

KodeKloud STUDIO

Applications

- KodeKloud Originals
- Free Labs
- LinkedIn Photo
- LinkedIn Optimizer

Navigation

- Courses
- Learning Paths
- Playgrounds
- Job Simulation
- Cohorts
- Notes

KodeKloud Account

- My Account
- Sign up

Docker Environment Variables

Learn how to use environment variables in Docker containers. **Beginner** 45-60 min

1/4

Inspect the environment variables set on the running container and identify the value set to the `APP_COLOR` variable.

pink

blue

yellow

green

```
~ → docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED
STATUS        PORTS                NAMES
bd7b5146d938  kodekloud/simple-webapp  "python app.py"         10 minutes ago
Up 9 minutes   8080/tcp              eloquent_goldwasser

~ → docker exec eloquent_goldwasser printenv
OCI runtime exec failed: exec failed: unable to start container process: exec: "printenv": executable file not found in $PATH: unknown

~ ❌ docker exec eloquent_goldwasser printenv
PATH=/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=bd7b5146d938
APP_COLOR=pink
LANG=C.UTF-8
GPG_KEY=0D96DF4D4110E5C43FBFB17F2D347EA6AA65421D
PYTHON_VERSION=3.6.6
PYTHON_PIP_VERSION=18.1
HOME=/root

~ →
```

Docker Environment Variables

Learn how to use environment variables in Docker containers. **Beginner** 45-60 min

2/4

Run a container named `blue-app` using image `kodekloud/simple-webapp` and set the environment variable `APP_COLOR` to `blue`. Make the application available on port `38282` on the host. The application listens on port `8080`.

Check

Next →

- ✓ Name: blue-app
- ✓ Image: kodekloud/simple-webapp
- ✓ ENV Variable: APP_COLOR=blue

```
~ → docker run -d --name blue-app -e APP_COLOR=blue -p 38282:8080 kodekloud/simple-webapp
ca59f70a3067e59620ca9bbc6a0056c973ab689719733db6f307e7f1da1716c8

~ →
```

CodeFix Originals x mysql - Official Image | Docker x +

studio.kodecloud.com/labs/docker/docker_env_vars

Back to Domain Docker Environment Variables Learn how to use environment variables in Docker containers. Beginner 45-60 min

Task Hint AI Assistant (Beta) 38:21

4 / 4 Next →

Deploy a `mysql` database using the `mysql` image and name it `mysql-db`.

Set the database password to use `db_pass123`. Lookup the `mysql` image on Docker Hub and identify the correct environment variable to use for setting the root password.

Check ✓

Next →

- ✓ Name: `mysql-db`
- ✓ Image: `mysql`
- ✓ Env: `MYSQL_ROOT_PASSWORD=db_pass123`

Terminal 1 HOST:38282 Stop Lab

```
~ → docker run -d --name mysql-db -e MYSQL_ROOT_PASSWORD=db_pass123 mysql
c010121517f8b933f20b84e03512802d0aa6aaf911c4617abc5be3b8e8029cfc
~ →
```

CodeFix Originals x mysql - Official Image | Docker x +

studio.kodecloud.com/labs/docker/docker_env_vars

Back to Domain Docker Environment Variables Learn how to use environment variables in Docker containers. Beginner 45-60 min

End 38:00

CONGRATULATIONS!!!!
You have successfully completed the challenge.

End lab session

Terminal 1 Stop Lab

```
~ → docker run -d --name mysql-db -e MYSQL_ROOT_PASSWORD=db_pass123 mysql
c010121517f8b933f20b84e03512802d0aa6aaf911c4617abc5be3b8e8029cfc
~ →
```