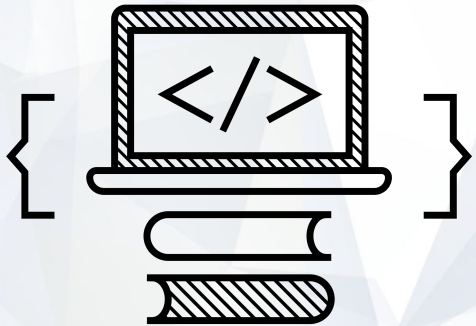
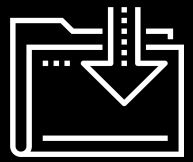


Docker Deployment

Java Accelerator 7

Lesson 6.1



© 2022 Trilogy Education Services, a 2U, Inc. brand. All Rights Reserved.

WELCOME

Learning Outcomes

By the end of this lesson, you will be able to:

Use Docker Compose to create a multicontainer application.

Deploy to Heroku using Docker.

docker

Docker Components

Client

- Docker build
- Docker pull
- Docker run

Docker_Host

Docker daemon

Containers

- Oracle
- NGINX
- NGINX
- Oracle

Images

- Oracle
- NGINX

Docker Registry

- Oracle
- NGINX
- Cassandra

What are some advantages to using containers?

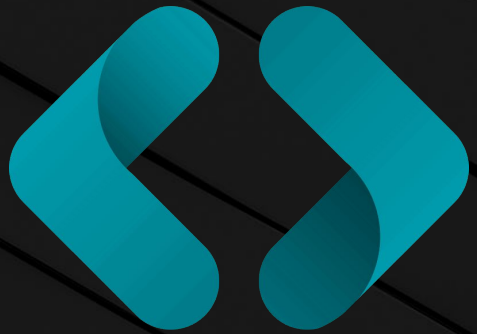
Docker Containers

Advantages of using containers:

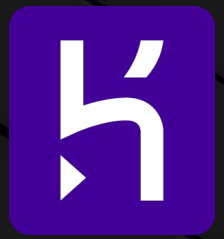
Isolation	The application is separate from other applications.
Easy deployment	The application and all of its dependencies are included in the container, so we can deploy the application by simply deploying its container.
Scalability	We can easily start or stop containers, as needed, to scale up or scale down.
Reliability	We can automatically stop or restart a container that is having problems.

What does Kubernetes provide for us?

Kubernetes provides **container orchestration**, meaning that it allows us to manage, deploy, and update containers.



Time to <code>



Docker Deployment to Heroku

Docker Compose

In the previous step, we containerized the Newsfeed application.

In this step, we'll containerize the database and the application using Docker Compose.

We've manually created and run the two containers for the Newsfeed application and database.

Now we'll use Docker Compose to automate that for us.

Docker Compose is a tool for defining and running multiple Docker containers.

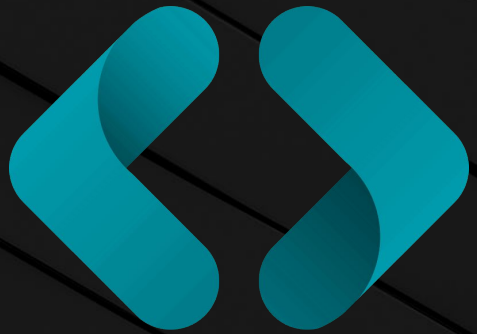
Docker Compose

`docker-compose.yml`

We'll use this `docker-compose.yml` file to configure the services of the application.

`docker-compose up`

Then, using the `docker compose up` command, we'll create and start all the services that the Compose file defines.



Time to <code>