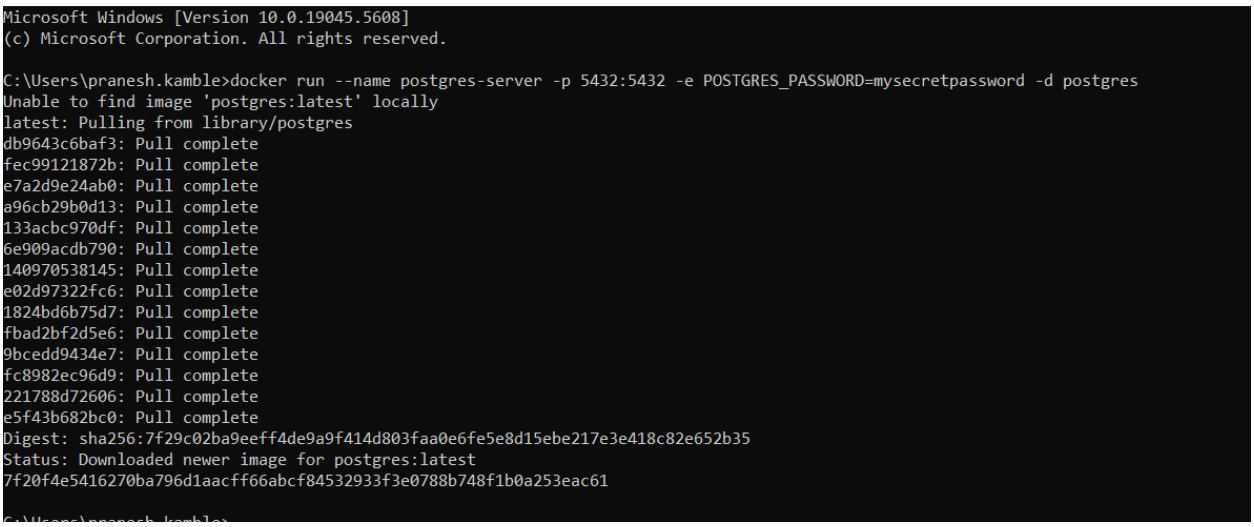
Documentation

TASK 1: To connect PostgreSQL using docker container.

1. Pull docker Postgres image using docker run.
   > docker run --name postgres-server -p 5432:5432 -e POSTGRES_PASSWORD=mysecretpassword -d postgres



   Here postgres-server is name of the container, 5432 is default port number of PostgreSQL , -e is enviornment variable for POSTGRES_PASSWORD, -d is detach mode this will run postgres in background & we can use command on terminal.

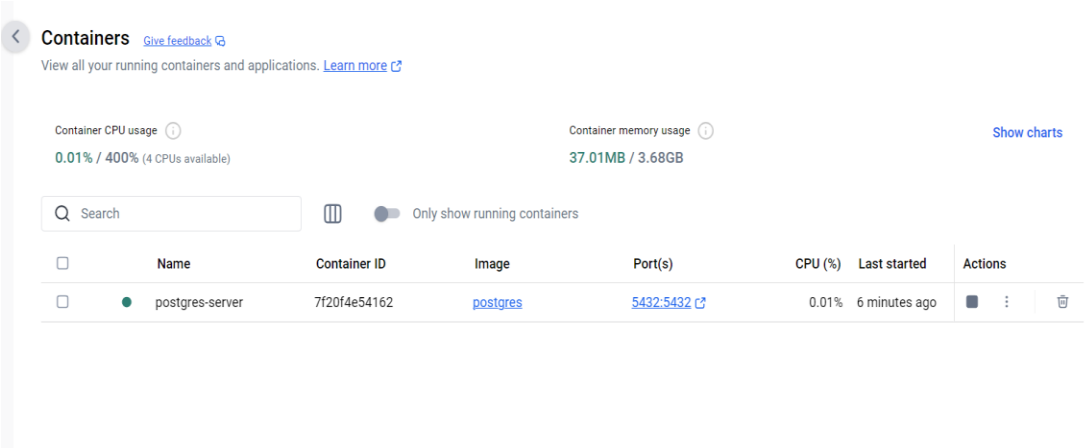
   This will create a docker container with name postgres-server and you can run this container.

2. Check if the container is running using both terminal & docker desktop.

   > docker ps

```
C:\Users\pranesh.kamble>docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED         STATUS         PORTS                    NAMES
7f20f4e54162   postgres   "docker-entrypoint.s…"   4 minutes ago   Up 4 minutes   0.0.0.0:5432->5432/tcp   postgres-server
```

   Also in docker desktop:



3. Connect to docker psql terminal:

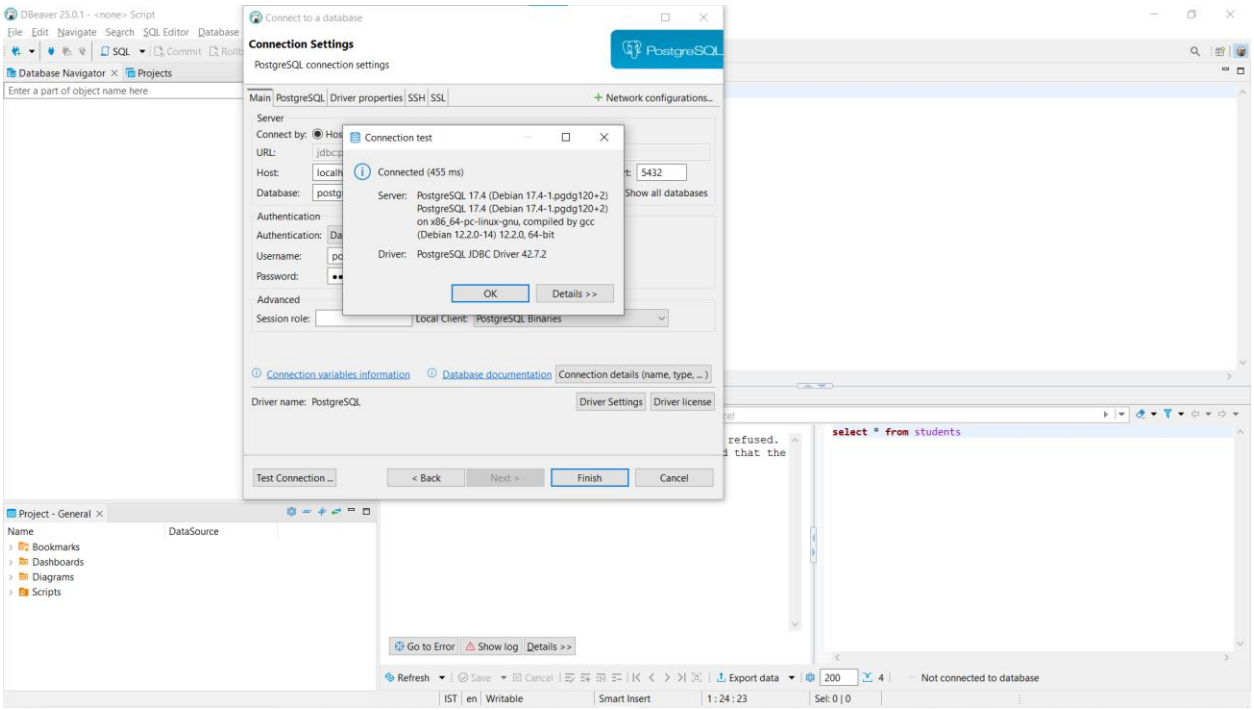
> docker exec –it postgres-server psql –U postgres

```
C:\Users\pranesh.kamble>docker exec -it postgres-server psql -U postgres
psql (17.4 (Debian 17.4-1.pgdg120+2))
Type "help" for help.

postgres=#
```

Here docker exec command runs a new command in a running container, -i is for interactive, t is for terminal session, container name, psql –U postgres for postgres for interactive terminal to run queries for user as postgres. -U is for username instead of default.

4. Connect to Dbeaver:



5. Create table using psql:

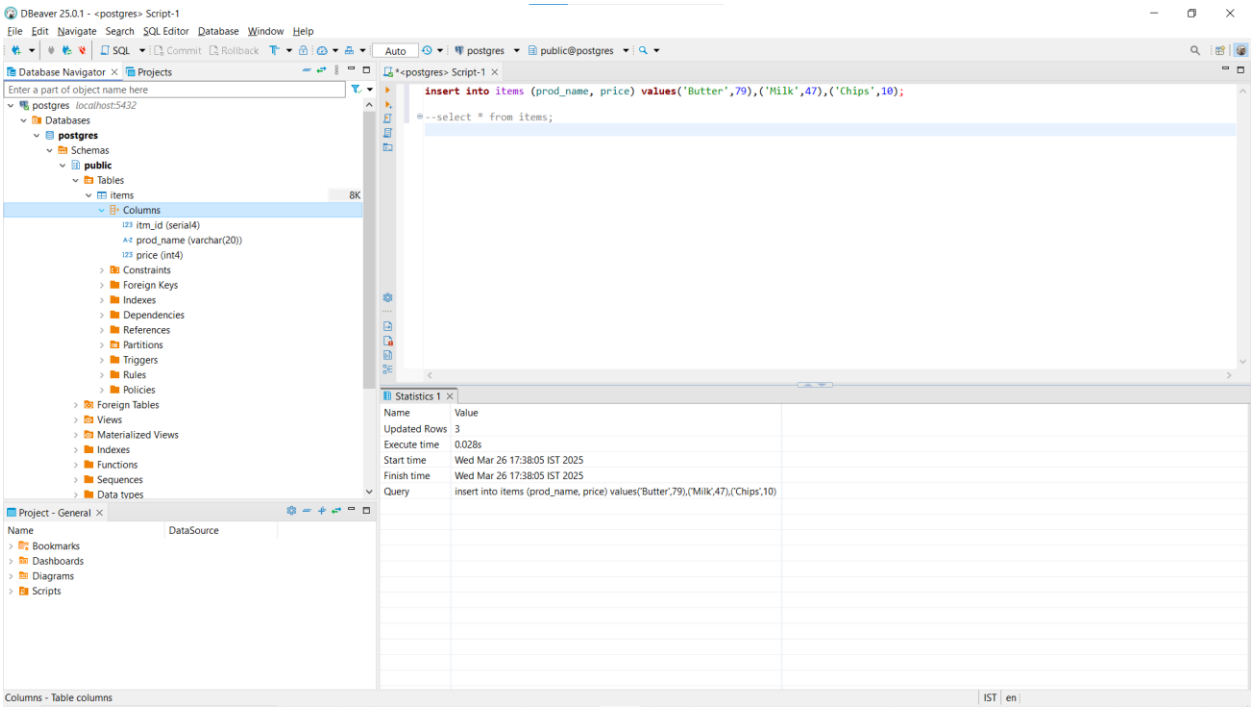
> CREATE TABLE items (
itm_id SERIAL PRIMARY KEY,
prod_name varchar(20) not null,
price int not null
);

This will create a table items with columns as itm_id, prod_name, price.

```
postgres=#
postgres=# CREATE TABLE items (
postgres(#     itm_id SERIAL PRIMARY KEY,
postgres(#     prod_name varchar(20) not null,
postgres(#     price int not null
postgres(# );
CREATE TABLE
```

6. Insert into table using terminal & dbeaver:
   > insert into items(prod_name, price)
   values ('Toor Dal',50),
   ('Basmati Rice',101),
   ('Soap',500),
   ('Shampoo',366);
   This will insert element/rows in table items.

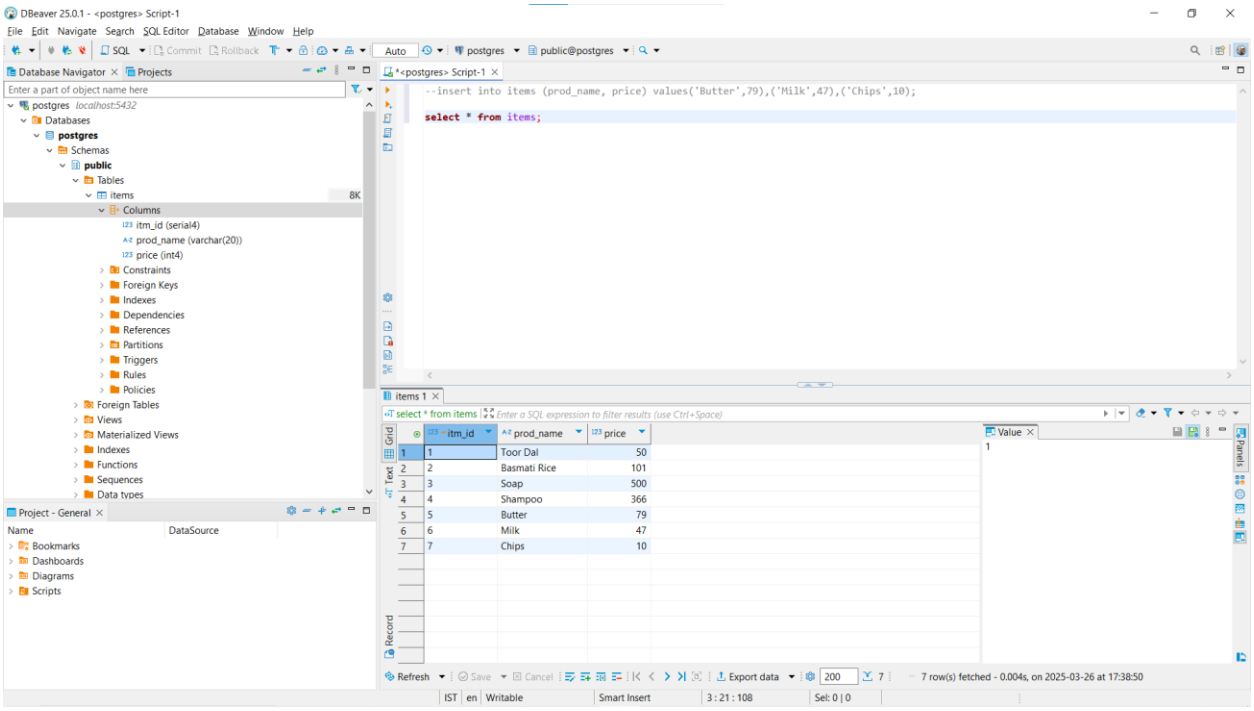
```
postgres=# insert into items(prod_name, price)
postgres-# values ('Toor Dal',50),('Basmati Rice',101),('Soap',500),('Shampoo',366);
INSERT 0 4
```



7. Return what inside the table items.

> select * from items;

```
postgres=# select * from items;
 itm_id |  prod_name   | price
--------+--------------+-------
      1 | Toor Dal     |    50
      2 | Basmati Rice |   101
      3 | Soap         |   500
      4 | Shampoo      |   366
(4 rows)
```




```
postgres=# select * from items;
 itm_id |  prod_name   | price
--------+--------------+-------
      1 | Toor Dal     |    50
      2 | Basmati Rice |   101
      3 | Soap         |   500
      4 | Shampoo      |   366
      5 | Butter       |    79
      6 | Milk         |    47
      7 | Chips        |    10
(7 rows)
```