# Запуск из контейнера

<https://container-registry.oracle.com/ords/f?p=113:4:102282896779006:::4:P4_REPOSITORY,AI_REPOSITORY,AI_REPOSITORY_NAME,P4_REPOSITORY_NAME,P4_EULA_ID,P4_BUSINESS_AREA_ID:1863,1863,Oracle%20Database%20Free,Oracle%20Database%20Free,1,0&cs=3z5k7acuakxlYxZcXs0UTTx9n7801qITRH8iOLiQcdt4VAuthwIQQxYmUZluCht8-XVKox9aLy8wZ2dqUWM2oIA>

Загрузить образ

docker pull container-registry.oracle.com/database/free:latest-lite

запуск контейнера

docker run \

-p 1521:1521 \

-e ORACLE\_PDB=<Oracle> \

-e ORACLE\_PWD=<MyPass12345> \

<container-name>

# Подключение к серверу БД

## Добавление юзера

## Проверка юзеров и список БД

Вход в контейнер

docker exec -it <container\_name\_or\_id> bash

Вход в БД как суперпользователь

sqlplus sys/<пароль>@//localhost:1521/<имя БД> as sysdba

Далее в SQL:

SELECT username FROM dba\_users ORDER BY username;

Выход из sql

Exit

Выход из exec подключения

Ctrl + D

# Подключение к БД

## Подключение из пайтон

С помощью библиотеки oracledb.

По умолчанию oracledb работает в thin моде, что значит, что эта библиотека работает самостоятельно и не нужно ничего для её работы, чтобы делать запросы в oracle DB.

import oracledb

connection = oracledb.connect(

    user="system",  # или другой пользователь (например, "sys as sysdba")

    password="MyPass12345",

    dsn = "localhost:1521/Oracle"  # или "localhost:1521/ORACLE"

)

print("Успешное подключение!")

cursor = connection.cursor()

cursor.execute("SELECT username FROM dba\_users ORDER BY username")

for row in cursor:

    print(row[0])

cursor.close()

connection.close()

Если добавить вот это в начале, то можно запустить **oracledb в thick моде**, что означает, что он использует oracle instant client.

oracledb.init\_oracle\_client(lib\_dir=r"<путь к папке instant>")