**1.create an alpine container in interactive mode and install python.**

docker container run -d -P --name sridhar alpine sleep 1d

docker container exec -it sridhar /bin/sh

apk add –update

apk add --update --no-cache python3 && ln -sf python3 /usr/bin/python

docker container ls -a

docker image inspect alpine

python3 –version

2. **create an ubuntu container with sleep 1d and then login using exec. Install python**

docker container run -d -p 35000:8080 --name sridhar1  ubuntu:latest sleep 1d

docker container run -it ubuntu /bin/bash

apt update

apt-get install -y python3

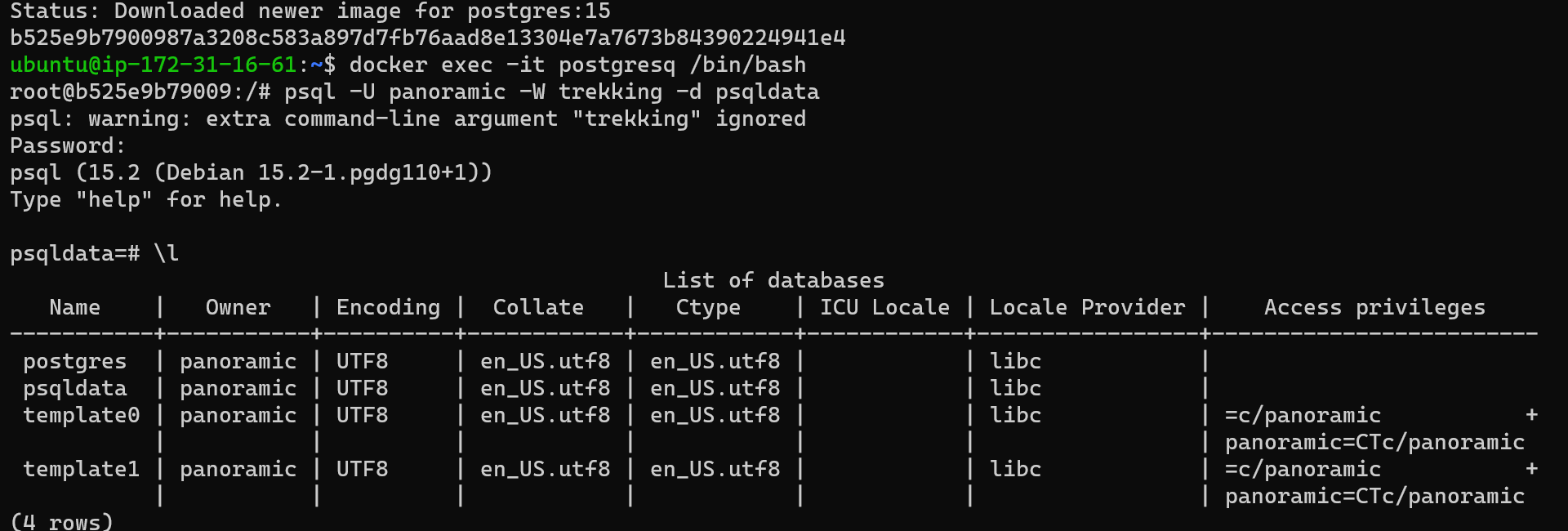
python3 –version

**3. Create a Postgres container with user & password**

docker container run --name some-postgresq -e POSTGRES\_USER=panoramic -e POSTGRES\_PASSWORD=trekking -e POSTGRES\_DB=psql -d -P postgres

docker exec -it some-postgresq /bin/bash

psql -U panoramic -W trekking



**4. Try creating dockerfile which runs on phpinfo page,user ARG & ENV whereever appropriate a) on apache server**

FROM ubuntu/apache2

LABEL Author="Sridhar" Organization="QT" Project="Learning"

RUN apt update

RUN apt install php libapache2-mod-php -y

RUN apt install php-cli -y

RUN apt install php-cgi -y

RUN apt install php-mysql -y

RUN apt install php-pgsql -y

RUN echo "<?php phpinfo(); ?>" >> /var/www/html/info.php

EXPOSE 80

**5.create a Jenkins image by creating your won dockerfile**

FROM ubuntu:22.04

LABEL Author="Sridhar" Organization="QT" Project="Learning"

RUN apt update

RUN apt install openjdk-11-jdk -y

RUN apt install curl -y

RUN curl -fsSL https://pkg.jenkins.io/debian-stable/jenkins.io-2023.key |  tee \

  /usr/share/keyrings/jenkins-keyring.asc > /dev/null

RUN echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \

  https://pkg.jenkins.io/debian-stable binary/ |  tee \

  /etc/apt/sources.list.d/jenkins.list > /dev/null

RUN apt update && apt install jenkins -y

EXPOSE 8080

CMD ["/usr/bin/jenkins"]

**6.Create Nopcommerce and my-sql server containers and try to make them work by configuring**

1.docker network create mybridge

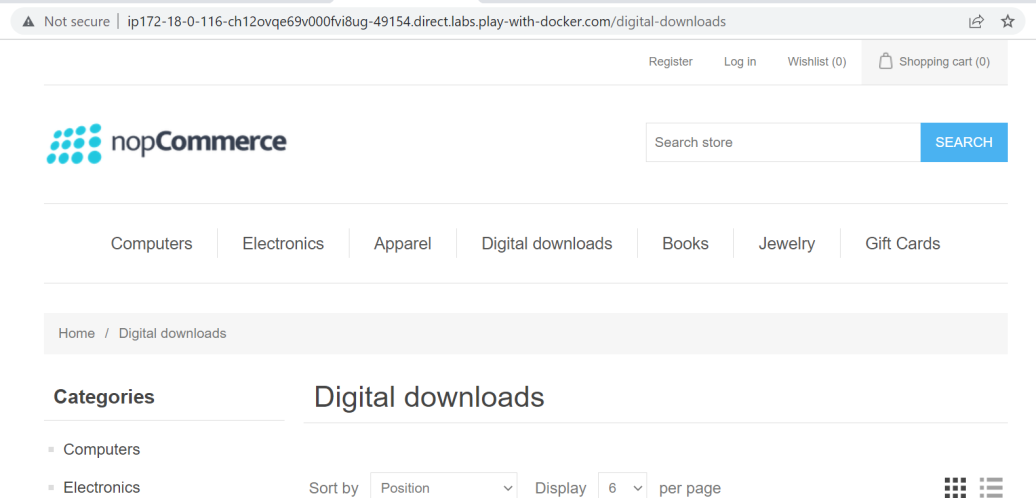
2.docker run -d --network mybridge --name mysri -v myvolume:/var/lib/mysql -e MYSQL\_ROOT\_PASSWORD=sridhar -e MYSQL\_DATABASE=employees -e MYSQL\_USER=qtdevops -e MYSQL\_PASSWORD=rootroot mysql:8

  vi Dockerfile

3.docker image build -t nopimage:4.60.2 .

4.docker container run -d -P --network mybridge --name nop -v myvolume:/var/lib/mysql -e

   MYSQL\_ROOT\_PASSWORD=sridhar -e MYSQL\_DATABASE=employees  nopimage:4.60.2

****