

Assignment 1.

Configure EC2 linux machine and install apache configuration

The screenshot shows the AWS CloudWatch Metrics interface. A single metric named 'CPUUtilization' is displayed over a time range from 1 hour ago to now. The metric value is shown as a blue line graph with a red shaded confidence interval. The average CPU utilization is approximately 10%.

The screenshot shows a web browser window displaying the Apache default page. The title bar reads "Not secure 15.206.94.230". The main content of the page is the text "It works!".

Assignment 2.

Creating a Custom Amazon Machine Image (AMI)

- Launch a New EC2 Instance

- Install http on the new instance, enable the http service to start at boot.

- Create a New AMI from customised instance and name the AMI

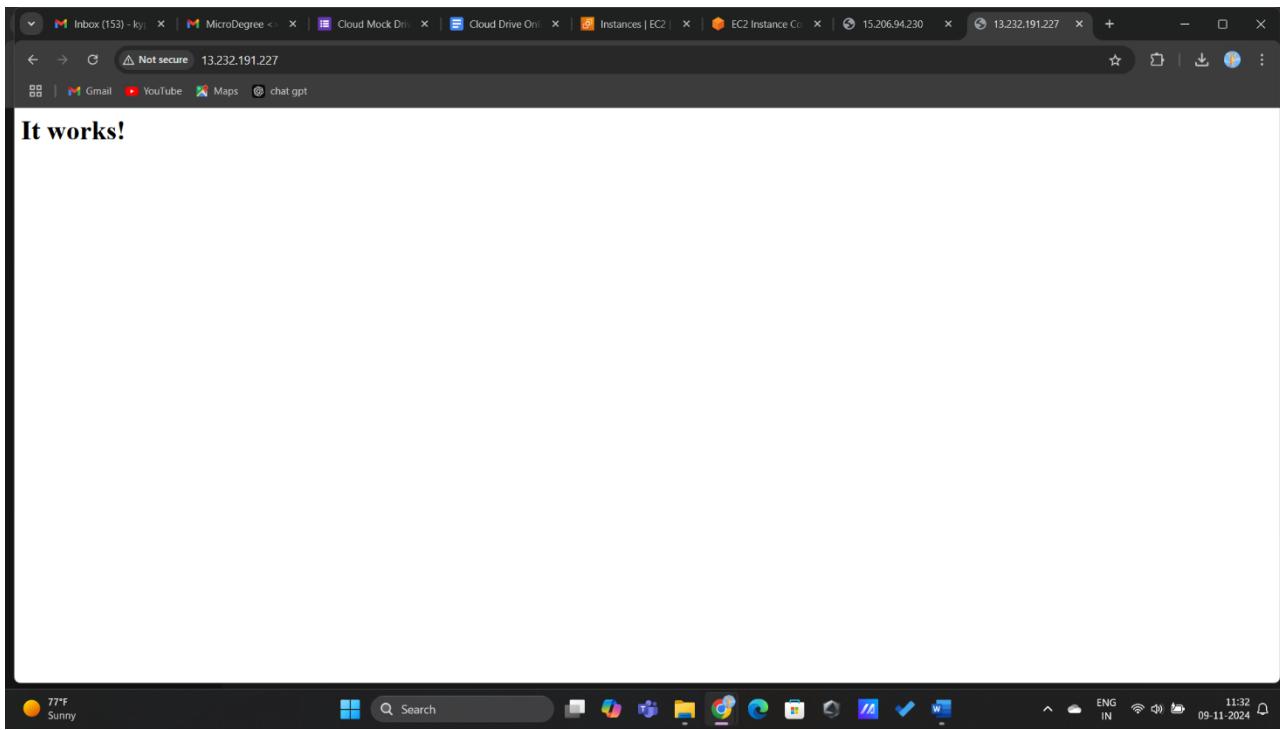
MicroDegreeWeb

- Launch a New Instance Using the Custom AM

- Verify that http is running

The screenshot shows the AWS Management Console with the EC2 Instances page open. The left sidebar shows navigation options like Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations (New), Images (selected), AMIs, AMI Catalog, and Elastic Block Store. The main content area displays a table of instances with columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4. Two instances are selected: 'Mock Test' (i-0bde07e877d5bbc69) and 'Mock Test2' (i-01668d110b99c9723), both in the 'Running' state. A modal window titled '2 instances selected' is open, showing monitoring metrics for CPU utilization, Network in (bytes), Network out (bytes), and Network packets in (cou...).

The screenshot shows the AWS Management Console with the Images | EC2 page open. The left sidebar shows the same navigation options as the previous screenshot. The main content area displays a table of AMIs with columns for Name, AMI name, AMI ID, Source, Owner, and Visibility. One AMI is selected: 'MicroDegreeW...' (ami-0a79a8d2468e86e5e). Below the table, a detailed view for the AMI 'ami-0a79a8d2468e86e5e (MicroDegreeWeb)' is shown with tabs for Details, Permissions, Storage, and Tags. The Details tab lists various configuration parameters such as AMI ID, Image type, Platform details, Root device type, AMI name, Owner account ID, Architecture, Usage operation, Root device name, Status, Source, Virtualization type, and Root mode.

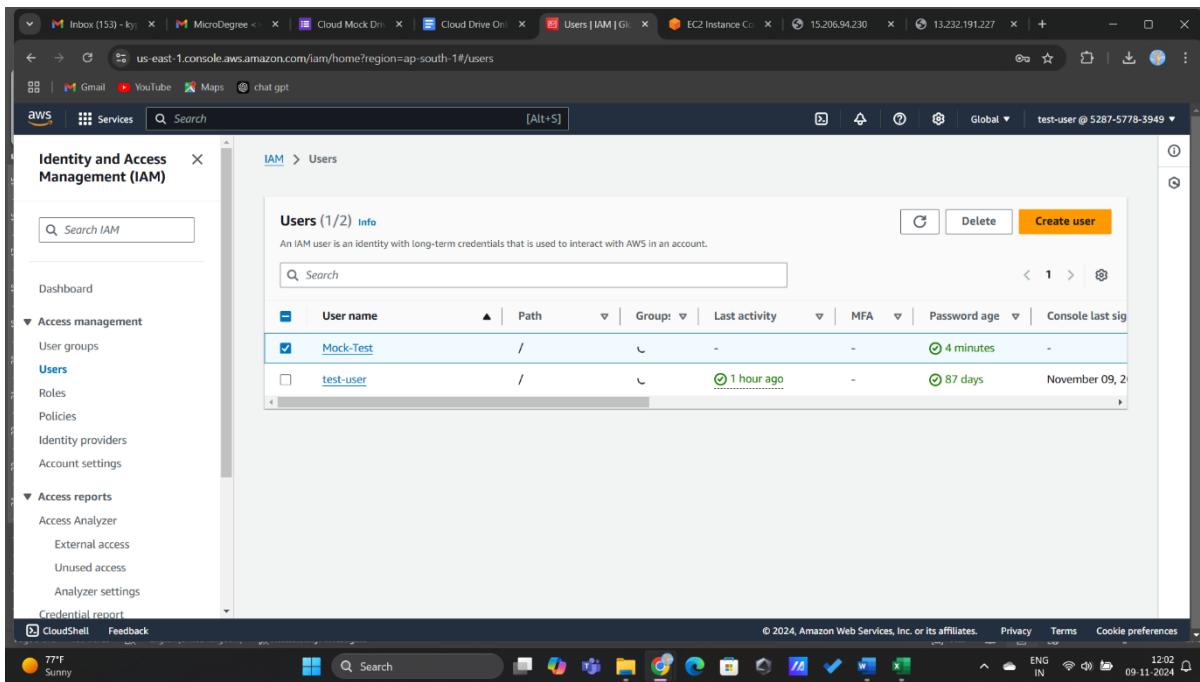


Assignment 4.

Creating an IAM User and Assigning Permissions:

Click on "Users" in the left-hand menu and choose "Add user."
Provide a username and select the access type (programmatic access, AWS Management Console access, or both).
Set permissions by adding the user to one or more IAM groups with predefined policies or by attaching custom policies directly.

Upload the final output Screenshot



Inbox (153) - ky | MicroDegree < | Cloud Mock Driv | Cloud Drive Onl | Create user | EC2 Instance Co | 15.206.94.230 | 13.232.191.227 | +

us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/users/create

Gmail YouTube Maps chat gpt

aws Services Search [Alt+S]

User created successfully

You can view and download the user's password and email instructions for signing in to the AWS Management Console.

IAM > Users > Create user

Step 1 Specify user details

Step 2 Set permissions

Step 3 Review and create

Step 4 Retrieve password

Retrieve password

You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

Console sign-in details

Email sign-in instructions

Console sign-in URL https://528757783949.signin.aws.amazon.com/console

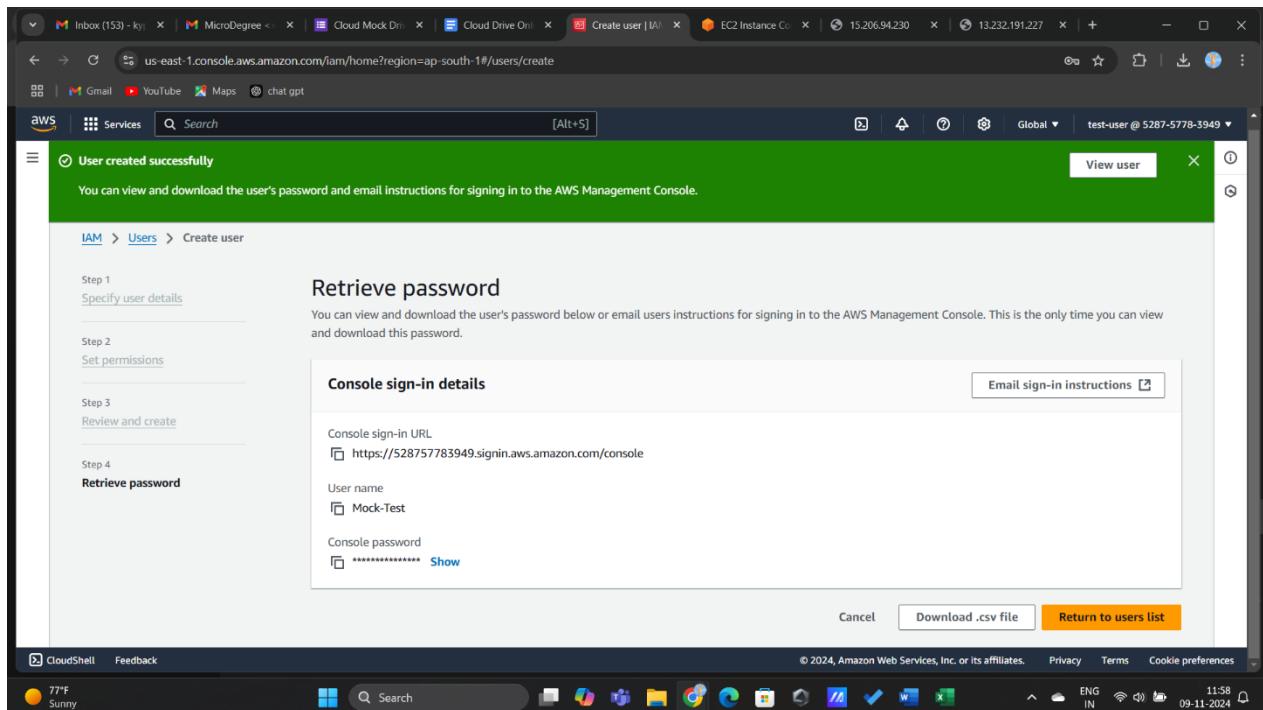
User name Mock-Test

Console password ***** Show

Cancel Download .csv file Return to users list

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

77°F Sunny ENG IN 11:58 09-11-2024



Inbox (153) - ky | MicroDegree < | Cloud Mock Driv | Cloud Drive Onl | Mock-Test | IAM | EC2 Instance Co | 15.206.94.230 | 13.232.191.227 | +

us-east-1.console.aws.amazon.com/iam/home?region=ap-south-1#/users/details/Mock-Test?section=permissions

Gmail YouTube Maps chat gpt

aws Services Search [Alt+S]

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

Access reports

Access Analyzer

External access

Unused access

Analyzer settings

Credential report

Mock-Test info

Delete

Summary

ARN arn:aws:iam::528757783949:user/Mock-Test

Created November 09, 2024, 11:57 (UTC+05:30)

Console access Enabled without MFA

Last console sign-in Never

Access key 1 Create access key

Permissions Groups Tags Security credentials Last Accessed

Permissions policies (3)

Permissions are defined by policies attached to the user directly or through groups.

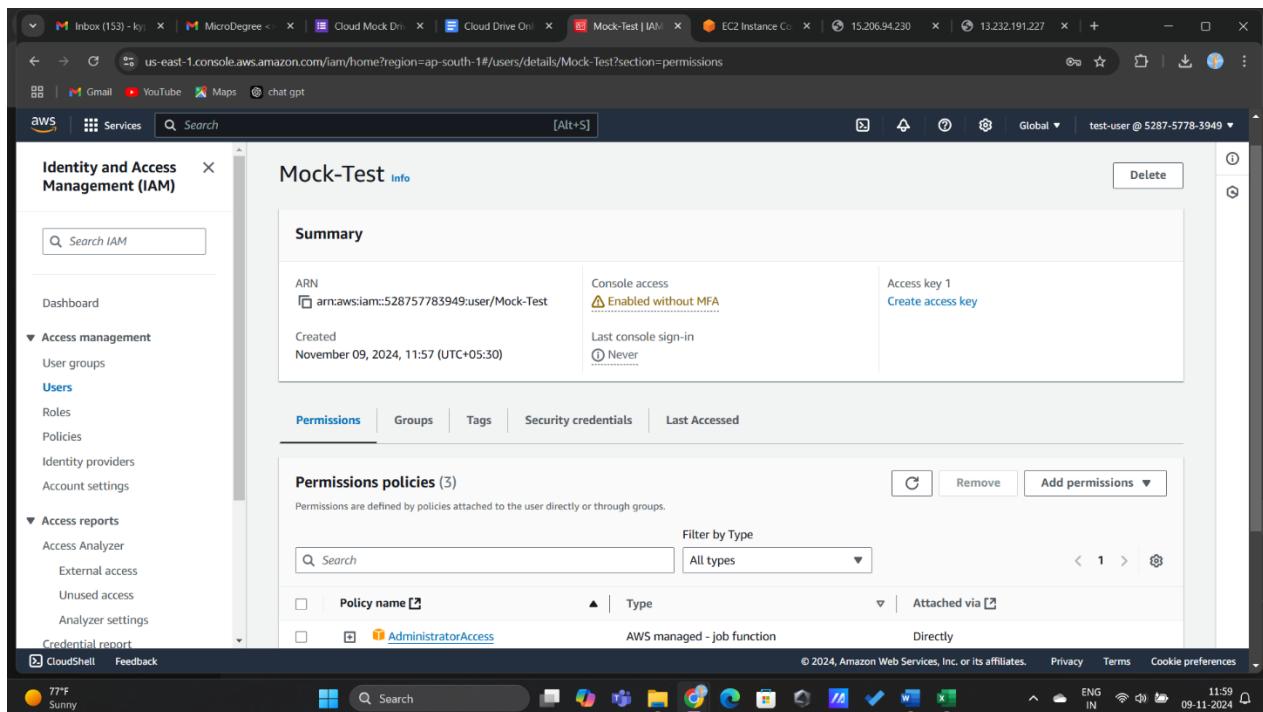
Filter by Type

Search All types

Policy name AdministratorAccess Type AWS managed - job function Attached via Directly

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

77°F Sunny ENG IN 11:58 09-11-2024



Identity and Access Management (IAM)

ARN: arn:aws:iam::528757783949:user/Mock-Test

Created: November 09, 2024, 11:57 (UTC+05:30)

Console access: Enabled without MFA

Last console sign-in: Never

Access key 1: Create access key

Permissions | Groups | Tags | Security credentials | Last Accessed

Permissions policies (3)

AdministratorAccess, AmazonEC2FullAccess, AmazonS3FullAccess

Assignment 5

Create 2 VPC's Named "Webapp-VPC" & "Db-VPC"
It should have 2 Subnets each, one with Class A IPv4 CIDR and Class B IPv4 CIDR, and 255 ports in each subnet.

VPC dashboard

Your VPCs (2/3)

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHC
-	vpc-005b7f0fa0f44ef74	Available	172.31.0.0/16	-	dopt
Webapp-VPC-vpc	vpc-01d81efe0e1b535ee	Available	10.0.0.0/16	-	dopt
Db-VPC-vpc	vpc-03da0f88187f401d9	Available	10.0.0.0/16	-	dopt

Screenshot of the AWS VPC console showing the VPC dashboard. The main pane displays three VPCs: 'vpc-005b7f0fa0f44ef74' (Available, 172.31.0.0/16), 'Webapp-VPC.-vpc' (Available, 10.0.0.0/16), and 'Db-VPC.-vpc' (Available, 10.0.0.0/16). The 'Db-VPC.-vpc' row is selected. Below the table, there are three cards: 'VPC Show details' (Db-VPC.-vpc), 'Subnets (4)' (ap-south-1a and ap-south-1b), and 'Route tables (4)' (rtb-0331d922923864878, Db-VPC-rtb-private2-ap-south-1b, Db-VPC-rtb-public, Db-VPC-rtb-private1-ap-south-1a). A network connection diagram shows the relationships between these resources.

Screenshot of the AWS VPC console showing the details for the selected VPC ('Db-VPC.-vpc'). The 'Details' tab is active, displaying the following configuration:

VPC ID	State	DNS hostnames	DNS resolution
vpc-03da0f88187f401d9	Available	Enabled	Enabled
Tenancy	DHCP option set	Main route table	Main network ACL
Default	dopt-03874d5eb2d7b5d67	rtb-0331d922923864878	acl-0abcf2a2d03770bf
Default VPC	IPv4 CIDR	IPv6 pool	IPv6 CDR (Network border group)
No	10.0.0.0/16	-	-
Network Address Usage metrics	Route 53 Resolver DNS Firewall rule groups	Owner ID	
Disabled	-	528757783949	

The 'Resource map' tab is also visible, showing a visual representation of the VPC structure.

Screenshot of the AWS VPC console showing the Resource map for the Webapp-VPC. The interface displays four subnets under ap-south-1a and ap-south-1b, each associated with a route table (rtb-0c2cd7710b7a19ed3, rtb-0c2cd7710b7a19ed5) which connects to a central network connection point.

VPC dashboard

EC2 Global View Filter by VPC

Virtual private cloud Your VPCs Subnets Route tables Internet gateways Egress-only internet gateways DHCP option sets Elastic IPs Managed prefix lists Endpoints Endpoint services NAT gateways Peering connections Security CloudShell Feedback

Resource map Info

VPC Show details Your AWS virtual network Webapp-VPC-vpc

Subnets (4) Subnets within this VPC

ap-south-1a

- Webapp-VPC-subnet-public1-ap-south-1a
- Webapp-VPC-subnet-private1-ap-south-1a

ap-south-1b

- Webapp-VPC-subnet-public2-ap-south-1b
- Webapp-VPC-subnet-private2-ap-south-1b

Route tables (4) Route network traffic to resources

- Webapp-VPC-rtb-private1-ap-south-1a
- Webapp-VPC-rtb-public
- Webapp-VPC-rtb-private2-ap-south-1b
- rtb-0c2cd7710b7a19ed3

Network connections to

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS VPC console showing the list of VPCs. The Webapp-VPC-vpc is selected, revealing its detailed configuration and associated resources.

VPC dashboard

EC2 Global View Filter by VPC

Virtual private cloud Your VPCs Subnets Route tables Internet gateways Egress-only internet gateways DHCP option sets Elastic IPs Managed prefix lists Endpoints Endpoint services NAT gateways Peering connections Security CloudShell Feedback

Your VPCs (1/3) Info

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHC
-	vpc-005b7f0fa0f44ef74	Available	172.31.0.0/16	-	dopt
<input checked="" type="checkbox"/> Webapp-VPC-vpc	vpc-01d81efe0e1b535ee	Available	10.0.0.0/16	-	dopt
<input type="checkbox"/> Db-VPC-vpc	vpc-03da0f88187f401d9	Available	10.0.0.0/16	-	dopt

VPC Show details Your AWS virtual network Webapp-VPC-vpc

Subnets (4) Subnets within this VPC

ap-south-1a

- Webapp-VPC-subnet-public1-ap-south-1a
- Webapp-VPC-subnet-private1-ap-south-1a

ap-south-1b

- Webapp-VPC-subnet-public2-ap-south-1b
- Webapp-VPC-subnet-private2-ap-south-1b

Route tables (4) Route network traffic to resources

- Webapp-VPC-rtb-private1-ap-south-1a
- Webapp-VPC-rtb-public
- Webapp-VPC-rtb-private2-ap-south-1b
- rtb-0c2cd7710b7a19ed3

Network connections to

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Screenshot of the AWS VPC console showing the Resource map for the Db-VPC-vpc. The interface displays four subnets under ap-south-1a and ap-south-1b, each associated with a route table (rtb-0331d922923864878, rtb-03da0f88187f401d9) which connects to a central network connection point.

VPC dashboard

EC2 Global View Filter by VPC

Virtual private cloud Your VPCs Subnets Route tables Internet gateways Egress-only internet gateways DHCP option sets Elastic IPs Managed prefix lists Endpoints Endpoint services NAT gateways Peering connections Security CloudShell Feedback

Resource map Info

VPC Show details Your AWS virtual network Db-VPC-vpc

Subnets (4) Subnets within this VPC

ap-south-1a

- Db-VPC-subnet-public1-ap-south-1a
- Db-VPC-subnet-private1-ap-south-1a

ap-south-1b

- Db-VPC-subnet-public2-ap-south-1b
- Db-VPC-subnet-private2-ap-south-1b

Route tables (4) Route network traffic to resources

- rtb-0331d922923864878
- Db-VPC-rtb-private2-ap-south-1b
- Db-VPC-rtb-public
- Db-VPC-rtb-private1-ap-south-1a

Network connections to

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Assignment 9

Create a new GitHub repository, push sample files from local to remote, with a commit message of 'First commit'. Please note that the repository should be set to private.

```
Running scriptlet: docker-25.0.6-1.amzn2023.0.2.x86_64
Created symlink /etc/systemd/system/sockets.target.wants/docker.socket → /usr/lib/systemd/system/docker.socket.

Verifying : containerd-1.7.22-1.amzn2023.0.2.x86_64
Verifying : docker-25.0.6-1.amzn2023.0.2.x86_64
Verifying : iptables-libs-1.8.8-3.amzn2023.0.2.x86_64
Verifying : iptables-nft-1.8.8-3.amzn2023.0.2.x86_64
Verifying : libcgroup-3.0-1.amzn2023.0.1.x86_64
Verifying : libnetfilter_conntrack-1.0.8-2.amzn2023.0.2.x86_64
Verifying : libnftnl-1.2.2-2.amzn2023.0.2.x86_64
Verifying : pigz-2.5-1.amzn2023.0.3.x86_64
Verifying : runc-1.1.14-1.amzn2023.0.1.x86_64

Installed:
containerd-1.7.22-1.amzn2023.0.2.x86_64
iptables-nft-1.8.8-3.amzn2023.0.2.x86_64
libnftnl-1.2.2-2.amzn2023.0.2.x86_64
runc-1.1.14-1.amzn2023.0.1.x86_64

Complete!
[root@ip-172-31-47-231 ~]# which git
/usr/bin/git
[root@ip-172-31-47-231 ~]# which docker
/usr/bin/docker
[root@ip-172-31-47-231 ~]# [root@ip-172-31-47-231 ~]# i-0bde07e877d5bbc69 (Mock Test)
PublicIPs: 15.206.94.230 PrivateIPs: 172.31.47.231

CloudShell Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
CloudShell Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
CloudShell Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```

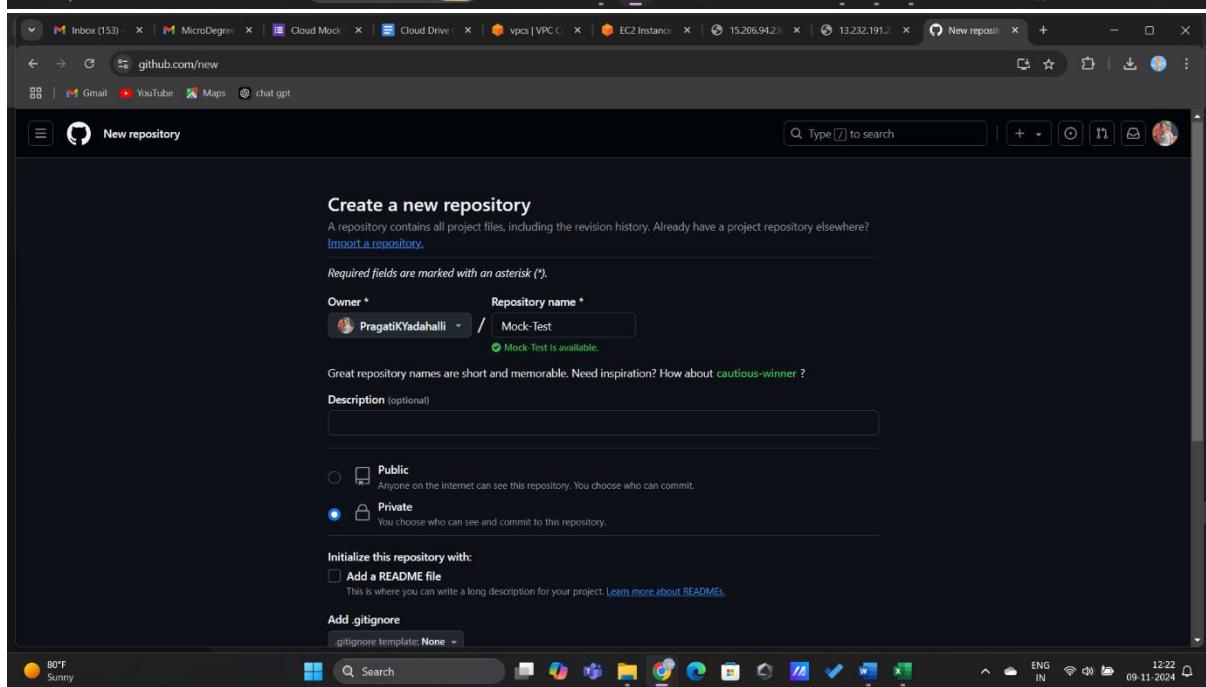
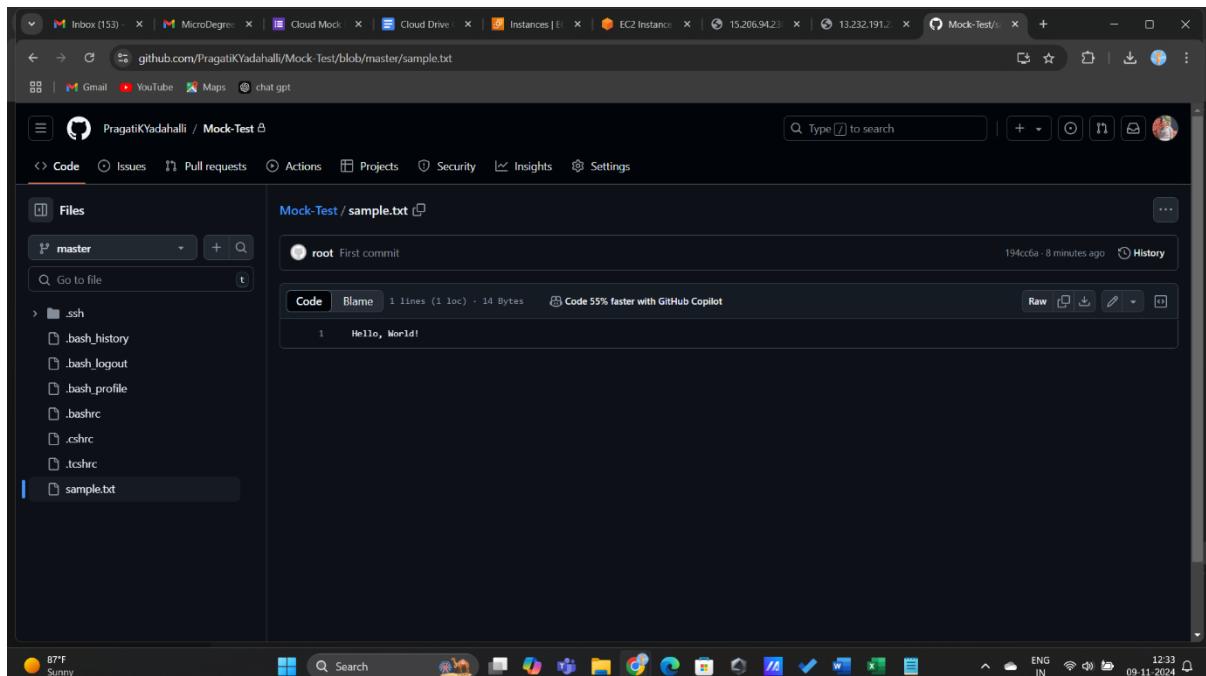
```
.lessht
nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-47-231 ~]# git commit -m "First commit"
On branch master
Untracked files:
 (use "git add <file>..." to include in what will be committed)
 .gitconfig
 .lessht

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-47-231 ~]# git commit -m "First commit"
On branch master
Untracked files:
 (use "git add <file>..." to include in what will be committed)
 .gitconfig
 .lessht

nothing added to commit but untracked files present (use "git add" to track)
[root@ip-172-31-47-231 ~]# git log
commit 194cc6a143c1400aa96af4d6bfff2c0cb2ff00 [HEAD -> master]
Author: root <root@ip-172-31-47-231.ap-south-1.compute.internal>
Date:   Sat Nov 9 06:55:18 2024 +0000

 First commit
[root@ip-172-31-47-231 ~]# [root@ip-172-31-47-231 ~]# i-0bde07e877d5bbc69 (Mock Test)
PublicIPs: 15.206.94.230 PrivateIPs: 172.31.47.231

CloudShell Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
CloudShell Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
CloudShell Feedback
© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences
```



```
commit 194cc6a143c1400aa96af4d6bfff2c0cb2ff00 (HEAD -> master)
Author: root <root@ip-172-31-47-231.ap-south-1.compute.internal>
Date:   Sat Nov 9 06:55:18 2024 +0000

First commit

diff --git a/.bash_history b/.bash_history
new file mode 100644
index 0000000..4ec969c
--- /dev/null
+++ b/.bash_history
@@ -0,0 +1,4 @@
+sudo yum update -y
+sudo yum install httpd -y
+sudo systemctl start httpd
+sudo systemctl enable httpd
diff --git a/.bash_logout b/.bash_logout
new file mode 100644
index 0000000..aff7c6fd
--- /dev/null
+++ b/.bash_logout
@@ -0,0 +1,2 @@
+#!/bin/bash
diff --git a/.bash_profile b/.bash_profile
:;

i-0bde07e877d5bbc69 (Mock Test)
PublicIPs: 15.206.94.230 PrivateIPs: 172.31.47.231
```

Assignment 10

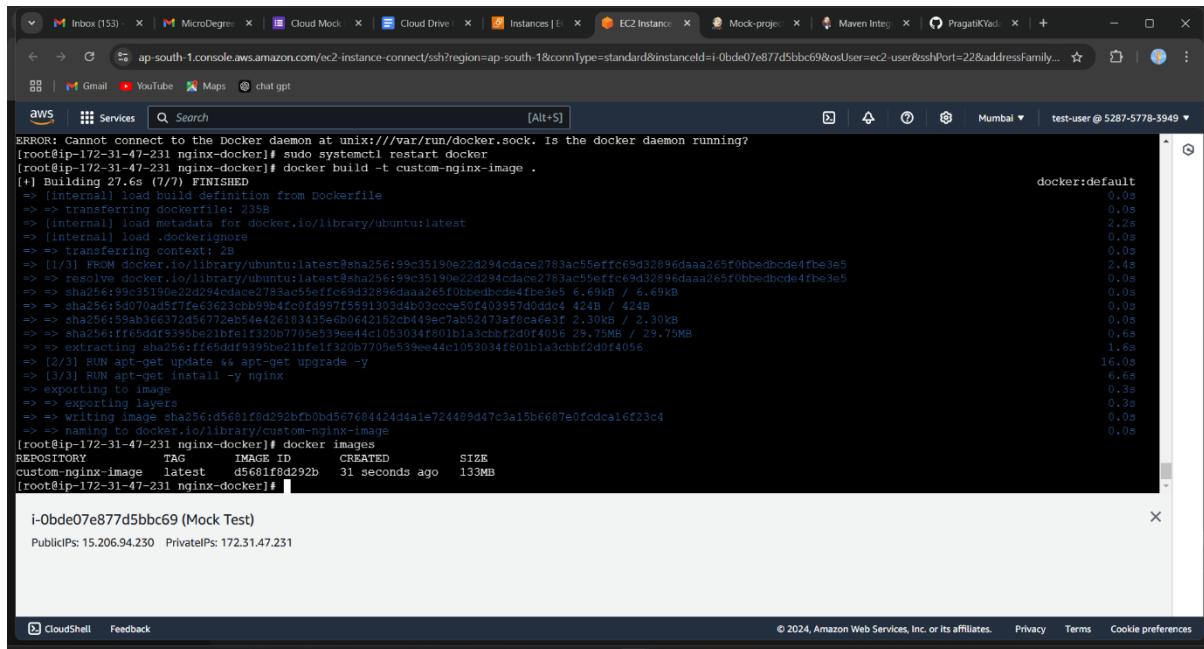
Create a Jenkins freestyle project, configure a sample build with the Maven plugin, and please share the output of the build.

The screenshot shows the Jenkins configuration interface for a 'Mock-project'. The 'Build Steps' section is expanded, showing a 'Invoke top-level Maven targets' step. Under 'Goals', the value 'clean install' is specified. Other options like 'Maven Version' (set to 'Maven') and 'Advanced' are also visible. Below the configuration area are 'Save' and 'Apply' buttons. The browser address bar shows the URL: 15.206.94.230:8080/job/Mock-project/configure.

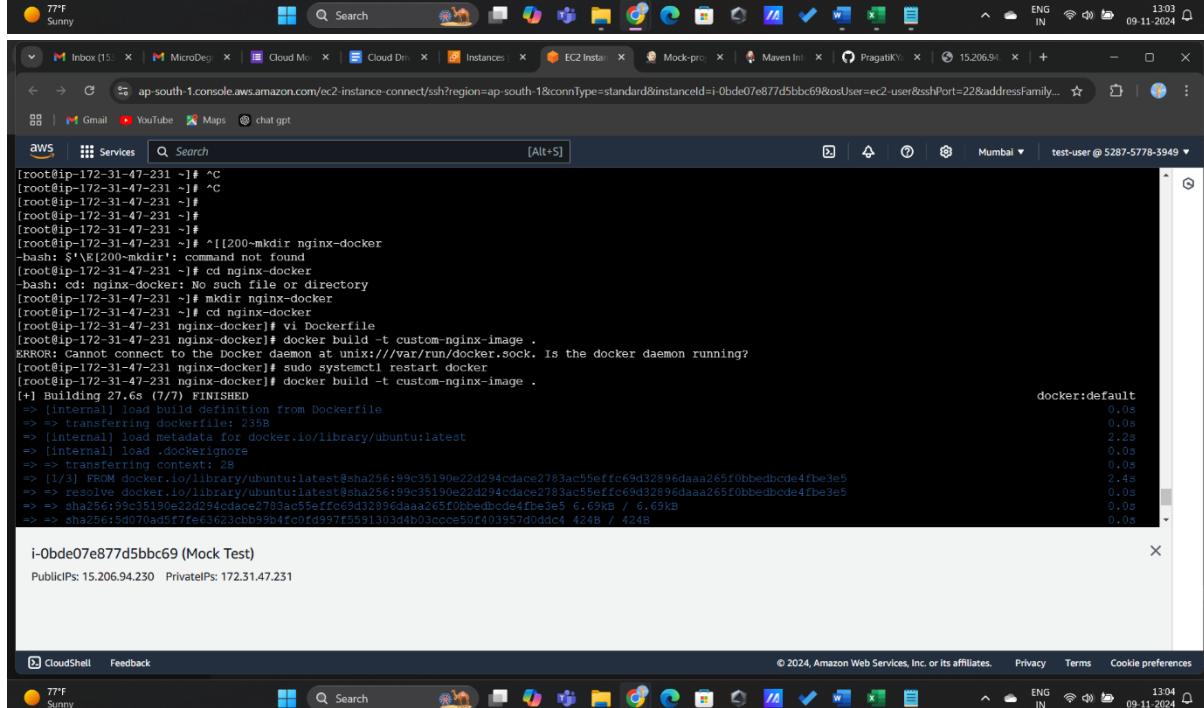
The screenshot shows the Jenkins dashboard for the 'Mock-project'. The left sidebar includes links for Status, Changes, Workspace, Build Now, Configure, Delete Project, and Rename. The main area displays the 'Mock-project' status as 'Up-to-date'. Below it, the 'Builds' section lists two builds: '#2 7:26 AM' and '#1 7:26 AM', both marked as successful (green). The bottom right corner shows the Jenkins version as 'Jenkins 2.479.1'. The browser address bar shows the URL: 15.206.94.230:8080/job/Mock-project/.

Assignment 11

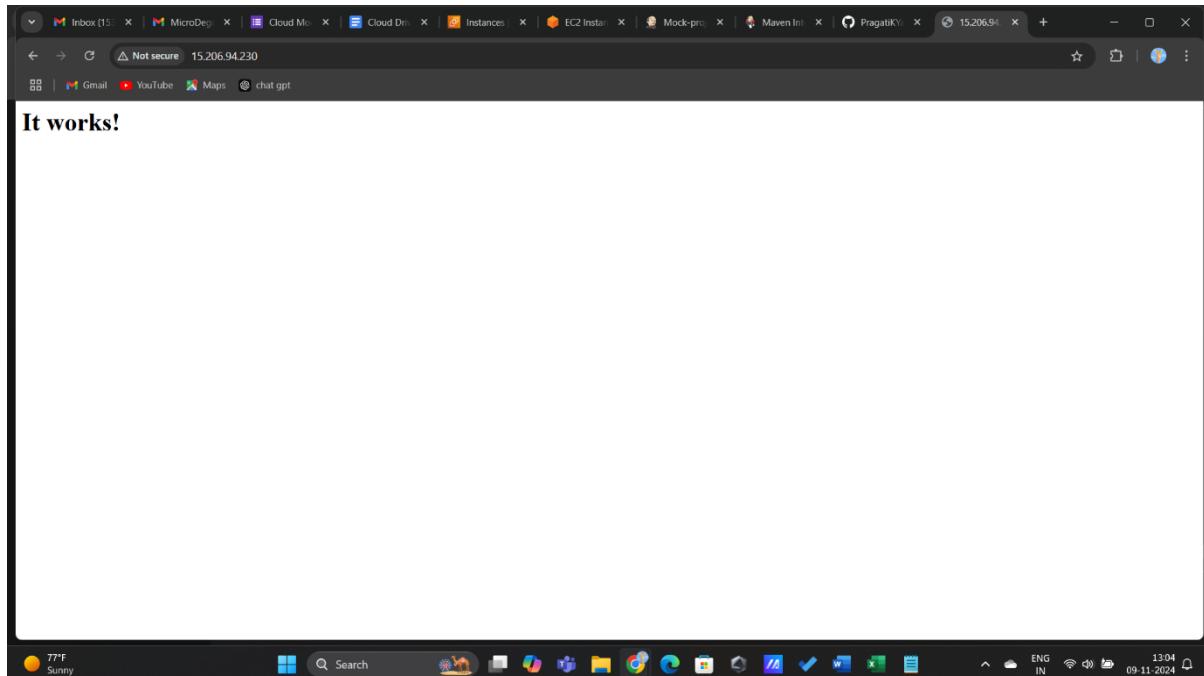
Build a custom docker image using Ubuntu as a base docker image and run the nginx application. - this docker image should be built using a Dockerfile. Once the docker image is build, start the docker image using the host network and make it accessible on Public IP



```
ERROR: Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?  
[root@ip-172-31-47-231 nginx-docker]# sudo systemctl restart docker  
[root@ip-172-31-47-231 nginx-docker]# docker build -t custom-nginx-image .  
[*] Building 27.6s (7/7) FINISHED  
=> [internal] load build definition from Dockerfile  
=> transferring dockerfile: 23B  
=> [internal] load metadata for docker.io/library/ubuntu:latest  
=> [internal] load .dockerignore  
=> transferring context: 2B  
=> [1/3] FROM docker.io/library/ubuntu:latest@sha256:99c35190e22d294cdace2783ac55efffc69d32896daaa265f0bbcdce4fbe3e5  
=> resolve docker.io/library/ubuntu:latest@sha256:99c35190e22d294cdace2783ac55efffc69d32896daaa265f0bbcdce4fbe3e5  
=> sha256:5d070ad577fe3623ccb984fc0fd997f55913034b03ccce50f403957d0ddc4 424B / 424B  
=> sha256:99c35190e22d294cdace2783ac55efffc69d32896daaa265f0bbcdce4fbe3e5  
=> sha256:f1f65ddf9395be21bfe1f320b705e539ee44c1053034f801b1a3cbff2df04056 29.75MB / 29.75MB  
=> extracting sha256:f1f65ddf9395be21bfe1f320b705e539ee44c1053034f801b1a3cbff2df04056  
=> [2/3] RUN apt-get update && apt-get upgrade -y  
=> [3/3] RUN apt-get install -y nginx  
=> exporting to image  
=> exporting layers  
=> writing image sha256:d5681f8d292fbf0bd56764424d4ale724409d47c3a15b6687e0fcddca16f23c4  
=> naming to docker.io/library/custom-nginx-image  
[root@ip-172-31-47-231 nginx-docker]# docker images  
REPOSITORY          TAG      IMAGE ID      CREATED             SIZE  
custom-nginx-image latest  d5681f8d292b  31 seconds ago   133MB  
[root@ip-172-31-47-231 nginx-docker]#  
  
i-0bde07e877d5bbc69 (Mock Test)  
PublicIPs: 15.206.94.230 PrivateIPs: 172.31.47.231
```



```
[root@ip-172-31-47-231 ~]# cd nginx-docker  
[root@ip-172-31-47-231 nginx-docker]# ls  
Dockerfile  index.html  nginx.conf  
[root@ip-172-31-47-231 nginx-docker]# docker build -t custom-nginx-image .  
[*] Building 27.6s (7/7) FINISHED  
=> [internal] load build definition from Dockerfile  
=> transferring dockerfile: 23B  
=> [internal] load metadata for docker.io/library/ubuntu:latest  
=> [internal] load .dockerignore  
=> transferring context: 2B  
=> [1/3] FROM docker.io/library/ubuntu:latest@sha256:99c35190e22d294cdace2783ac55efffc69d32896daaa265f0bbcdce4fbe3e5  
=> resolve docker.io/library/ubuntu:latest@sha256:99c35190e22d294cdace2783ac55efffc69d32896daaa265f0bbcdce4fbe3e5  
=> sha256:99c35190e22d294cdace2783ac55efffc69d32896daaa265f0bbcdce4fbe3e5  
=> sha256:5d070ad577fe3623ccb984fc0fd997f55913034b03ccce50f403957d0ddc4 424B / 424B  
=> sha256:99c35190e22d294cdace2783ac55efffc69d32896daaa265f0bbcdce4fbe3e5  
=> sha256:f1f65ddf9395be21bfe1f320b705e539ee44c1053034f801b1a3cbff2df04056 29.75MB / 29.75MB  
=> extracting sha256:f1f65ddf9395be21bfe1f320b705e539ee44c1053034f801b1a3cbff2df04056  
=> [2/3] RUN apt-get update && apt-get upgrade -y  
=> [3/3] RUN apt-get install -y nginx  
=> exporting to image  
=> writing image sha256:d5681f8d292fbf0bd56764424d4ale724409d47c3a15b6687e0fcddca16f23c4  
=> naming to docker.io/library/custom-nginx-image  
[root@ip-172-31-47-231 nginx-docker]# docker images  
REPOSITORY          TAG      IMAGE ID      CREATED             SIZE  
custom-nginx-image latest  d5681f8d292b  31 seconds ago   133MB  
[root@ip-172-31-47-231 nginx-docker]#  
  
i-0bde07e877d5bbc69 (Mock Test)  
PublicIPs: 15.206.94.230 PrivateIPs: 172.31.47.231
```



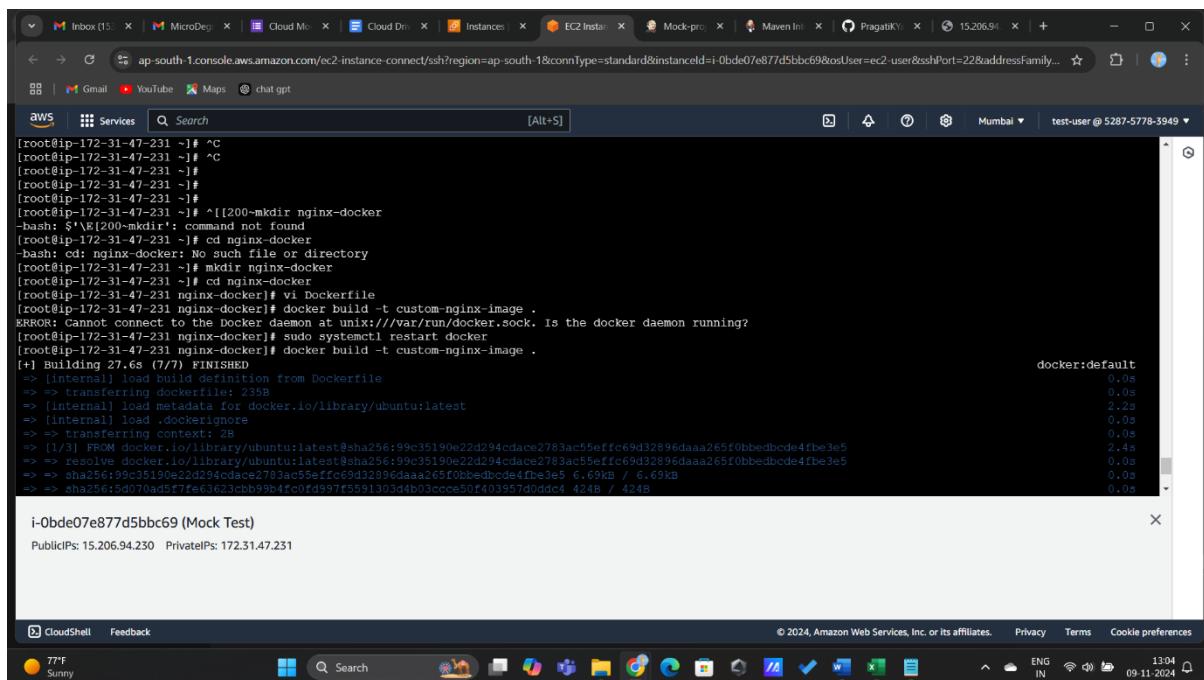
Assignment 15

Creating a Docker Compose Configuration:

Write a Docker Compose YAML file (docker-compose.yml) to define a multi-container application setup.

Specify the services, their configurations, and any necessary links or dependencies between containers.

Use the docker-compose up command to start the containers defined in the Docker



77°F
Sunny

Search

Cloud Monitor Instances EC2 Instances Mock-pro Maven IntelliJ PragatiKY 15.206.94.230 1318 09-11-2024

Inbox (15) MicroDev Cloud Monitor Instances EC2 Instances Mock-pro Maven IntelliJ PragatiKY 15.206.94.230 +

ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-0bde07e877d5bbc69&osUser=ec2-user&sshPort=22&addressFamily=... ☆

Gmail YouTube Maps chat gpt

aws Services Search [Alt+S] Mumbai test-user @ 5287-5778-3949

✓ 90986b8d6e6 Pull complete 8.2s
✓ ae71319cb779 Pull complete 9.1s
✓ ffc89e94fdff88 Pull complete 9.1s
✓ 43d05e938198 Pull complete 16.5s
✓ 064bd2d29fbfa Pull complete 16.5s
✓ df9a4d85569b Pull complete 16.6s
✓ web Pulled 9.3s
✓ a480a496ba95 Pull complete 3.2s
✓ f3ace1b8ce45 Pull complete 6.1s
✓ l1d6fd8d0e8a7 Pull complete 6.2s
✓ f1091da6fd5c Pull complete 6.2s
✓ 40eea07b53d8 Pull complete 6.2s
✓ 6476794e50f4 Pull complete 6.3s
✓ 70850b3e6c62 Pull complete 6.3s

[+] Running 2/3
✓ Network docker-compose-example_app-network Created 0.2s
✓ Container mysql-container Started 1.3s
: Container nginx-container Starting 0.6s

Error response from daemon: driver failed programming external connectivity on endpoint nginx-container (2201b81c47612c25e4e33617ec5e4db3d5aa72021e190196582115
c39e6bc74): Error starting userland proxy: listen tcp4 0.0.0.0:8080: bind: address already in use

[root@ip-172-31-47-231 docker-compose-example]# docker-compose ps
[WARN[0000] /root/docker-compose-example/docker-compose.yaml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion
NAME IMAGE COMMAND SERVICE CREATED STATUS PORTS
mysql-container mysql:5.7 "docker-entrypoint.s..." db About a minute ago Up About a minute 3306/tcp, 33060/tcp
[root@ip-172-31-47-231 docker-compose-example]#

i-0bde07e877d5bbc69 (Mock Test)

Public IPs: 15.206.94.230 Private IPs: 172.31.47.231

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preference

The screenshot shows the Jenkins dashboard. On the left, there's a sidebar with links for 'New Item', 'Build History', 'Manage Jenkins', and 'My Views'. Below that is a 'Build Queue' section stating 'No builds in the queue.' To the right, a main panel displays a table with one row for 'Mock-project'. The columns are 'S' (Status), 'W' (Workflow), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. The 'Mock-project' row shows a green circle icon for status, a yellow sun icon for workflow, the name 'Mock-project', 'Last Success' at '20 min', 'N/A' for last failure, and '7 ms' for last duration. At the bottom, there's a 'Build Executor Status' section showing '0/2' executors. The top of the screen has a navigation bar with various icons and a search bar.

```

Inbox (15) X Gmail YouTube Maps chat gpt
Cloud Mo X Instances X EC2 Insta X Mock-pro X Maven Int X PragatiKY X 15.206.94.230 X + 
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-0bde07e877d5bbc69&osUser=ec2-user&sshPort=22&addressFamily=IPv4
aws Services Search [Alt+S]
90986b08de6 Pull complete
ae71319cb79 Pull complete
ffc89e94df88 Pull complete
43d05e938198 Pull complete
064b2d298fba Pull complete
df9a4d85569b Pull complete
✓ web Pulled
✓ a4804a96ba95 Pull complete
f3ace1b6ce45 Pull complete
11d6ffd0e0a7 Pull complete
✓ f1091da6fd5c Pull complete
✓ 40cea07b53d8 Pull complete
✓ e476794e5f14 Pull complete
✓ 70850b3ec0b2 Pull complete
[*] Running 2/3
✓ Network docker-compose-example_app-network Created
  |> container mysql-container Started
  |> container nginx-container Starting
Error response from daemon: driver failed programming external connectivity on endpoint nginx-container (2201b81c47612c25e4e33617ec5e4dbb3d5aa72021e190196582115
c93e6bc74): Error starting userland proxy: listen tcp4 0.0.0.0:8080: bind: address already in use
[root@ip-172-31-47-231 docker-compose-example]# docker-compose ps
WARN[0000] /root/docker-compose-example/docker-compose.yaml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confus
ing behavior
NAME           IMAGE          COMMAND       SERVICE      CREATED        STATUS        PORTS
mysql-container mysql:5.7   "docker-entrypoint.s..."   db          About a minute ago   Up About a minute   3306/tcp, 33060/tcp
[root@ip-172-31-47-231 docker-compose-example]#

```

i-0bde07e877d5bbc69 (Mock Test)
PublicIPs: 15.206.94.230 PrivateIPs: 172.31.47.231

Assignment 13

Working with Docker Images

- Pull the latest `httpd` image.
- Pull the latest `alpine` image.
- verify images pulled and create 2 containers in each server

```

Inbox (15) X Gmail YouTube Maps chat gpt
Cloud Mo X Instances X EC2 Insta X Mock-pro X Maven Int X PragatiKY X 15.206.94.230 X + 
ap-south-1.console.aws.amazon.com/ec2-instance-connect/ssh?region=ap-south-1&connType=standard&instanceId=i-0bde07e877d5bbc69&osUser=ec2-user&sshPort=22&addressFamily=IPv4
aws Services Search [Alt+S]
Digest: sha256:beefdb081adaed2915566fde36db9db0b524eb737fc57cd1367effd16dc0d0d6
Status: Downloaded newer image for alpine:latest
docker.io/library/alpine:latest
[root@ip-172-31-47-231 docker-compose-example]# docker images
REPOSITORY          TAG      IMAGE ID      CREATED             SIZE
custom-nginx-image latest   d5c31fd2d92b  20 minutes ago   139MB
nginx              latest   3b25b6d2ea82  5 weeks ago      192MB
alpine              latest   91ef04a61f39  2 months ago     7.8MB
httpd              latest   1b0f11fa154f  3 months ago     149MB
mysql              5.7     5107333e0a8  11 months ago    501MB
[root@ip-172-31-47-231 docker-compose-example]# docker run -d --name httpd-container1 httpd:latest
fce962c29ea516beaf88b767b016fffae8e94d0e5cb36f927e3bc90d123a
[root@ip-172-31-47-231 docker-compose-example]# docker run -d --name httpd-container2 httpd:latest
03fcf88e9d4f17c801fc202ea5c24e0cd890a6eb920a65fe2f079b501c67275
[root@ip-172-31-47-231 docker-compose-example]# docker run -d --name alpine-container1 alpine:latest sleep 3600
ce7cf560e6ba01f4249c0b8f14eb816f73a4cf2a7c0ledf3388e72b157c5dc27a
[root@ip-172-31-47-231 docker-compose-example]# docker run -d --name alpine-container2 alpine:latest sleep 3600
4c5937bbe2bf97d1293707561894d029b74cd526e252162844296308934c0
[root@ip-172-31-47-231 docker-compose-example]# docker ps
CONTAINER ID   IMAGE          COMMAND       CREATED        STATUS        PORTS      NAMES
4c5937bbe2bf  alpine:latest  "sleep 3600"  9 seconds ago  Up 9 seconds   alpine-container2
ce7cf560e6ba  alpine:latest  "sleep 3600"  20 seconds ago Up 20 seconds   alpine-container1
03fcf88e9d4f  httpd:latest   "httpd-foreground" 32 seconds ago Up 31 seconds  80/tcp     httpd-container2
fce962c29ea5  httpd:latest   "httpd-foreground" 47 seconds ago Up 46 seconds  80/tcp     httpd-container1
b4bbc671190d0  mysql:5.7    "docker-entrypoint.s..." 10 minutes ago Up 10 minutes  3306/tcp, 33060/tcp  mysql-container
[root@ip-172-31-47-231 docker-compose-example]#

```

i-0bde07e877d5bbc69 (Mock Test)
PublicIPs: 15.206.94.230 PrivateIPs: 172.31.47.231

```
a480a496ba95: Already exists
3a2663e66670: Pull complete
4f4fb700ef54: Pull complete
dbde712f81fb: Pull complete
867b2ea3628d: Pull complete
6bd9d3710aae: Pull complete
Digest: sha256:bbea29057f25d9543e6a96a8e3cc7c7c937206d20eab2323f478fdb2469d536d
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
[root@ip-172-31-47-231 docker-compose-example]#
[root@ip-172-31-47-231 docker-compose-example]# docker pull alpine:latest
latest: Pulling from library/alpine
43c4264eeed91: Pull complete
Digest: sha256:beefdb8a1da6d2915566fde36db9db0b524eb737fc57cd1367effd16dc0d06d
Status: Downloaded newer image for alpine:latest
docker.io/library/alpine:latest
[root@ip-172-31-47-231 docker-compose-example]# docker images
REPOSITORY          TAG      IMAGE ID      CREATED             SIZE
custom-nginx-image  latest   d5681f8d292b  20 minutes ago   133MB
nginx               latest   3b25b682ea82  5 weeks ago       192MB
alpine              latest   91ef0af61f39  2 months ago      7.8MB
httpd               latest   1bcf11fa154f  3 months ago      149MB
mysql               5.7     510733e08a8  11 months ago     501MB
[root@ip-172-31-47-231 docker-compose-example]# docker run -d --name httpd-container1 httpd:latest
i-0bde07e877d5bbc69 (Mock Test)
PublicIPs: 15.206.94.230 PrivateIPs: 172.31.47.231
```

```
a480a496ba95: Already exists
3a2663e66670: Pull complete
4f4fb700ef54: Pull complete
dbde712f81fb: Pull complete
867b2ea3628d: Pull complete
6bd9d3710aae: Pull complete
Digest: sha256:bbea29057f25d9543e6a96a8e3cc7c7c937206d20eab2323f478fdb2469d536d
Status: Downloaded newer image for httpd:latest
docker.io/library/httpd:latest
[root@ip-172-31-47-231 docker-compose-example]#
[root@ip-172-31-47-231 docker-compose-example]# docker pull alpine:latest
latest: Pulling from library/alpine
43c4264eeed91: Pull complete
Digest: sha256:beefdb8a1da6d2915566fde36db9db0b524eb737fc57cd1367effd16dc0d06d
Status: Downloaded newer image for alpine:latest
docker.io/library/alpine:latest
[root@ip-172-31-47-231 docker-compose-example]# docker images
REPOSITORY          TAG      IMAGE ID      CREATED             SIZE
custom-nginx-image  latest   d5681f8d292b  20 minutes ago   133MB
nginx               latest   3b25b682ea82  5 weeks ago       192MB
alpine              latest   91ef0af61f39  2 months ago      7.8MB
httpd               latest   1bcf11fa154f  3 months ago      149MB
mysql               5.7     510733e08a8  11 months ago     501MB
[root@ip-172-31-47-231 docker-compose-example]# 
i-0bde07e877d5bbc69 (Mock Test)
PublicIPs: 15.206.94.230 PrivateIPs: 172.31.47.231
```

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

77°F Sunny ENG IN 09-11-2024 13:23

A screenshot of a terminal window titled "aws Services" with a search bar and a "Search [Alt+S]" button. The terminal displays a series of Docker commands being run on an EC2 instance. The commands include pulling images from Docker Hub, creating containers for Nginx, Alpine Linux, and MySQL, and running them with specific ports and names. The output shows the creation of several containers, each with its status, ports, and names. The terminal window has a dark theme and is set against a background of a Microsoft Windows desktop environment.