

Docker

INSTALL

```
sudo yum install -y yum-utils
sudo yum-config-manager --add-repo
https://download.docker.com/linux/centos/docker-ce.repo
```

```
sudo yum install docker-ce
sudo systemctl enable docker
sudo systemctl start docker
```

Add your current user to the Docker group

```
sudo usermod -a -G docker cloud_user
```

Create your first container using the hello-world image

```
docker run hello-world
```

docker-composer install:

```
curl -L
"https://github.com/docker/compose/releases/download/1.26.0/do
cker-compose-$(uname -s)-$(uname -m)" -o
/usr/local/bin/docker-compose
mv /usr/local/bin/docker-compose /usr/bin/docker-compose
chmod +x /usr/bin/docker-compose
```

Базовые команды

Версия

```
docker -v
```

Запущенные контейнеры

```
docker ps
```

Все контейнеры

```
docker ps -a
```

Образы

`docker images`

Поиск образа

`docker search`

Скачать образ

`docker pull tomcat`

Запустить интерактивно на порту

`docker run -it -p 7777:8080 tomcat`

Запустить неинтерактивно демоном

`docker run -d -p 8888:8080 tomcat`

`docker run -d -p 80:80 nginx`

Переименовать образ

`docker tag oldname:v1 newname:v2`

Сделать копию

`docker build -t myimage:latest .`

И запустить

`docker run -d myimage:latest`

START DOCKER

`docker start {b298bda36d02,124c0e4946d9,704993e52d21}`

`docker start $(docker ps -a -f status=exited -q)`

STOP DOCKER

`docker stop $(docker ps -q)`

Удалить контейнер

`docker rm 1234567890`

Удалить образ

```
docker rmi 1112227778
```

Delete all Containers

```
docker rm -f $(docker ps -aq)
```

Удалить все неиспользуемые контейнеры

```
docker container prune
```

Delete all Images

```
docker rmi -f $(docker images -q)
```

Попасть внутрь контейнера

```
docker exec -it 5267e21d140 /bin/bash  
echo "New Version v2" >> /var/www/html/index.html  
exit
```

Создать новый образ

```
docker commit 5267e21d140 newimage_v2:latest
```

REMOVE DOCKER

```
systemctl stop docker  
systemctl stop docker.socket
```

```
yum remove -y docker-ce  
rm -rd /var/lib/docker
```

Пример докер файла

Логи сервисов

```
docker logs name_service
```

Посмотреть в реальном времени:

```
docker logs -f name_service
```

Передать в файл вместе с ошибками:

```
docker logs name_service > /home/dataverse.log 2>&1
```

Хранилище образов

`/var/lib/docker`