



Если контейнер имеет статус:

```
Exited
```

это означает, что приложение внутри контейнера завершилось с ошибкой.

---

???????????? ???? ??????????????

```
docker compose logs <service>
```

Пример:

```
docker compose logs php
```

Часто в логах можно увидеть причину проблемы:

- ошибка конфигурации
  - отсутствующий файл
  - ошибка подключения к базе данных
  - неправильные переменные окружения
- 

???????????? ??????????????

????????????????????

Иногда контейнер запускается, но сразу падает и перезапускается.

Проверить статус:

```
docker ps
```

Пример:

```
Restarting (1) 10 seconds ago
```

---

???????????? ???? ? ??????????  
?????????

```
docker compose logs -f <service>
```

Это позволит увидеть, на каком этапе происходит ошибка.

???? ???? ??????????????????

Ошибка запуска контейнера может выглядеть так:

```
Bind for 0.0.0.0:80 failed: port is already allocated
```

Это означает, что порт уже используется другим процессом.

???????????, ??? ?????????????????

```
sudo lsof -i :80
```

или

```
netstat -tulpn | grep 80
```

???????????? ??????????

- остановить другой сервис
- изменить порт в `docker-compose.yml`

Пример изменения порта:

```
ports:  
  - "8080:80"
```



????????? ???? ?

- неправильный путь к папке
- проблемы с правами доступа
- файлы редактируются не в той директории

????????? ? ???? ?

?????

Если сервисы не могут взаимодействовать:

например:

```
php -> mysql
```

????????? ?

```
docker network ls
```

????????? ??????????

?????????

```
docker network inspect <network>
```

????????? ?????????? ?

?????????

```
docker compose exec php ping mysql
```

Если соединение не работает — возможно сервис не запущен или контейнеры находятся в разных сетях.

---

?????????? ???????

Иногда контейнер перестаёт реагировать на команды.

Попробовать перезапустить:

```
docker restart <container>
```

Если это не помогает:

```
docker kill <container>
```

---

Docker ?????????? ??????????  
?????? ???????

Со временем Docker может занимать большое количество дискового пространства.

Проверить использование:

```
docker system df
```

---

????????? ?????????????????????? ??????????

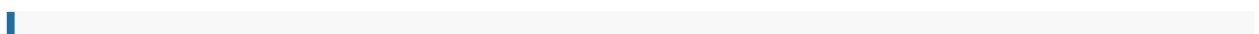
```
docker system prune -f
```

---

????????? volumes

```
docker volume prune -f
```

---



**Важно:** Перед удалением volumes убедитесь, что они не содержат важных данных.

?????????? ?? ?????????? ??

# HTTP

Если сайт или API не открывается:

?????????? ??????????

```
docker compose ps
```

?????????? ?????????? ??????????

```
docker ps
```

Пример:

```
0.0.0.0:8080->80/tcp
```

?????????? ?????????? ??????????

????????????

```
docker compose exec nginx curl localhost
```

Если внутри контейнера всё работает — проблема может быть в сетевых настройках.

????????? ????????????????

# Docker Compose

Перед запуском полезно проверить итоговую конфигурацию:

```
docker compose config
```

Команда показывает:

- итоговый YAML
- подставленные переменные окружения
- конфигурацию сервисов

Это помогает обнаружить:

- ошибки YAML
- пропущенные переменные
- неправильные настройки сервисов

---

????????? ?????????????????? ( ??????????)

Если сервис не работает:

1 ?????????????? ??????????????

```
docker compose ps
```

2 ?????????????? ??????

```
docker compose logs --tail=100
```

3 ?????????????? ??????????

```
docker stats
```

4 ?????????? ?????

```
docker network inspect <network>
```

5 ?????????? ????????????????

```
docker compose config
```

????????? ?????????? ????????????

????????? Docker-????????? ?? ??

????????????????? ???????????

Если команда `docker compose` выполняется не там, где лежит `docker-compose.yml`, Docker не сможет найти конфигурацию.

????????????????????? ?????????? ???????????????

В Docker Compose используется **имя сервиса**, а не имя контейнера.

Пример правильного имени:

```
mysql
```

????????????????? `.env` ???????

Если в `docker-compose.yml` используются переменные окружения, но файл `.env` отсутствует, сервис может не запуститься.

