یلدا جعفری

به‌نام خدا

تمرین اول:

```
devops@ubuntu-server:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor
 -o /usr/share/keyrings/docker.gpg
devops@ubuntu-server:~$ echo \
> "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker.gpg] \
> https://download.docker.com/linux/ubuntu \
> $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

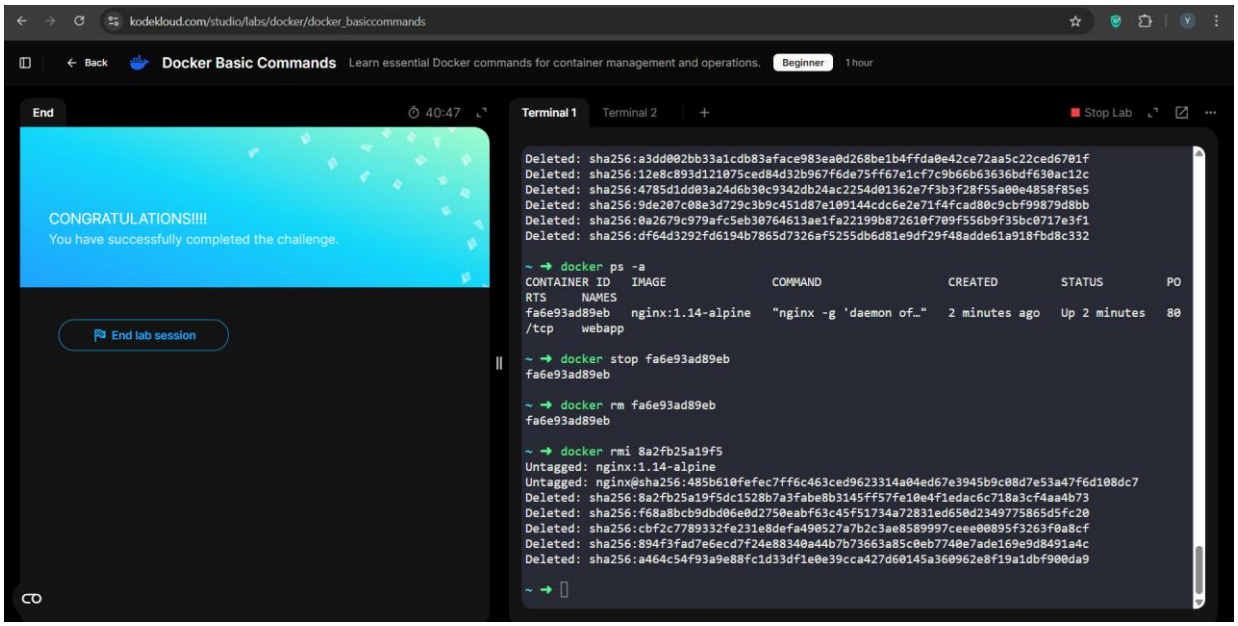
```
devops@ubuntu-server:~$ cat /etc/apt/sources.list.d/docker.list
deb [arch=amd64 signed-by=/usr/share/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu j
ammy stable
devops@ubuntu-server:~$
```

```
devops@ubuntu-server:~$ sudo systemctl start docker
devops@ubuntu-server:~$ sudo systemctl enable docker
Synchronizing state of docker.service with SysV service script with /lib/systemd/systemd-sysv-instal
l.
Executing: /lib/systemd/systemd-sysv-install enable docker
devops@ubuntu-server:~$ docker --version
Docker version 28.4.0, build d8eb465
devops@ubuntu-server:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
17eec7bbc9d7: Pull complete
Digest: sha256:54e66cc1dd1fcb1c3c58bd8017914dbed8701e2d8c74d9262e26bd9cc1642d31
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 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/

devops@ubuntu-server:~$
```

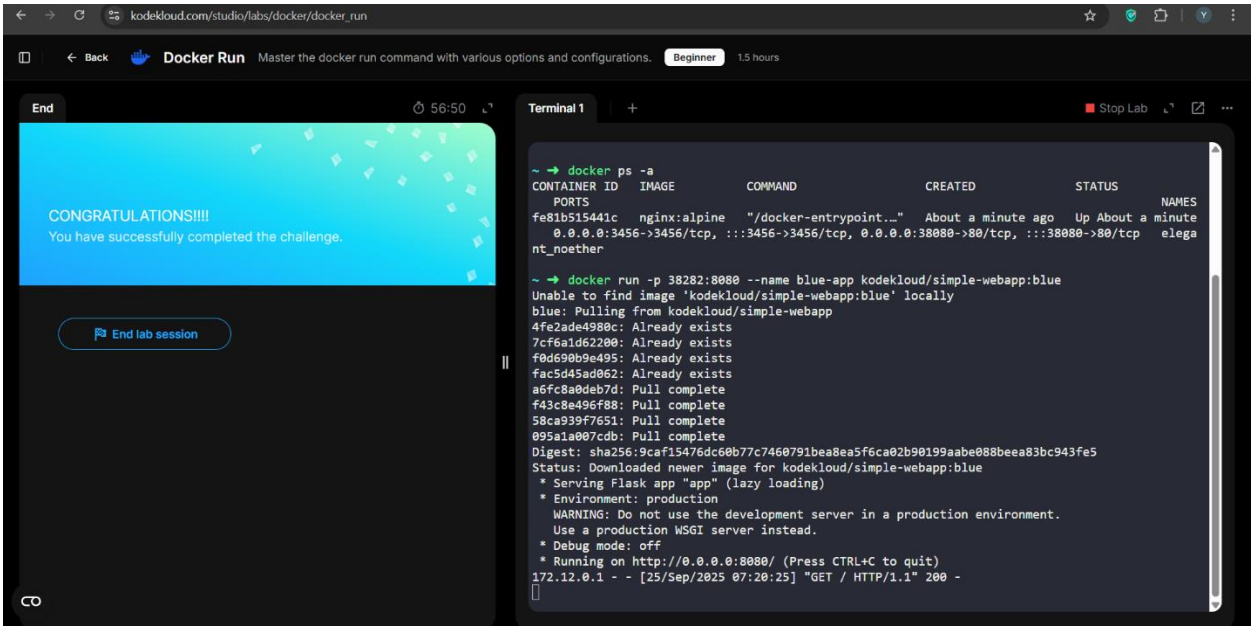
یلدا جعفری

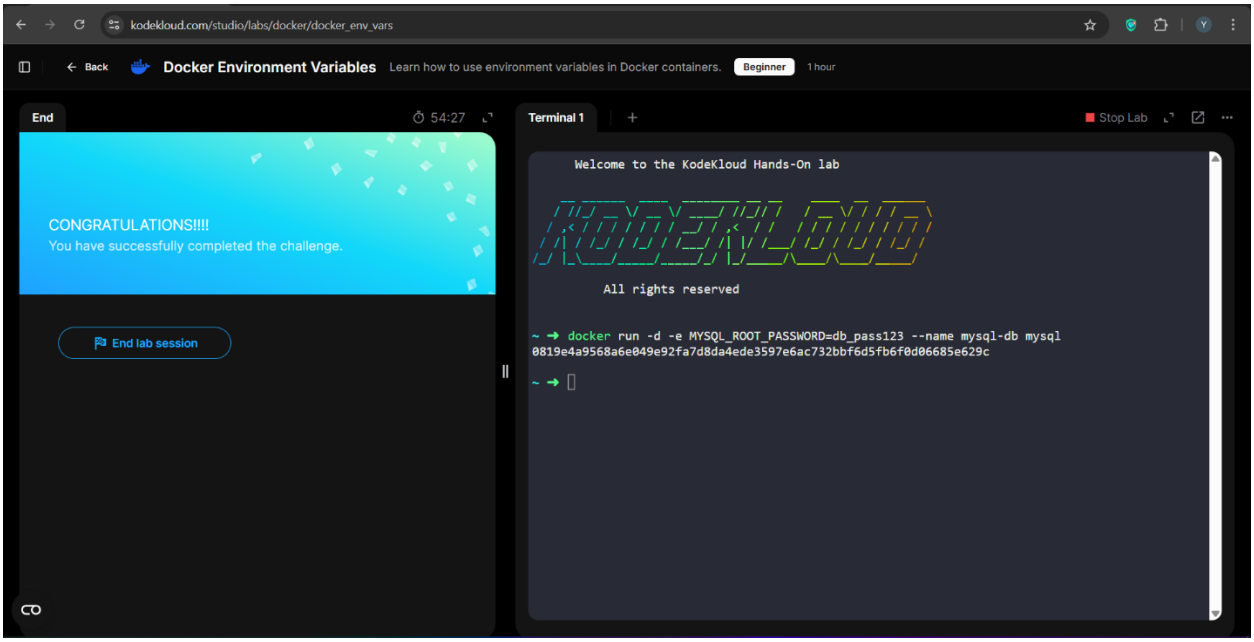
```
devops@ubuntu-server:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
devops@ubuntu-server:~$ sudo docker ps -a
CONTAINER ID   IMAGE         COMMAND    CREATED         STATUS                     PORTS     NAMES
c306c3e1cd99   hello-world   "/hello"   2 minutes ago   Exited (0) 2 minutes ago             vigilan
t_mendeleev
devops@ubuntu-server:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED       SIZE
hello-world   latest    1b44b5a3e06a   6 weeks ago   10.1kB
devops@ubuntu-server:~$ sudo docker run -it ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
953cdd413371: Pull complete
Digest: sha256:353675e2a41babd526e2b837d7ec780c2a05bca0164f7ea5dbbd433d21d166fc
Status: Downloaded newer image for ubuntu:latest
root@bc5f3c372f61:/# sudo docker ps
bash: sudo: command not found
root@bc5f3c372f61:/# exit
exit
devops@ubuntu-server:~$ docker ps
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock
: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.51/containers/json": dial unix /var/run/docker.sock: con
nect: permission denied
devops@ubuntu-server:~$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
devops@ubuntu-server:~$ sudo docker ps -a
CONTAINER ID   IMAGE         COMMAND    CREATED          STATUS                       PORTS     NAM
ES
bc5f3c372f61   ubuntu        "bash"     51 seconds ago   Exited (127) 19 seconds ago           heu
ristic_yalow
c306c3e1cd99   hello-world   "/hello"   4 minutes ago    Exited (0) 4 minutes ago              vig
ilant_mendeleev
devops@ubuntu-server:~$ sudo docker images
REPOSITORY    TAG       IMAGE ID       CREATED       SIZE
ubuntu        latest    6d79abd4c962   2 weeks ago   78.1MB
hello-world   latest    1b44b5a3e06a   6 weeks ago   10.1kB
devops@ubuntu-server:~$ _
```

تمرین دوم:

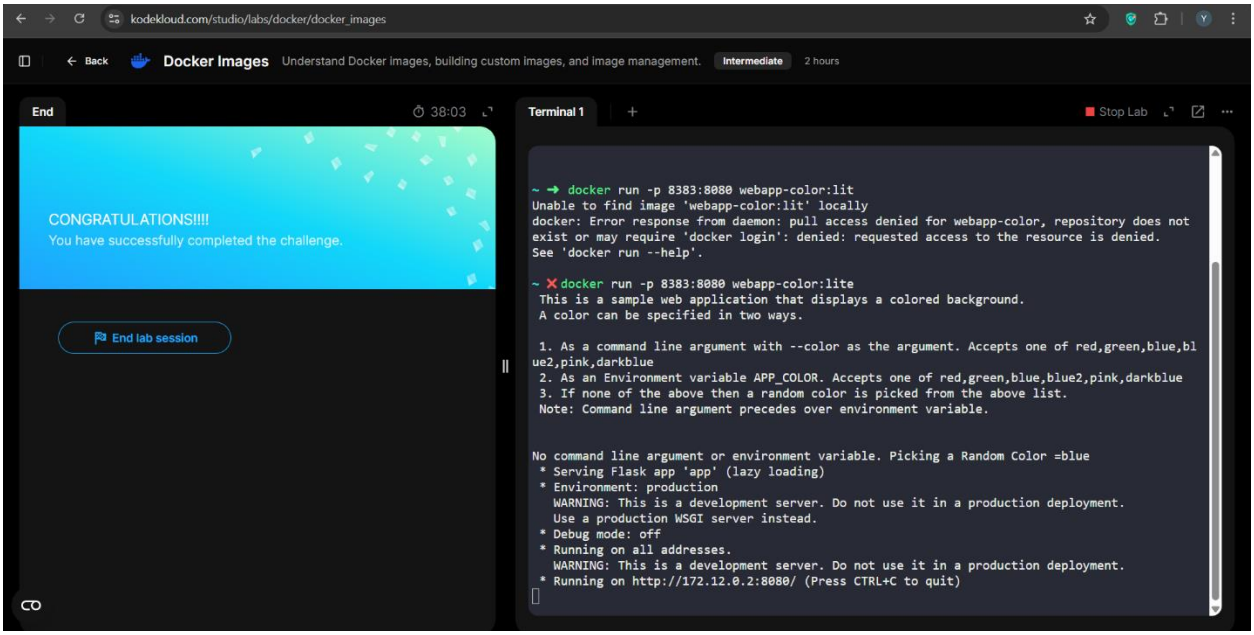


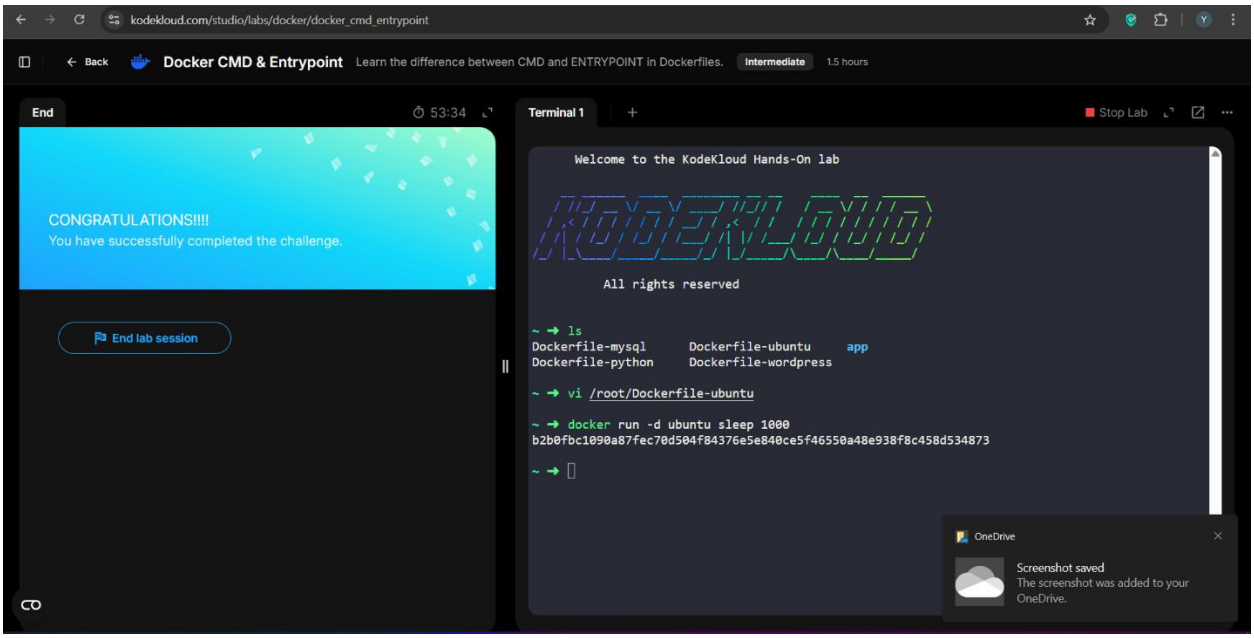
یلدا جعفری


kodekloud.com/studio/labs/docker/docker_run
Back
Docker Run
Master the docker run command with various options and configurations.
Beginner
1.5 hours
End
56:50
CONGRATULATIONS!!!!
You have successfully completed the challenge.
End lab session
Terminal 1
Stop Lab
~ ➜ docker ps -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
fe81b515441c nginx:alpine "/docker-entrypoint.…" About a minute ago Up About a minute 0.0.0.0:3456->3456/tcp, :::3456->3456/tcp, 0.0.0.0:38080->80/tcp, :::38080->80/tcp elegant_noether
~ ➜ docker run -p 38282:8080 --name blue-app kodekloud/simple-webapp:blue
Unable to find image 'kodekloud/simple-webapp:blue' locally
blue: Pulling from kodekloud/simple-webapp
4fe2ade4980c: Already exists
7cf6a1d62200: Already exists
f0d690b9e495: Already exists
fac5d45ad062: Already exists
a6fc8a0deb7d: Pull complete
f43c8e496f88: Pull complete
58ca939f7651: Pull complete
095a1a007cdb: Pull complete
Digest: sha256:9caf15476dc60b77c7460791bea8ea5f6ca02b90199aabe088beea83bc943fe5
Status: Downloaded newer image for kodekloud/simple-webapp:blue
* Serving Flask app "app" (lazy loading)
* Environment: production
WARNING: Do not use the development server in a production environment.
Use a production WSGI server instead.
* Debug mode: off
* Running on http://0.0.0.0:8080/ (Press CTRL+C to quit)
172.12.0.1 - - [25/Sep/2025 07:20:25] "GET / HTTP/1.1" 200 -

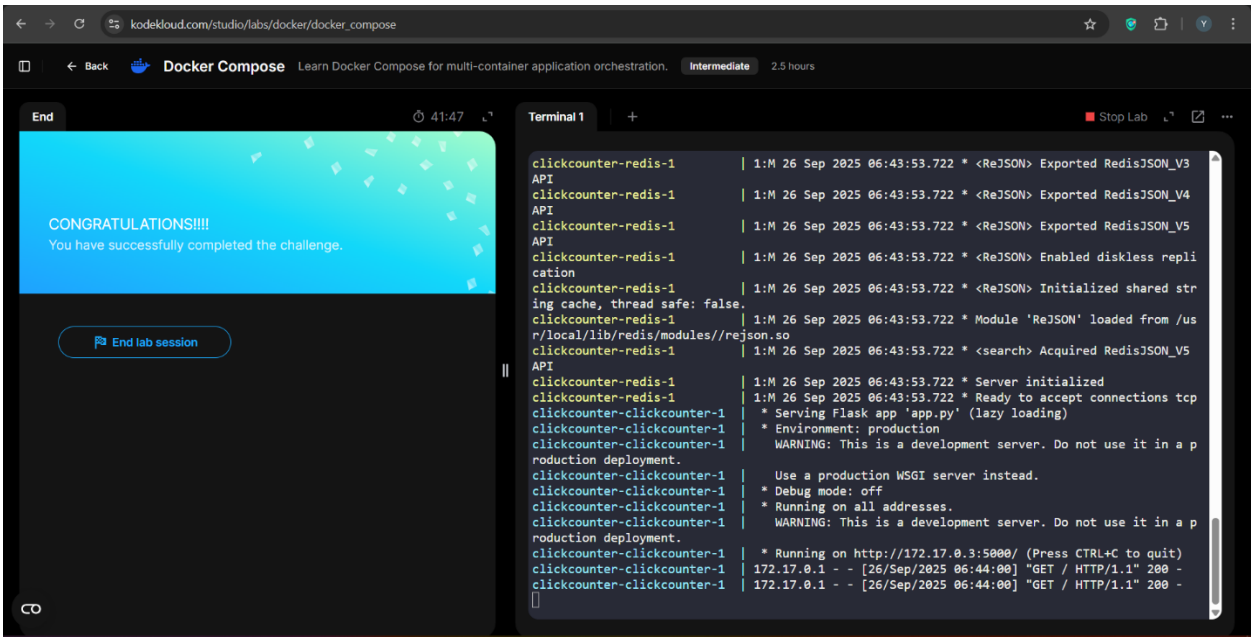

kodekloud.com/studio/labs/docker/docker_env_vars
Back
Docker Environment Variables
Learn how to use environment variables in Docker containers.
Beginner
1 hour
End
54:27
CONGRATULATIONS!!!!
You have successfully completed the challenge.
End lab session
Terminal 1
Stop Lab
Welcome to the KodeKloud Hands-On lab
All rights reserved
~ ➜ docker run -d -e MYSQL_ROOT_PASSWORD=db_pass123 --name mysql-db mysql
0819e4a9568a6e049e92fa7d8da4ede3597e6ac732bbf6d5fb6f0d06685e629c
~ ➜

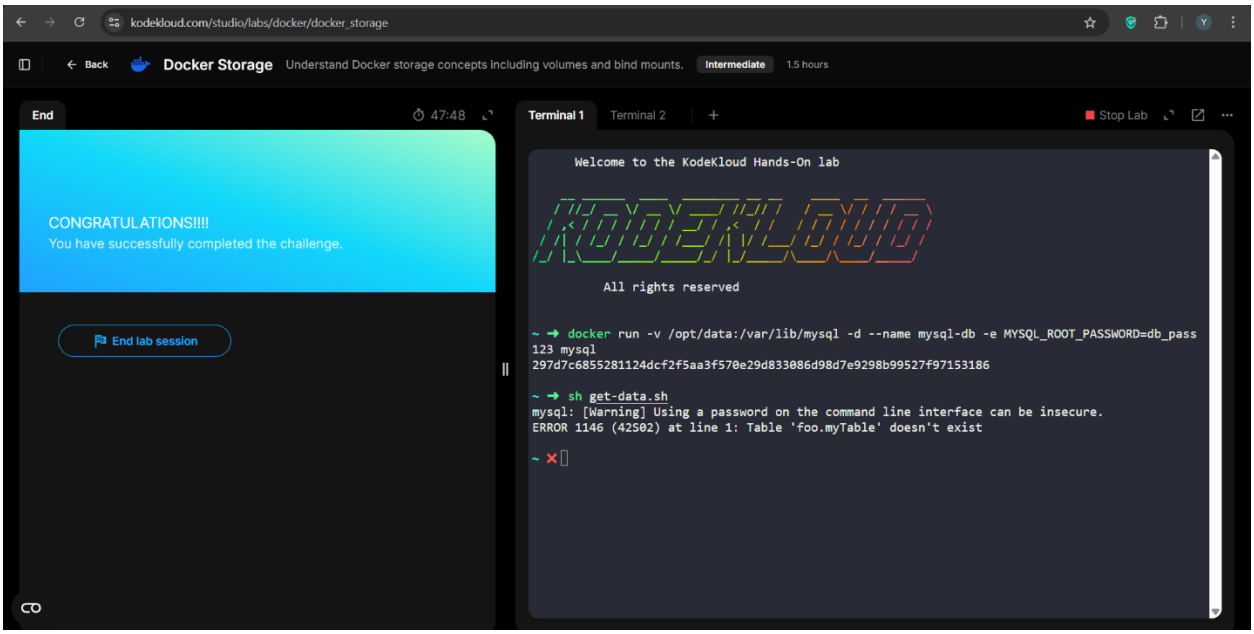
یلدا جعفری


kodekloud.com/studio/labs/docker/docker_images
Back
Docker Images
Understand Docker images, building custom images, and image management.
Intermediate
2 hours
End
38:03
CONGRATULATIONS!!!!
You have successfully completed the challenge.
End lab session
Terminal 1
Stop Lab
~ ➜ docker run -p 8383:8080 webapp-color:lit
Unable to find image 'webapp-color:lit' locally
docker: Error response from daemon: pull access denied for webapp-color, repository does not exist or may require 'docker login': denied: requested access to the resource is denied.
See 'docker run --help'.
~ ✕ docker run -p 8383:8080 webapp-color:lite
This is a sample web application that displays a colored background.
A color can be specified in two ways.
1. As a command line argument with --color as the argument. Accepts one of red,green,blue,blue2,pink,darkblue
2. As an Environment variable APP_COLOR. Accepts one of red,green,blue,blue2,pink,darkblue
3. If none of the above then a random color is picked from the above list.
Note: Command line argument precedes over environment variable.
No command line argument or environment variable. Picking a Random Color =blue
* Serving Flask app 'app' (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on all addresses.
WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://172.12.0.2:8080/ (Press CTRL+C to quit)

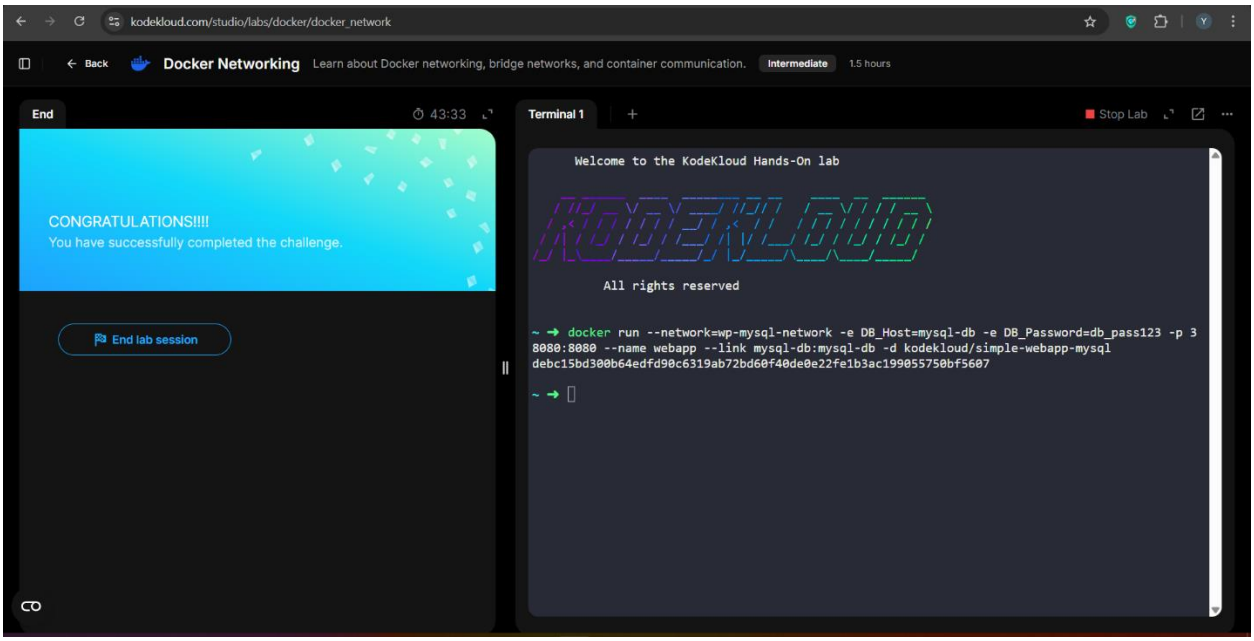

kodekloud.com/studio/labs/docker/docker_cmd_entrypoint
Back
Docker CMD & Entrypoint
Learn the difference between CMD and ENTRYPOINT in Dockerfiles.
Intermediate
1.5 hours
End
53:34
CONGRATULATIONS!!!!
You have successfully completed the challenge.
End lab session
Terminal 1
Stop Lab
Welcome to the KodeKloud Hands-On lab
All rights reserved
~ ➜ ls
Dockerfile-mysql Dockerfile-ubuntu app
Dockerfile-python Dockerfile-wordpress
~ ➜ vi /root/Dockerfile-ubuntu
~ ➜ docker run -d ubuntu sleep 1000
b2b0fbc1090a87fec70d504f84376e5e840ce5f46550a48e938f8c458d534873
~ ➜
OneDrive
Screenshot saved
The screenshot was added to your OneDrive.

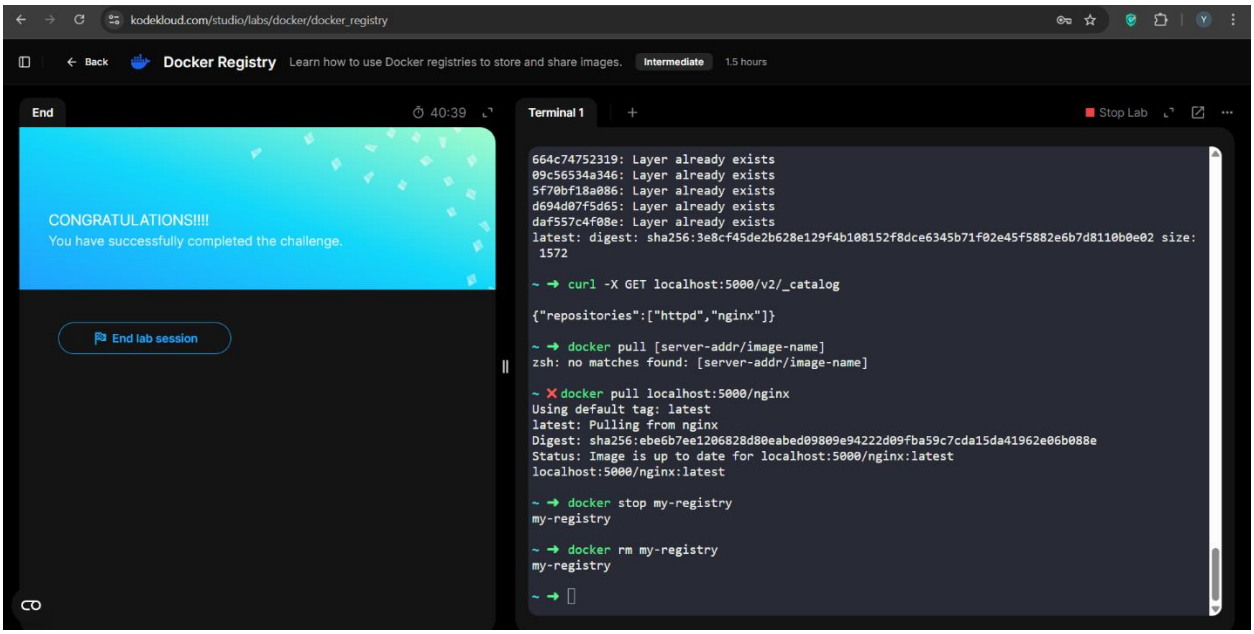
یلدا جعفری


kodekloud.com/studio/labs/docker/docker_compose
Back
Docker Compose
Learn Docker Compose for multi-container application orchestration.
Intermediate
2.5 hours
End
41:47
CONGRATULATIONS!!!!
You have successfully completed the challenge.
End lab session
Terminal 1
Stop Lab
clickcounter-redis-1 | 1:M 26 Sep 2025 06:43:53.722 * <ReJSON> Exported RedisJSON_V3 API
clickcounter-redis-1 | 1:M 26 Sep 2025 06:43:53.722 * <ReJSON> Exported RedisJSON_V4 API
clickcounter-redis-1 | 1:M 26 Sep 2025 06:43:53.722 * <ReJSON> Exported RedisJSON_V5 API
clickcounter-redis-1 | 1:M 26 Sep 2025 06:43:53.722 * <ReJSON> Enabled diskless replication
clickcounter-redis-1 | 1:M 26 Sep 2025 06:43:53.722 * <ReJSON> Initialized shared string cache, thread safe: false.
clickcounter-redis-1 | 1:M 26 Sep 2025 06:43:53.722 * Module 'ReJSON' loaded from /usr/local/lib/redis/modules//rejson.so
clickcounter-redis-1 | 1:M 26 Sep 2025 06:43:53.722 * <search> Acquired RedisJSON_V5 API
clickcounter-redis-1 | 1:M 26 Sep 2025 06:43:53.722 * Server initialized
clickcounter-redis-1 | 1:M 26 Sep 2025 06:43:53.722 * Ready to accept connections tcp
clickcounter-clickcounter-1 | * Serving Flask app 'app.py' (lazy loading)
clickcounter-clickcounter-1 | * Environment: production
clickcounter-clickcounter-1 | WARNING: This is a development server. Do not use it in a production deployment.
clickcounter-clickcounter-1 | Use a production WSGI server instead.
clickcounter-clickcounter-1 | * Debug mode: off
clickcounter-clickcounter-1 | * Running on all addresses.
clickcounter-clickcounter-1 | WARNING: This is a development server. Do not use it in a production deployment.
clickcounter-clickcounter-1 | * Running on http://172.17.0.3:5000/ (Press CTRL+C to quit)
clickcounter-clickcounter-1 | 172.17.0.1 - - [26/Sep/2025 06:44:00] "GET / HTTP/1.1" 200 -
clickcounter-clickcounter-1 | 172.17.0.1 - - [26/Sep/2025 06:44:00] "GET / HTTP/1.1" 200 -


kodekloud.com/studio/labs/docker/docker_storage
Back
Docker Storage
Understand Docker storage concepts including volumes and bind mounts.
Intermediate
1.5 hours
End
47:48
CONGRATULATIONS!!!!
You have successfully completed the challenge.
End lab session
Terminal 1
Terminal 2
Stop Lab
Welcome to the KodeKloud Hands-On lab
All rights reserved
~ ➜ docker run -v /opt/data:/var/lib/mysql -d --name mysql-db -e MYSQL_ROOT_PASSWORD=db_pass123 mysql
297d7c6855281124dcf2f5aa3f570e29d833086d98d7e9298b99527f97153186
~ ➜ sh get-data.sh
mysql: [Warning] Using a password on the command line interface can be insecure.
ERROR 1146 (42S02) at line 1: Table 'foo.myTable' doesn't exist
~ ✗

یلدا جعفری


kodekloud.com/studio/labs/docker/docker_network
Back
Docker Networking
Learn about Docker networking, bridge networks, and container communication.
Intermediate
1.5 hours
End
43:33
CONGRATULATIONS!!!!
You have successfully completed the challenge.
End lab session
Terminal 1
Stop Lab
Welcome to the KodeKloud Hands-On lab
All rights reserved
~ ➜ docker run --network=wp-mysql-network -e DB_Host=mysql-db -e DB_Password=db_pass123 -p 3
8080:8080 --name webapp --link mysql-db:mysql-db -d kodekloud/simple-webapp-mysql
debc15bd300b64edfd90c6319ab72bd60f40de0e22fe1b3ac199055750bf5607
~ ➜


kodekloud.com/studio/labs/docker/docker_registry
Back
Docker Registry
Learn how to use Docker registries to store and share images.
Intermediate
1.5 hours
End
40:39
CONGRATULATIONS!!!!
You have successfully completed the challenge.
End lab session
Terminal 1
Stop Lab
664c74752319: Layer already exists
09c56534a346: Layer already exists
5f70bf18a086: Layer already exists
d694d07f5d65: Layer already exists
daf557c4f08e: Layer already exists
latest: digest: sha256:3e8cf45de2b628e129f4b108152f8dce6345b71f02e45f5882e6b7d8110b0e02 size:
1572
~ ➜ curl -X GET localhost:5000/v2/_catalog
{"repositories":["httpd","nginx"]}
~ ➜ docker pull [server-addr/image-name]
zsh: no matches found: [server-addr/image-name]
~ ✗ docker pull localhost:5000/nginx
Using default tag: latest
latest: Pulling from nginx
Digest: sha256:ebe6b7ee1206828d80eabed09809e94222d09fba59c7cda15da41962e06b088e
Status: Image is up to date for localhost:5000/nginx:latest
localhost:5000/nginx:latest
~ ➜ docker stop my-registry
my-registry
~ ➜ docker rm my-registry
my-registry
~ ➜