

# Контейнеры

## Основные команды

Команда	Доп. пар.	Описание
<b>docker ps</b>		список работающих контейнеров
	-a	список остановленных но еще существующих контейнеров
<b>docker logs &lt;name&gt;</b>		список событий внутри контейнера name
<b>docker start &lt;name&gt;</b>		запуск остановленного контейнера
<b>docker restart &lt;name&gt;</b>		Перезапускает один или несколько контейнеров. Можно считать приблизительным аналогом выполнения для заданных контейнеров команды docker stop, за которой сразу следует команда docker start.
	-t	определяет интервал времени ожидания, необходимого для завершения работы контейнера, перед его остановом по сигналу SIGTERM.
<b>docker stop &lt;name&gt;</b>		Останавливает (но не удаляет) один или несколько контейнеров. После выполнения этой команды заданный контейнер переходит в состояние «остановлен».
	-t	Аналогично restart
<b>docker kill &lt;name&gt;</b>		Сигнал основному процессу (PID=1) в контейнере. По умолчанию SIGKILL (немедленное завершение работы). Возвращает идентификатор контейнера.
	-s	другой сигнал
<b>docker rm &lt;name или id&gt;</b>		удаление остановленного контейнера
	-f	позволяет удалять работающие контейнеры.

Команда	Доп. пар.	Описание
	-v	<p>удалить тома, созданные удаляемым контейнером (если эти тома не смонтированы на каталоги и не используются другими контейнерами)</p> <pre>docker rm -v \$(docker ps -aq -f status=exited)</pre> <p>удаляет все остановленные контейнеры</p>
<b>docker top &lt;name или id&gt;</b>		информация о процессах внутри контейнера
<b>docker port &lt;name или id&gt;</b>		номера портов, назначенные механизмом Docker
<b>docker create &lt;name или id&gt;</b>		Создает контейнер из образа, но не запускает его. Аргументы как у docker run.
<b>docker run</b>		создание и запуск контейнера из образа
	-it или -d	сеанс интерактивной работы или запуск в фоновом режиме
	--link <namecur:nameinnew>	<p>соединение между новым контейнером и существующим контейнером myredis, в новом контейнере ссылка на существующий обозначена именем redis</p> <pre>--link myredis:redis</pre>
	--name	имя для дальнейшего взаимодействия
	--hostname	имя для обращения
	--mount	монтирование
	--volumes-from CONTAINER	использование томов контейнера
	-p HostPort:ContPort	перенаправление с портов хоста на порт контейнера
	--rm	удаление остановленного контейнера и файловой системы после запуска и выполнения команды. Несовместим с ключом -d.

Команда	Доп. пар.	Описание
	--restart	<p>Позволяет настроить образ действий при попытке Docker перезапустить остановленный контейнер.</p> <ul style="list-style-type: none"> <li>• no запрещает любые попытки перезапуска контейнера.</li> <li>• unless-stopped Будет перезагружаться до остановки. После перезагрузки демона Docker контейнер останется выключенным.</li> <li>• always попытки перезапуска выполняются в любом случае вне зависимости от состояния контейнера после выхода. После перезагрузки демона Docker контейнер опять запустится.</li> <li>• on-failure: перезапуск выполняются для контейнера, завершившего работу с ненулевым статусом. После перезагрузки демона Docker контейнер опять запустится. Может быть задан аргумент, определяющий максимальное количество попыток перезапуска (иначе попытки будут выполняться бесконечно).</li> </ul> <div> docker run --restart on-failure:10 postgres </div>
	-t, --tty	Создает псевдоустройство ТТУ (терминал). Как правило, используется вместе с ключом -i для запуска контейнера в интерактивном режиме.
	-e, --env	Определяет переменные среды внутри контейнера.

Команда	Доп. пар.	Описание
	--entrypoint	Определяет точку входа для запускаемого контейнера в соответствии с заданным аргументом, заменяя содержимое любой инструкции ENTRYPOINT из Dockerfile.
	-u, --user	Определяет пользователя, от имени которого выполняются команды. Может быть задано как символьное имя пользователя или как числовой идентификатор UID. Заменяет содержимое инструкции USER из Dockerfile.
	-w, --workdir	Устанавливает рабочий каталог в контейнере в соответствии с заданным путевым именем. Заменяет любые значения, определенные в файле Dockerfile.
	--add-host=docker:10.180.0.1	X3, узнать
<b>docker pause/unpause</b>		Временно приостанавливает/запускает все процессы внутри контейнера. Процессы не получают никаких сигналов приостановки(не могут быть остановлены и завершены или удалены).
<b>docker commit &lt;container name&gt; &lt;repo name&gt;</b>		образ контейнера. По умолчанию приостанавливаются, без приостановки --pause=false.
<b>docker network ls</b>		список сетей

## Взаимодействие реального времени

Ctrl-PQ (или Ctrl+P, Ctrl+Q) отключение от контейнера без остановки

Команда	Доп. пар.	Описание
<b>docker attach &lt;name или id&gt;</b>		наблюдать или взаимодействовать с основным процессом внутри контейнера.
<b>docker cp</b>		<p>Позволяет копировать файлы между файловыми системами контейнера и хоста.</p> <pre>docker cp /tmp/config.ini grafana:/usr/share/grafana/conf/</pre>

Команда	Доп. пар.	Описание
<b>docker exec</b>		<p>Запускает заданную команду внутри контейнера (может быть работающий сейчас). Может использоваться для выполнения задач сопровождения или в качестве замены ssh при входе (регистрации) в контейнер.</p> <div><code>docker exec -it ubuntu:latest bash</code></div>
<b>docker events</b>		<p>Выводит в реальном времени события от демона демону. Выхода Ctrl-C.</p>

Примеры

- Запуск контейнера образа ubuntu:latest в интерактивном режиме с bash

`docker run -it ubuntu:latest /bin/bash`

- Подключение к работающему контейнеру и запуск в нем bash

`docker exec -it e37f24dc7e0a bash`