PostgreSQL :

- postgreSQL is a powerful open-source relational database management system that provides robust data storage, management, and retrieval capabilities. It is known for its scalability, extensibility, and reliability.

PgAdmin :

- pgAdmin is a popular open-source administration and management tool for PostgreSQL databases.
- It provides a graphical user interface (GUI) that allows database administrators and developers to manage and manipulate their PostgreSQL databases easily.
- including server management, user and role management, database management, query building and execution.

Run PostgreSQL and pgAdmin in docker for local development using docker compose :

create a directory

```
mkdir sql-pgadmin
```

create a file and name it as

```
docker-compose.yml
```

docker compose

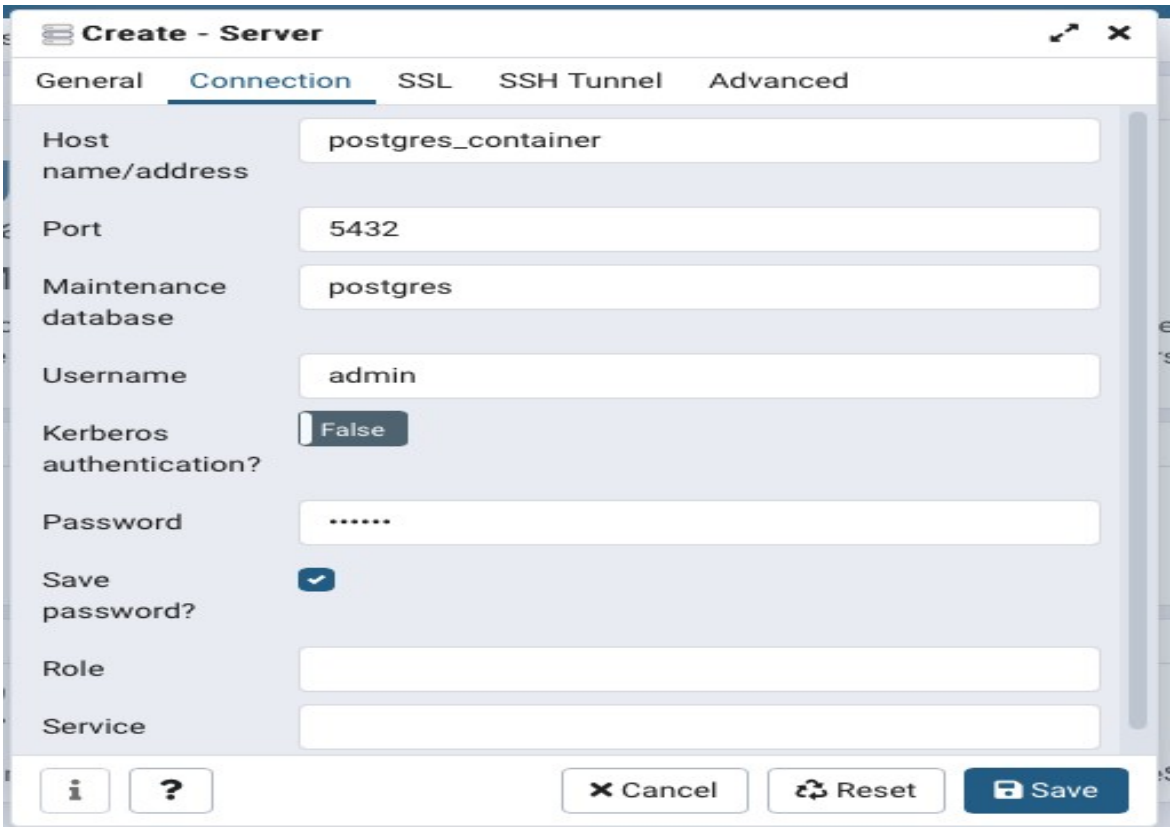
Docker Compose: Compose allows you to define and run multi-container Docker applications

PgAdmin Docker Setup: Set up the PostgreSQL Connection :

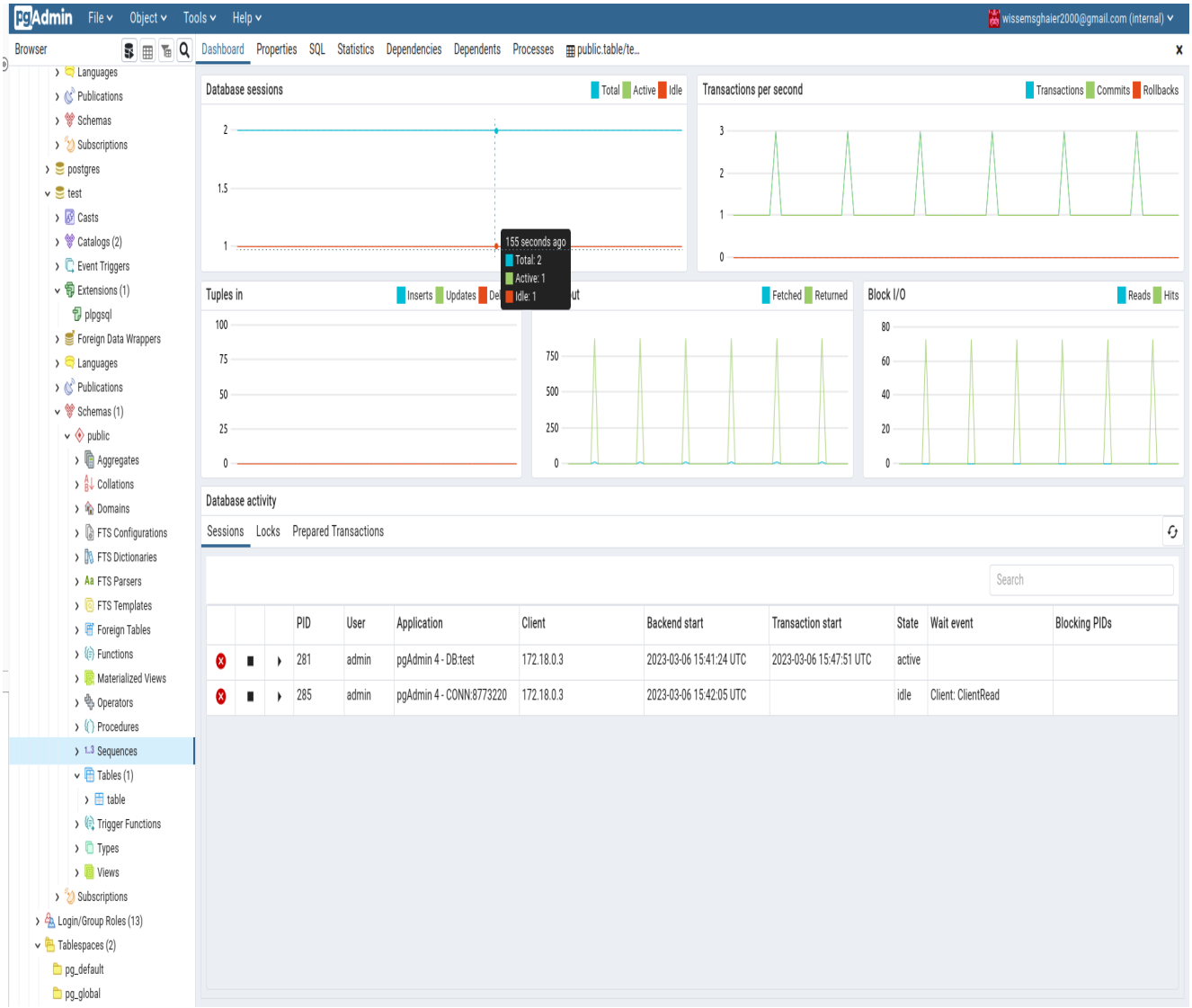
Next, we need to add PGAdmin access to our database to manage and use it for the PgAdmin Docker setup. Select “Add New Server”, and you should see a screen like this one:

Click on the “Connection” tab, and in the field “Hostname/address”, type in the name that appears on the docker-compose file for this Postgres container:

Also, you will use the username and password that you specified in your docker-compose:



The connection to the database has been established. We can now create a database table, manage users and quickly adjust our PostgreSQL database.



```yaml
version: "3.8"
services:
  db:
    image: postgres
    container_name: local_pgdb
    restart: always
    ports:
      - "54320:5432"
    environment:
      POSTGRES_USER: admin
      POSTGRES_PASSWORD: admin
    volumes:
      - local_pgdata:/var/lib/postgresql/data
  pgadmin:
    image: dpage/pgadmin4
    container_name: pgadmin4_container
    restart: always
    ports:
      - "5050:80"
    environment:
      PGADMIN_DEFAULT_EMAIL: wissemsghaier2000@gmail.com
      PGADMIN_DEFAULT_PASSWORD: admin
    volumes:
      - pgadmin-data:/var/lib/pgadmin

volumes:
  local_pgdata:
  pgadmin-data:
```

- First line defines the version of the compose file which is 3.8.
- Next we have services section. Inside this, we have to define 2 services **postgreSQL**and **pgAdmin**
- container_name is used to define container names for postgreSQL & pgAdmin
- restart always will restart the container when either the Docker daemon restarts
- ports is used to define both host and container ports
- environment defines : for both services we have set the user id and password
- volumes tag is used to mount a folder from the host machine to the container :create volume name local_pgdata and mount this volume to container’s path.
- When you stop or down the docker container, the database and connection details will still be there.