC2	AcceptReservedInstancesExch...	not required	*	allowed