

# Docker

**Docker** - це відкрите програмне забезпечення, яке надає платформу для розгортання, управління та ізоляції програмних додатків в контейнерах. Контейнери Docker дозволяють упаковувати програмні додатки та їх залежності разом у стандартизовані виконувані середовища, що називаються контейнерами.

Docker надає зручний та легко використовуваний спосіб впакування, розгортання та управління додатками, що дозволяє забезпечити переносимість, ефективність та ізоляцію середовищ виконання. Він став популярним інструментом у сфері розробки програмного забезпечення та розгортання додатків у хмарних середовищах.

Основні поняття Docker включають:

1. Контейнери: Контейнери Docker є виконуваними одиницями програмного забезпечення, які містять програми, залежності та всі необхідні файли для їх роботи. Контейнери запускаються на основі образів Docker.
2. Образи: Образ Docker - це статичний пакет, який включає всі необхідні файли та залежності для виконання певного додатка. Він визначає початковий стан контейнера та параметри його роботи.
3. Dockerfile: Dockerfile - це текстовий файл, який містить інструкції для автоматичної побудови образу Docker. Він описує потрібні кроки для встановлення середовища та налаштування додатка в контейнері.
4. Реєстр Docker: Реєстр Docker є централізованим сховищем образів Docker. Ви можете використовувати публічний реєстр Docker Hub або створити свій власний приватний реєстр для зберігання та розповсюдження образів Docker.
5. Docker Compose: Docker Compose - це інструмент для визначення та управління багатоконтейнерними додатками. Він дозволяє описувати

структуру додатка, включаючи кілька контейнерів, мереж та інших ресурсів, у файлі docker-compose.yml.

6. Docker Swarm: Docker Swarm - це інтегрована в Docker вбудована система оркестрації контейнерів. Вона дозволяє керувати кластером контейнерів Docker, розподіляти завдання між вузлами та забезпечувати високу доступність та масштабованість додатків.

### Налаштування Docker на Ubuntu:

1. Встановити Docker Engine:

```
carrie@carrie-VirtualBox:~$ sudo apt-get install docker.io
```

2. Додати користувача до групи docker, щоб мати можливість запускати Docker-команди без потреби використання sudo:

```
carrie@carrie-VirtualBox:~$ sudo usermod -aG docker carrie
```

3. Оновити систему або виконати команду для оновлення групових налаштувань:

```
carrie@carrie-VirtualBox:~$ newgrp docker
```

4. Переконайтеся, що служба Docker запущена:

```
carrie@carrie-VirtualBox:~$ sudo service docker start
```

5. Перевірити, чи Docker встановлений і працює, запустивши наступну команду:

```
carrie@carrie-VirtualBox:~$ docker run hello-world

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

```

carrie@carrie-VirtualBox:~$ docker run hello-world

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/

```

## 6. Додаткова перевірка роботи Docker:

```

carrie@carrie-VirtualBox:~$ sudo service docker status
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor prese
   Active: active (running) since Thu 2023-06-08 13:31:46 EEST; 6min ago
   TriggeredBy: ● docker.socket
     Docs: https://docs.docker.com
    Main PID: 2503 (dockerd)
      Tasks: 9
     Memory: 29.3M
        CPU: 1.788s
    CGroup: /system.slice/docker.service
            └─2503 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/con>

чеп 08 13:31:44 carrie-VirtualBox dockerd[2503]: time="2023-06-08T13:31:44.678>
чеп 08 13:31:45 carrie-VirtualBox dockerd[2503]: time="2023-06-08T13:31:45.291>
чеп 08 13:31:45 carrie-VirtualBox dockerd[2503]: time="2023-06-08T13:31:45.736>
чеп 08 13:31:46 carrie-VirtualBox dockerd[2503]: time="2023-06-08T13:31:46.299>
чеп 08 13:31:46 carrie-VirtualBox dockerd[2503]: time="2023-06-08T13:31:46.444>
чеп 08 13:31:46 carrie-VirtualBox dockerd[2503]: time="2023-06-08T13:31:46.444>
чеп 08 13:31:46 carrie-VirtualBox systemd[1]: Started Docker Application Conta>
чеп 08 13:31:46 carrie-VirtualBox dockerd[2503]: time="2023-06-08T13:31:46.595>
чеп 08 13:37:21 carrie-VirtualBox dockerd[2503]: time="2023-06-08T13:37:21.767>
чеп 08 13:37:36 carrie-VirtualBox dockerd[2503]: time="2023-06-08T13:37:36.228>
lines 1-22/22 (END)

```