**Ass-8**

**Connect from your local machine to a postgresql database in docker**

**Requirement:-**

1. **Install docker**

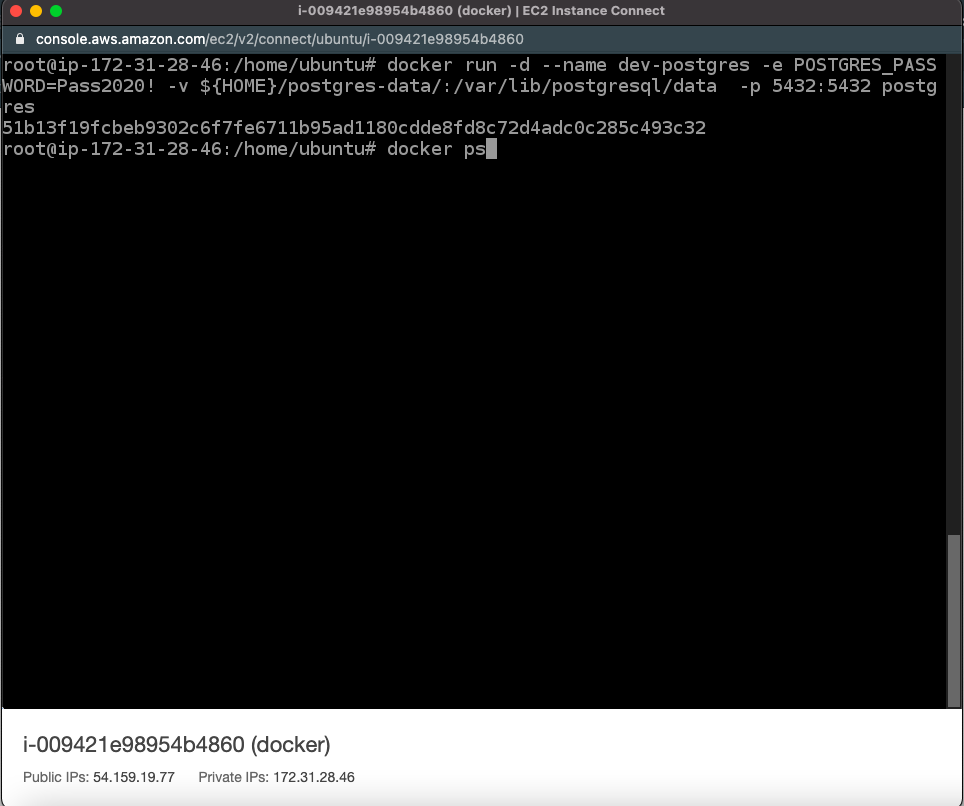
**Cmd:- sudo apt-get install -y docker.io**

**Steps:-**

**Step-1:- I have created a aws ec2 instance and launched it**

**Step-2:- Now installed docker**

**Cmd:- apt-get install -y docker.io**

****

**Step-3:- I have pulled the postgres image from docker hub**

**Cmd:- docker pull postgres**

**Step-4:- created a home location**

**Cmd:-mkdir ${Home}/postgres-data/**

**Step-5:- have given the host and port numbers username password**

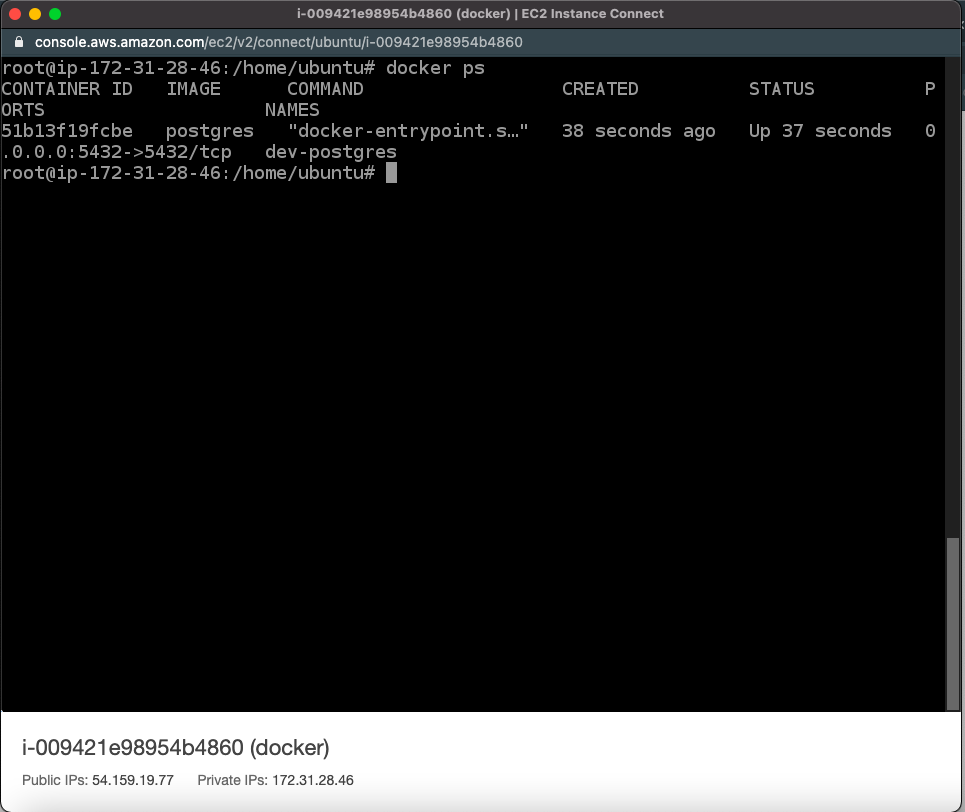
**Cmd:- docker run -d –name dev-postgres -e POSTGRES\_PASSWORD=Pass2021! -v ${HOME}/postgres-data/:var/lib/postgresql/data/ -p 5432:5432 postgres**

**Text

Description automatically generated**

**Step-6:- Check the whether it is running or not**

**Cmd:- docker ps**

****

**Step-7:- Now exec the postgressql with docker cmd which we have created the contaired**

**Cmd:- docker exec -it dev-postgres bash**

**Cmd:- psql -h localhost -U postgres**

**Text

Description automatically generated**

**Step-8:-Now start the pgadmin instance**

**Cmd:- docker pull dpage/pgadmin4**

**Text

Description automatically generated**

**Step-9:- now run the created container**

**Cmd:- docker run -p 80:80 -e ‘PGADMIN\_DEFAULT\_EMAIL=user@domain.local’ -e ‘PGADMIN\_DEFAULT\_PASSWORD=SuperSecret’ --name dev-pgadmin -d dpage/pgadmin4**

**Text

Description automatically generated**

**Step-10:- check the container is created or not**

**Cmd:- docker ps**

**Text

Description automatically generated**

**Step-11:-we will need to look the ipaddress of postgres using docker inspect command**

**Cmd:- docker inspect dev-postgres -f “{{json .NetworkSettings.Networks}}”**

**Text

Description automatically generated**

**Step-12:- we should go to the chrome and using ipadress and port**

**Cmd:-** 54.159.19.77:80

Graphical user interface, website

Description automatically generated

**Step-13:- we should login using details which we given while creating the containerGraphical user interface, text, application, email

Description automatically generated**