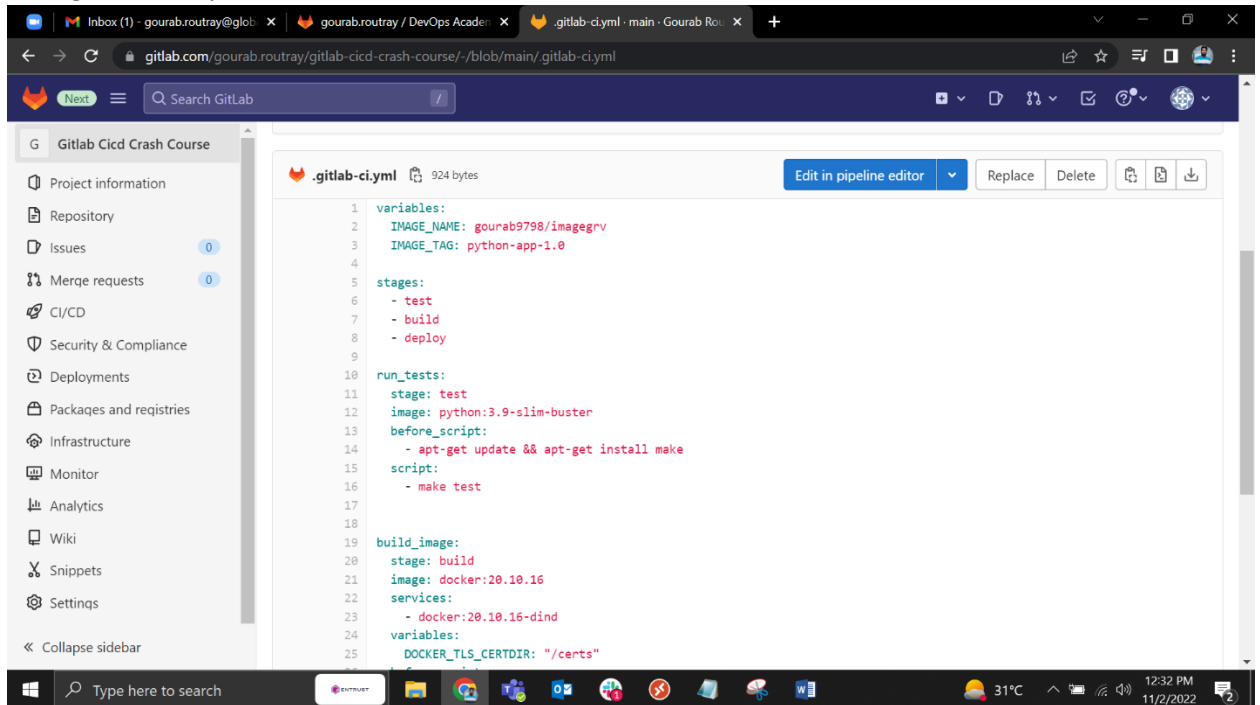


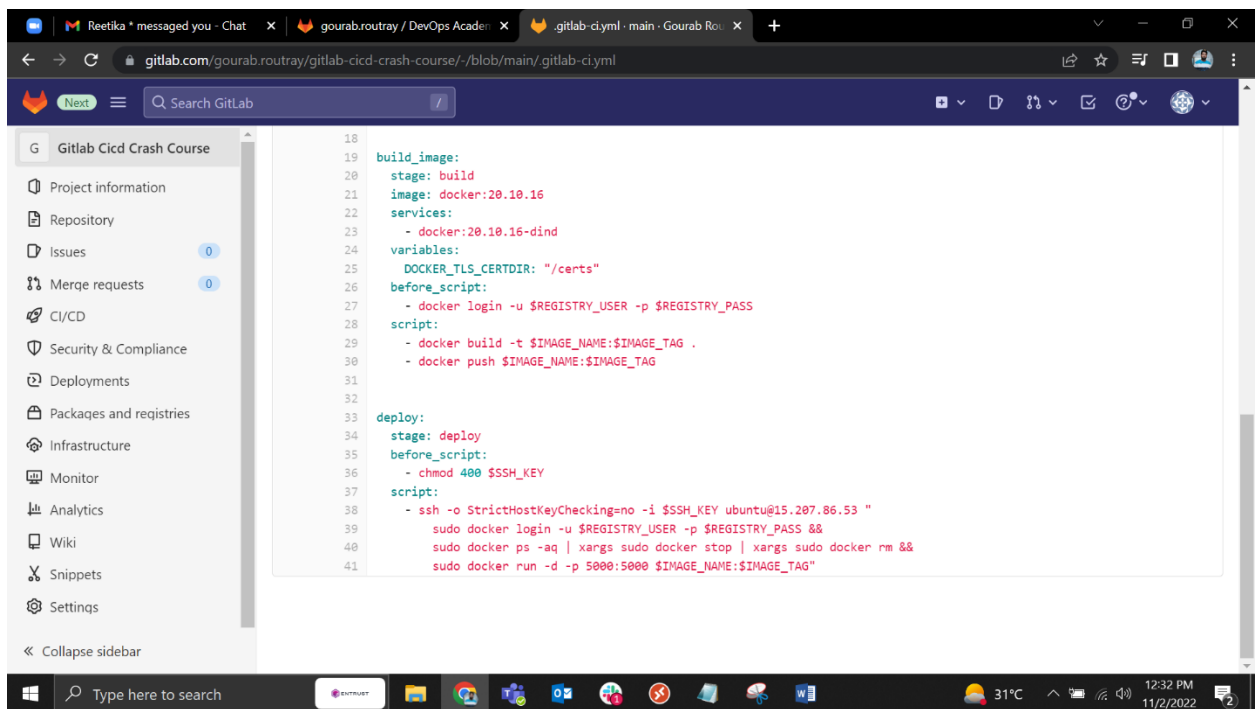
ASSIGNMENT-5

1- Configure the CI.yml file as follows



The screenshot shows the GitLab web interface for a repository named "Gitlab Cid Crash Course". The left sidebar contains navigation links for Project information, Repository, Issues, Merge requests, CI/CD, Security & Compliance, Deployments, Packages and registries, Infrastructure, Monitor, Analytics, Wiki, Snippets, and Settings. The main area displays the ".gitlab-ci.yml" file with 924 bytes. The file content is as follows:

```
1 variables:
2   IMAGE_NAME: gourab9798/imagegrv
3   IMAGE_TAG: python-app-1.0
4
5 stages:
6   - test
7   - build
8   - deploy
9
10 run_tests:
11   stage: test
12   image: python:3.9-slim-buster
13   before_script:
14     - apt-get update && apt-get install make
15   script:
16     - make test
17
18
19 build_image:
20   stage: build
21   image: docker:20.10.16
22   services:
23     - docker:20.10.16-dind
24   variables:
25     DOCKER_TLS_CERTDIR: "/certs"
```



The screenshot shows the same GitLab web interface, but the ".gitlab-ci.yml" file has been updated. The file content is now as follows:

```
18
19 build_image:
20   stage: build
21   image: docker:20.10.16
22   services:
23     - docker:20.10.16-dind
24   variables:
25     DOCKER_TLS_CERTDIR: "/certs"
26   before_script:
27     - docker login -u $REGISTRY_USER -p $REGISTRY_PASS
28   script:
29     - docker build -t $IMAGE_NAME:$IMAGE_TAG .
30     - docker push $IMAGE_NAME:$IMAGE_TAG
31
32
33 deploy:
34   stage: deploy
35   before_script:
36     - chmod 400 $SSH_KEY
37   script:
38     - ssh -o StrictHostKeyChecking=no -i $SSH_KEY ubuntu@15.207.86.53 "
39       sudo docker login -u $REGISTRY_USER -p $REGISTRY_PASS &&
40       sudo docker ps -aq | xargs sudo docker stop | xargs sudo docker rm &&
41       sudo docker run -d -p 5000:5000 $IMAGE_NAME:$IMAGE_TAG"
```

ASSIGNMENT-5

2- Add the variables in order to configure the build

The screenshot shows the GitLab CI/CD settings page for a project. The left sidebar contains navigation links: Infrastructure, Monitor, Analytics, Wiki, Snippets, Settings (selected), General, Integrations, Webhooks, Repository, Merge requests, CI/CD (selected), Packages and registries, Pages, and Monitor. The main content area is titled 'Variables can be:' and lists two types of variables: Protected (Only exposed to protected branches or protected tags) and Masked (Hidden in job logs. Must match masking requirements. [Learn more.](#)). Below this, a message states: 'Environment variables are configured by your administrator to be **protected** by default.' A table lists the configured variables:

Type	Key	Value	Options	Environments
Variable	REGISTRY_PASS	*****	Protected	All (default)
Variable	REGISTRY_USER	*****	Protected	All (default)
File	SSH_KEY	*****	Protected	All (default)

Below the table are buttons for 'Add variable' and 'Reveal values'. At the bottom, there is a section for 'Pipeline triggers' with an 'Expand' button.

3- Commit the changes and the build starts executing automatically

The screenshot shows the GitLab CI/CD job execution page for the 'build_image' job. The left sidebar contains navigation links: Project information, Repository, Issues (0), Merge requests (0), CI/CD (selected), Pipelines, Editor, Jobs (selected), Schedules, Security & Compliance, Deployments, Packages and registries, Infrastructure, and Monitor. The main content area displays the job log for 'build_image' (ID: 3260793917). The log shows the following steps:

```
308 a8f9e3c1138b: Preparing
309 74f8a626d86f: Preparing
310 c6b69518a1a5: Waiting
311 2bd15eb2618e: Waiting
312 d0569174cd96: Waiting
313 a8f9e3c1138b: Waiting
314 74f8a626d86f: Waiting
315 301f64924704: Pushed
316 8f951ddd725c: Pushed
317 f568ce070f70: Pushed
318 8822acd6a252: Pushed
319 2bd15eb2618e: Mounted from library/python
320 c6b69518a1a5: Mounted from library/python
321 a8f9e3c1138b: Mounted from library/python
322 d0569174cd96: Mounted from library/python
323 74f8a626d86f: Mounted from library/python
324 98c57edd29d9: Pushed
325 python-app-1.0: digest: sha256:b73bb17d5f0ff8a626f950eead48a64adfea18c8662eaf9415b
30f10f4409c9e size: 2412
326 Cleaning up project directory and file based variables
327 Job succeeded
```

On the right side, the job details are shown: Duration: 1 minute 23 seconds, Finished: in 2 minutes, Queued: 0 seconds, Timeout: 1h (from project), Runner: #12270848 (ns46NMmJ) 2-green.shared.runners-manager.gitlab.com/default. Below this, the commit hash 68cf85d7 is shown, along with a dropdown menu for the pipeline (build) and a button to view the job details (build_image).

ASSIGNMENT-5

- 4- The image is fetched from the docker hub and deployed in the ec-2 instance

The screenshot shows the GitLab CI/CD interface for a project named 'Gitlab Cid Crash Course'. The 'Jobs' tab is selected, displaying a list of jobs for the 'deploy' pipeline. The jobs are as follows:

Job ID	Status	Message
61 eaa89787764f	Pull complete	
62 9c9b5d1149d6	Verifying Checksum	
63 9c9b5d1149d6	Download complete	
64 0f2d713f28f8	Pull complete	
65 feee6dd7fa43	Pull complete	
66 4a1bf4dcccce2	Pull complete	
67 4378398ea058	Pull complete	
68 d8fa7bc9e58a	Verifying Checksum	
69 d8fa7bc9e58a	Download complete	
70 9c9b5d1149d6	Pull complete	
71 b5e91146fe35	Download complete	
72 bebb012f18ee	Verifying Checksum	
73 bebb012f18ee	Download complete	
74 bebb012f18ee	Pull complete	
75 d8fa7bc9e58a	Pull complete	
76 b5e91146fe35	Pull complete	
77	Digest: sha256:2d3c9ed8aee235b8db9643589edf0626180b37903628ef23b21117571e867b87	
78	Status: Downloaded newer image for gourab9798/imagegrv:python-app-1.0	
79 4c2001b491d728b238fdef346bce0a2968a8c1520c00a5212e57d15725aa57e8		
81	Cleaning up project directory and file based variables	
83	Job succeeded	

The right sidebar shows the 'deploy' pipeline details:

- Duration: 26 seconds
- Finished: in 1 minute
- Queued: 0 seconds
- Timeout: 1h (from project)
- Runner: #12270859 (xS6Vzpvo) 5-green.shared.runners-manager.gitlab.com/default
- Commit: 4a7813a6
- Update .gitlab-ci.yml files
- Pipeline #683145752 for main
- Jobs: deploy

- 5- The container hello-world is running before the deployment

The screenshot shows the AWS Management Console for an EC2 instance. The terminal output displays the following steps:

3. The Docker daemon created a new container from that image which runs the executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it to your terminal.

To try something more ambitious, you can run an Ubuntu container with:

```
$ docker run -it ubuntu bash
```

Share images, automate workflows, and more with a free Docker ID:
<https://hub.docker.com/>

For more examples and ideas, visit:
<https://docs.docker.com/get-started/>

ubuntu@ip-172-31-43-230:~\$ docker ps -a

```
Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/containers/json?all=1": dial unix /var/run/docker.sock: connect: permission denied
ubuntu@ip-172-31-43-230:~$ sudo docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
6bce6fa66d5d	hello-world	"/hello"	12 seconds ago	Exited (0) 12 seconds ago		flamboyant_turing

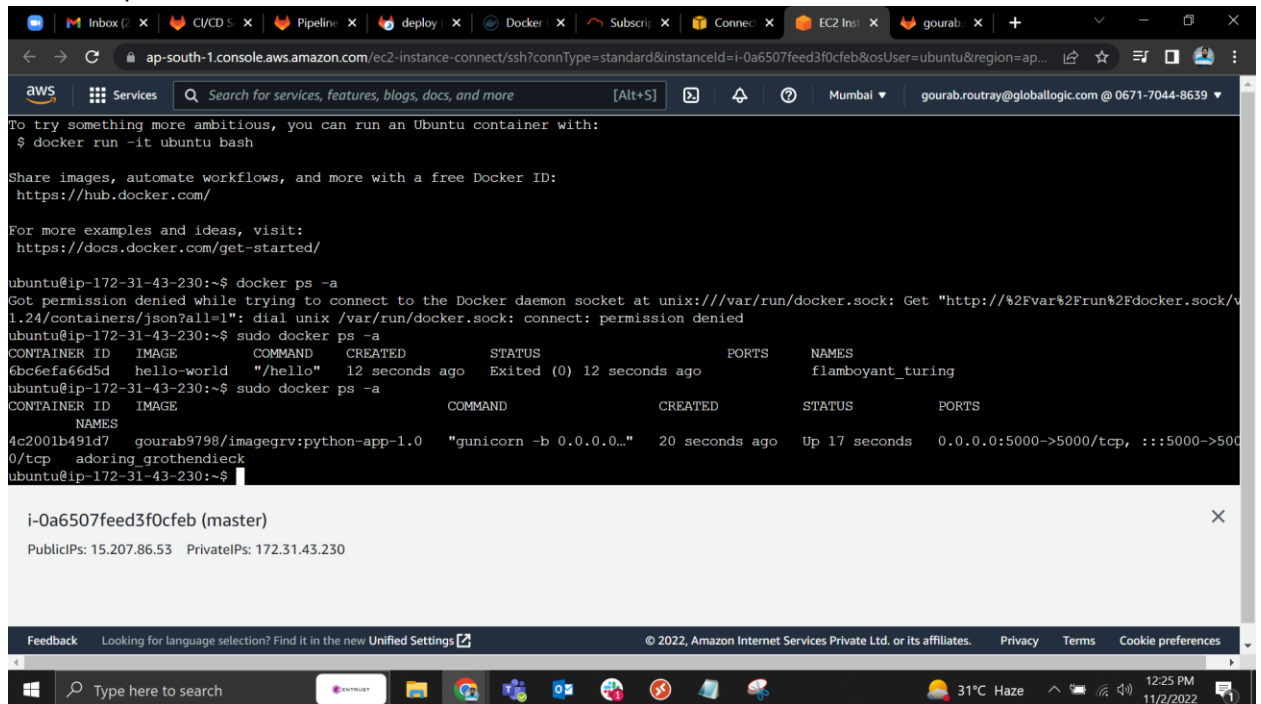
ubuntu@ip-172-31-43-230:~\$

i-0a6507feed3f0cfcb (master)

PublicIPs: 15.207.86.53 PrivateIPs: 172.31.43.230

ASSIGNMENT-5

- 6- The container hello-world got terminated and the deployed container is running as mentioned in the script



```
To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

ubuntu@ip-172-31-43-230:~$ docker ps -a
Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/containers/json?all=1": dial unix /var/run/docker.sock: connect: permission denied
ubuntu@ip-172-31-43-230:~$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
6bc6efa66d5d   hello-world    "/hello"                 12 seconds ago Exited (0)    12 seconds ago          flamboyant_turing
ubuntu@ip-172-31-43-230:~$ sudo docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
4c2001b491d7   gourab9798/imagegrv:python-app-1.0  "gunicorn -b 0.0.0.0..."  20 seconds ago Up 17 seconds  0.0.0.0:5000->5000/tcp, :::5000->5000/tcp   adoring_grothendieck
ubuntu@ip-172-31-43-230:~$
```

i-0a6507feed3f0cfeb (master)

PublicIPs: 15.207.86.53 PrivateIPs: 172.31.43.230

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Internet Services Private Ltd. or its affiliates. Privacy Terms Cookie preferences

Type here to search

31°C Haze 12:25 PM 11/2/2022