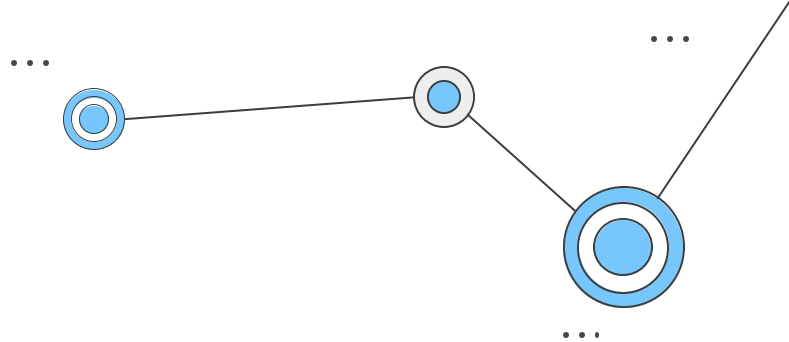


docker




Docker

Introduction to Docker by
Oleksandr Demchenko



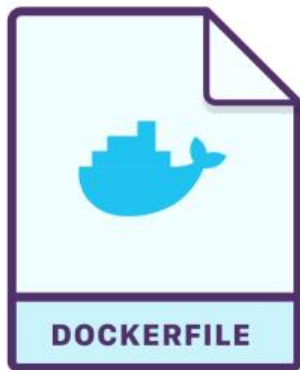
01

**What is
Docker?**



What is docker?

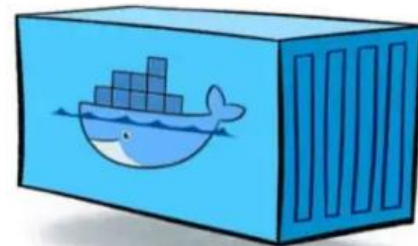
Docker - це платформа, яка дає змогу створювати, розгортати та запускати додатки в ізольованих контейнерах. Контейнери можна порівняти з віртуальними машинами, але вони легші, простіші та швидші у використанні.



Docker file



Docker Image

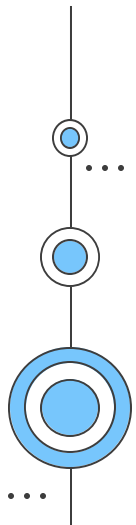


Docker Container

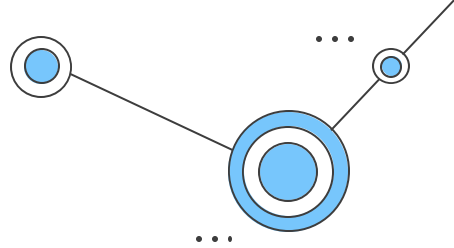


02

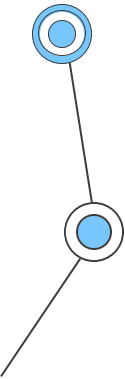
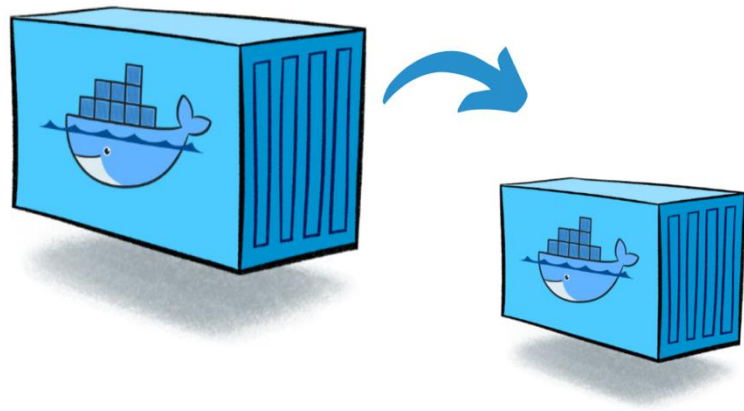
**What are
containers?**



What are containers?



Контейнер - це легкий та ізольований пакет, у якому міститься додаток разом з усіма його залежностями, файлами та налаштуваннями. Він дає змогу запускати додаток однаково на будь-якій системі, незалежно від того, на якій операційній системі або комп'ютері він працює.



03

Difference between containers and VMs



Difference between containers and VMs

Контейнери та віртуальні машини - це два різні підходи до віртуалізації, кожен з яких має свої сильні та слабкі сторони. Ось ключові відмінності, переваги та сценарії використання для обох підходів:

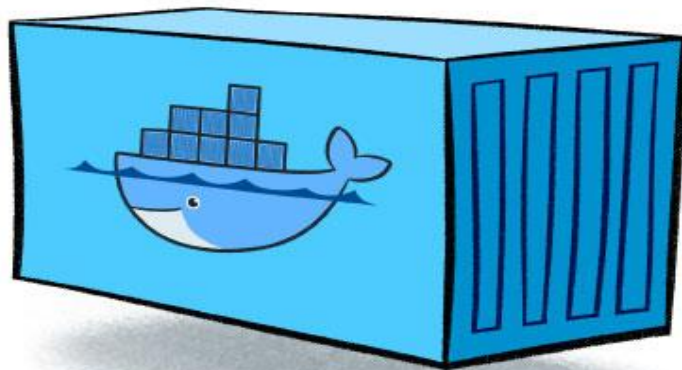
...



Операційна система:

Віртуальна машина: містить повну операційну систему (наприклад, Windows або Linux), так, наче це окремий комп'ютер. Кожен VM має своє власне ядро.

Контейнер: використовує ядро операційної системи хоста (основної системи, на якій запущено Docker), тому контейнеру не потрібна своя окрема операційна система.

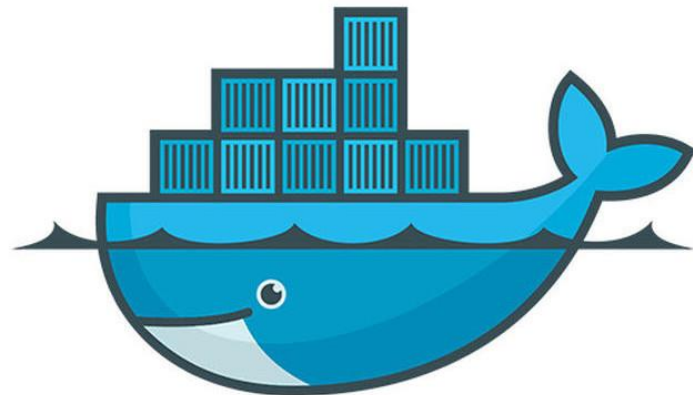




Ресурси:

Віртуальна машина: потребує більше ресурсів, тому що кожна VM запускає свою власну копію операційної системи та використовує значну кількість оперативної пам'яті та процесорних потужностей.

Контейнер: використовує менше ресурсів, оскільки немає необхідності запускати окрему операційну систему. Контейнери запускаються швидше і займають менше місця.

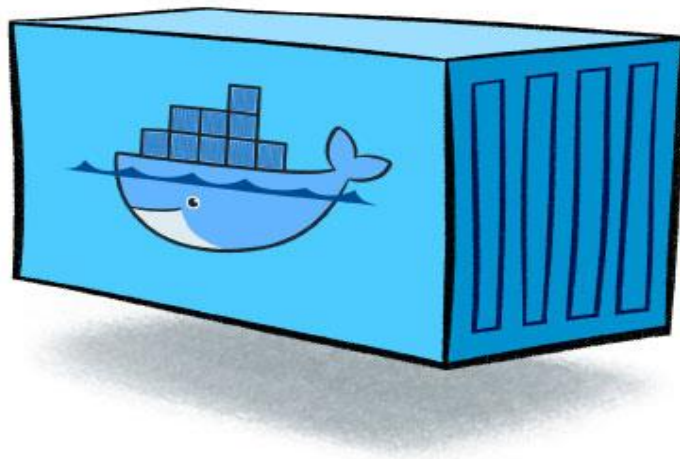




Ізоляція:

Віртуальна машина: повністю ізольована від основної системи та інших віртуальних машин, у неї є свої ресурси (процесор, пам'ять тощо).

Контейнер: також ізольований, але використовує спільне ядро з основною системою. Ця ізоляція легша і менш ресурсовитратна.

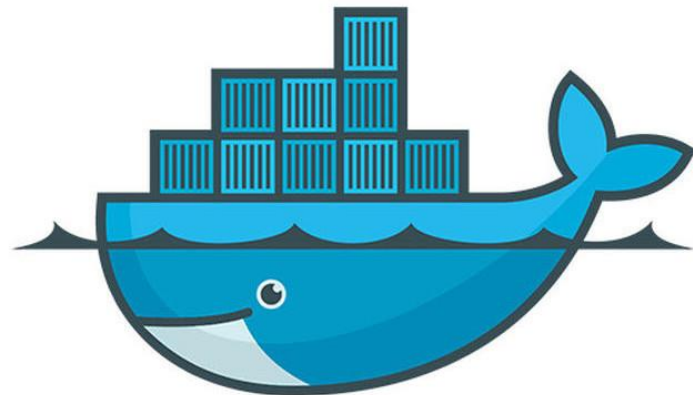




Час запуску:

Віртуальна машина: може запускатися кілька хвилин, тому що їй потрібно завантажити операційну систему.

Контейнер: запускається за секунди, тому що використовує вже завантажену операційну систему.





04

Practical part



Installation

Установка:

```
curl -fsSL https://get.docker.com -o get-docker.sh  
sudo sh get-docker.sh
```

Запуск служби Docker:

```
sudo service docker start
```

Перевірка статусу служби Docker:

```
sudo service docker status
```

Додати поточного користувача до групи Docker:

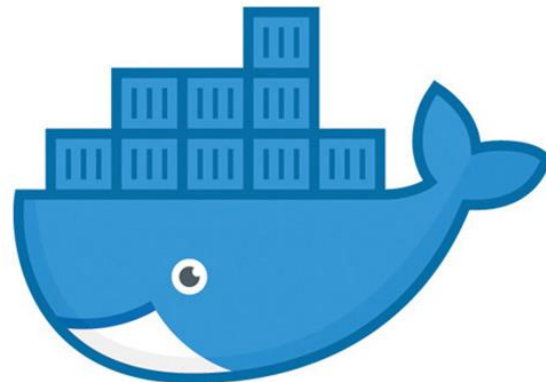
```
sudo usermod -a -G docker $USER
```

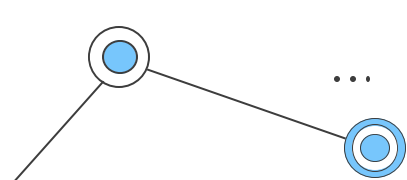
Перегружити термінал:

```
newgrp docker
```

Перевірка версії:

```
docker --version
```





Apache HTTP Server Project образ

`docker pull httpd`

Перевірка скачаного образу

`docker images`

Запуск образа

`docker run --name <name> -d -p 8080:80 httpd`

Docker перевірка контейнера

`docker ps`

Docker стоп

`docker stop <container_id_or_name>`

Docker старт

`docker start <container_id_or_name>`

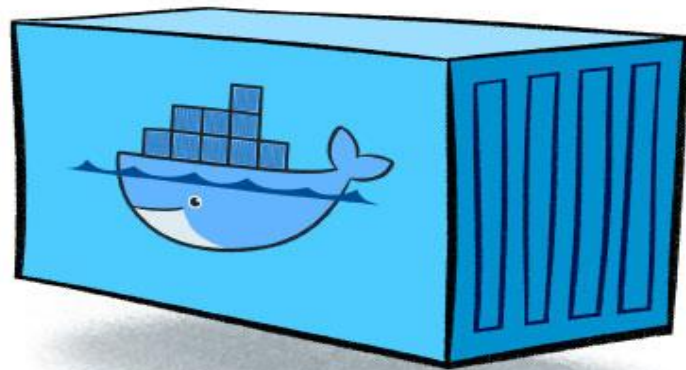
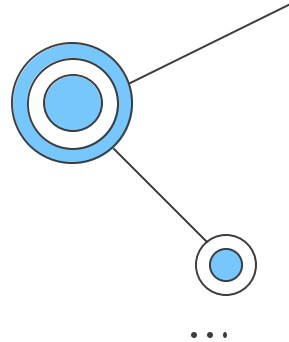
Видалити контейнер

`docker rm <container_id_or_name>`

Видалити образ

`docker rmi <image_id_or_name>`

Run



Resources

01

Docker

<https://docs.docker.com/>

02

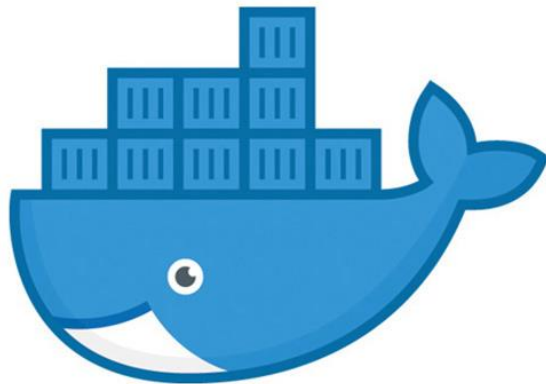
Docker Engine overview

<https://docs.docker.com/engine/install/>

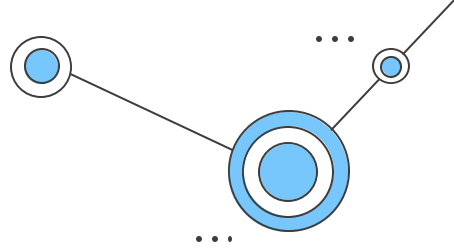
03

Docker hub

<https://hub.docker.com/>



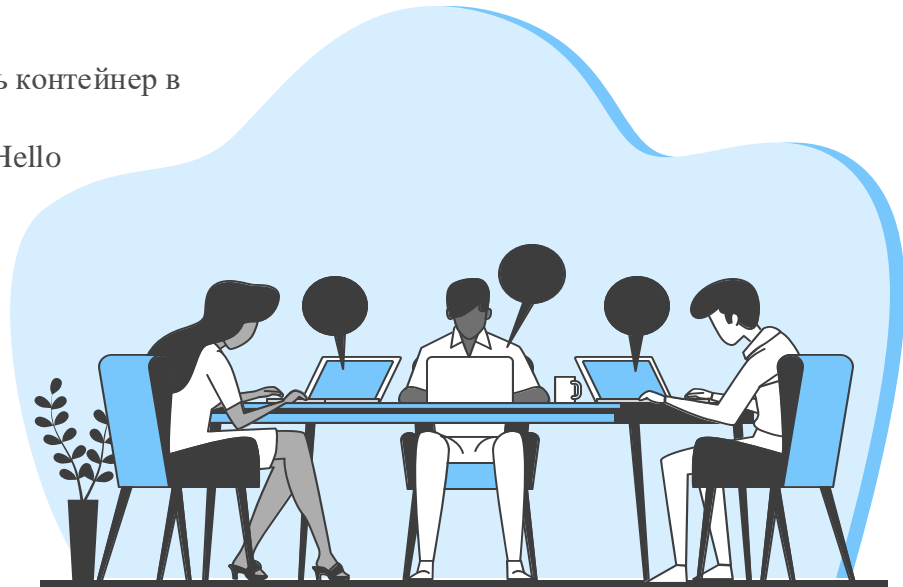
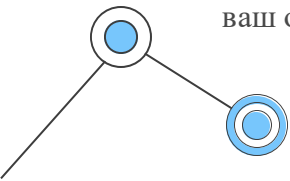
Homework



1. Встановіть Docker на своєму комп'ютері
2. Створіть HTML-файл з текстом «hello DevOps course» у тілі
3. Підготуйте Docker-файл, який буде містити наступні кроки:
 - образ докера Nginx
 - скопіюйте HTML-файл в директорію веб-контенту Nginx
 - відкрити 80 портів для доступу до Nginx
4. Створіть власний контейнер за допомогою Docker-файл
5. Створіть обліковий запис на Docker Hub
6. Перенесіть ваш контейнер на Docker Hub
7. Створіть bash-скрипт, який скачає із Docker Hub та запустить контейнер в фоновому режимі на 80 порту
8. В результаті ми зможемо побачити веб-сторінку з текстом "Hello DevOps course" яка завантажується на 80 порті
9. Збережіть файли до вашого репозиторію GitHub

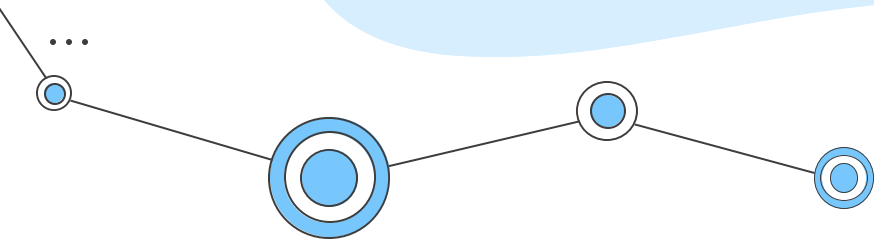
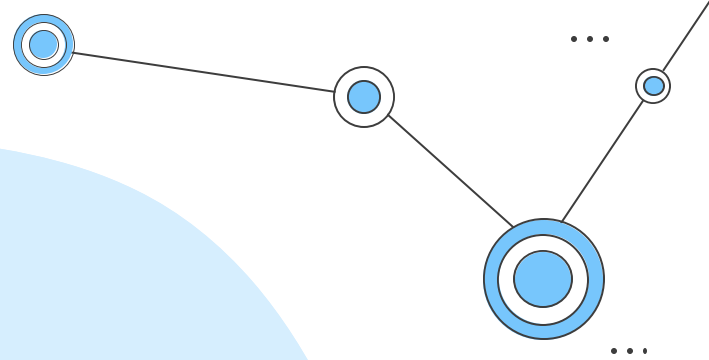
Що мені потрібно від Вас для перевірки:

1. Посилання на GitHub де буде знаходитись Docker-файл, HTML-файл і bash-скрипт
2. Посилання на Docker Hub де буде завантажений ваш образ





Questions
?



Thanks!

