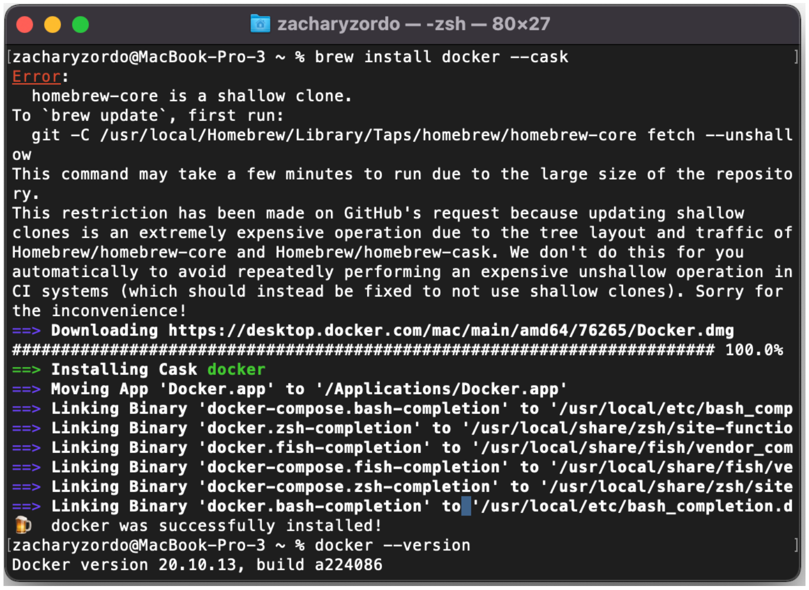
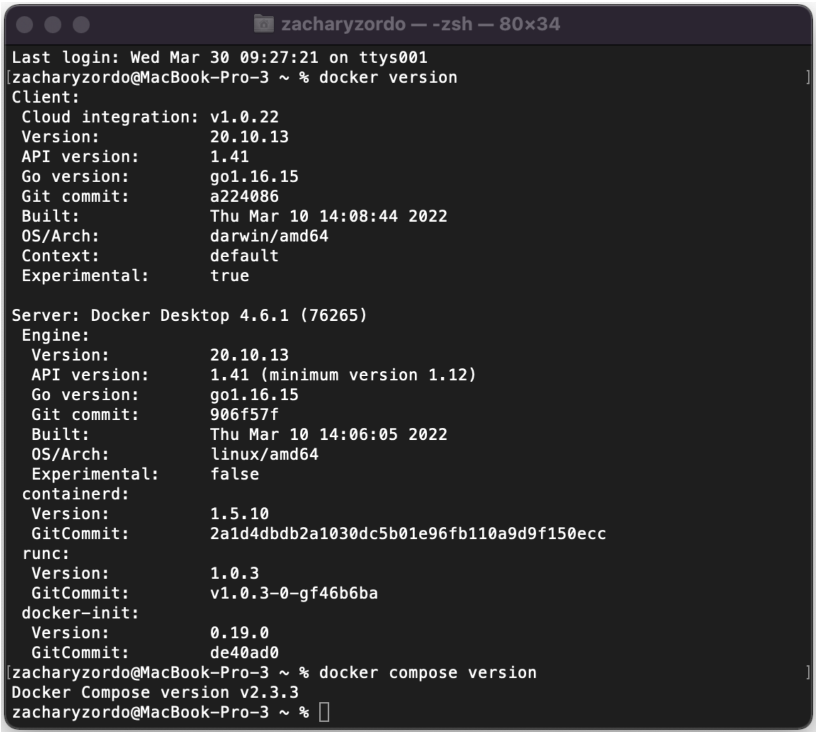
Q3 – Docker Support Engineer Test

1. Install Dependencies



```
[zacharyzordo@MacBook-Pro-3 ~ % brew install docker --cask
Error:
  homebrew-core is a shallow clone.
To `brew update`, first run:
  git -C /usr/local/Homebrew/Library/Taps/homebrew/homebrew-core fetch --unshall
ow
This command may take a few minutes to run due to the large size of the reposito
ry.
This restriction has been made on GitHub's request because updating shallow
clones is an extremely expensive operation due to the tree layout and traffic of
Homebrew/homebrew-core and Homebrew/homebrew-cask. We don't do this for you
automatically to avoid repeatedly performing an expensive unshallow operation in
CI systems (which should instead be fixed to not use shallow clones). Sorry for
the inconvenience!
==> Downloading https://desktop.docker.com/mac/main/amd64/76265/Docker.dmg
######################################################################## 100.0%
==> Installing Cask docker
==> Moving App 'Docker.app' to '/Applications/Docker.app'
==> Linking Binary 'docker-compose.bash-completion' to '/usr/local/etc/bash_comp
==> Linking Binary 'docker.zsh-completion' to '/usr/local/share/zsh/site-functio
==> Linking Binary 'docker.fish-completion' to '/usr/local/share/fish/vendor_com
==> Linking Binary 'docker-compose.fish-completion' to '/usr/local/share/fish/ve
==> Linking Binary 'docker-compose.zsh-completion' to '/usr/local/share/zsh/site
==> Linking Binary 'docker.bash-completion' to '/usr/local/etc/bash_completion.d
🍺  docker was successfully installed!
[zacharyzordo@MacBook-Pro-3 ~ % docker --version
Docker version 20.10.13, build a224086
```

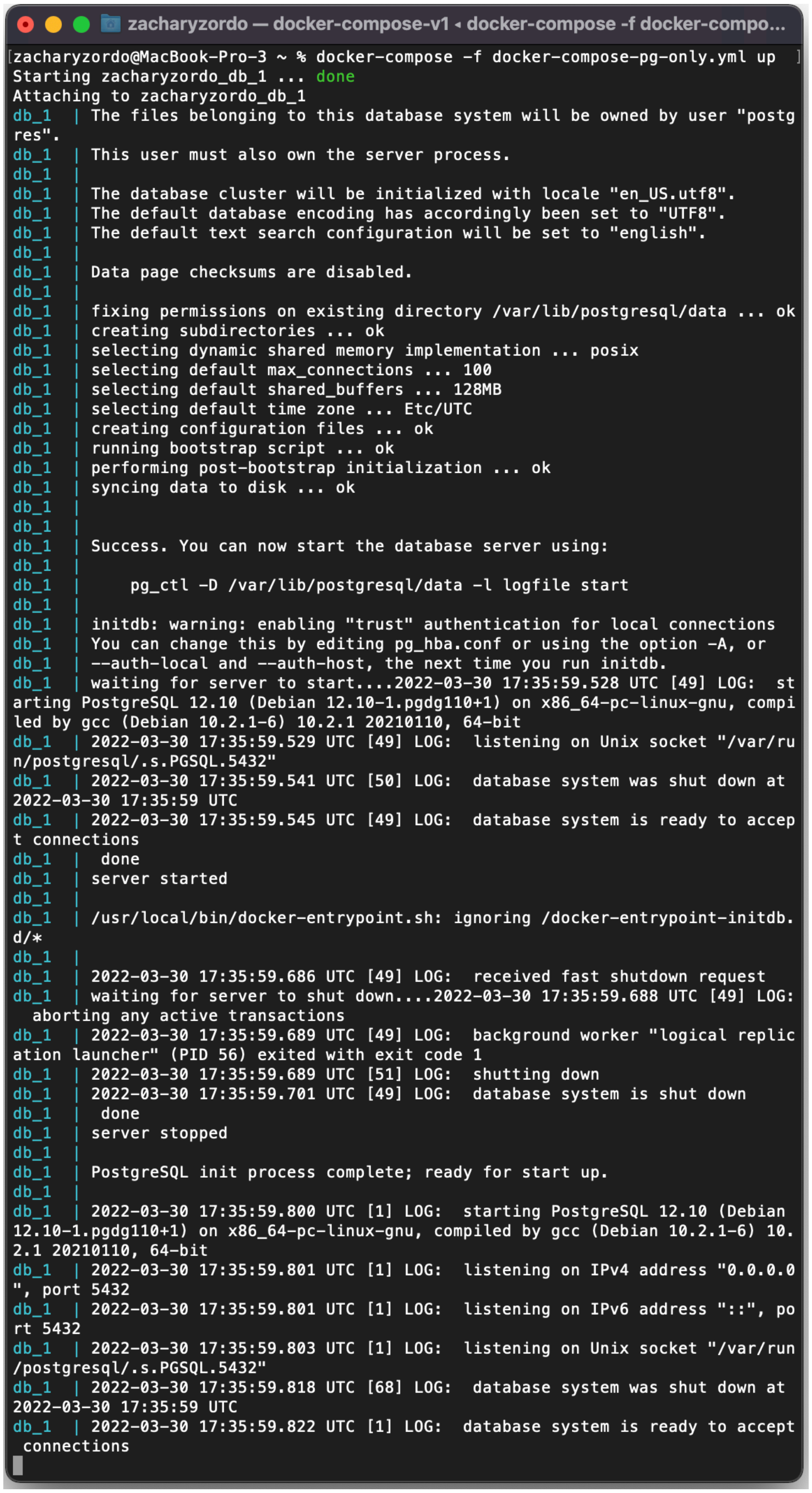
2. Check Docker & Docker Compose Version



```
Last login: Wed Mar 30 09:27:21 on ttys001
[zacharyzordo@MacBook-Pro-3 ~ % docker version
Client:
 Cloud integration: v1.0.22
 Version:           20.10.13
 API version:       1.41
 Go version:        go1.16.15
 Git commit:        a224086
 Built:             Thu Mar 10 14:08:44 2022
 OS/Arch:           darwin/amd64
 Context:           default
 Experimental:      true

Server: Docker Desktop 4.6.1 (76265)
 Engine:
  Version:          20.10.13
  API version:      1.41 (minimum version 1.12)
  Go version:       go1.16.15
  Git commit:       906f57f
  Built:            Thu Mar 10 14:06:05 2022
  OS/Arch:          linux/amd64
  Experimental:     false
 containerd:
  Version:          1.5.10
  GitCommit:        2a1d4dbdb2a1030dc5b01e96fb110a9d9f150ecc
 runc:
  Version:          1.0.3
  GitCommit:        v1.0.3-0-gf46b6ba
 docker-init:
  Version:          0.19.0
  GitCommit:        de40ad0
[zacharyzordo@MacBook-Pro-3 ~ % docker compose version
Docker Compose version v2.3.3
zacharyzordo@MacBook-Pro-3 ~ % 
```

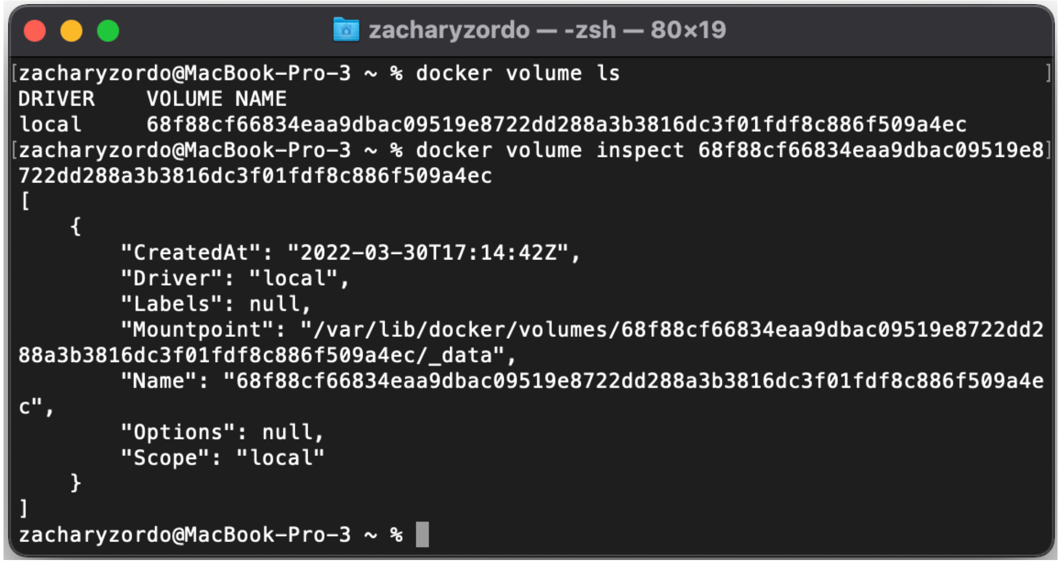
Docker Compose is packaged with Docker on Mac, so I was unable to rollback/downgrade the Docker Compose version to 1.25.3, however, I was able to proceed with all other steps requested.

3. Start postgres 12 instance using docker-compose


zacharyzordo — docker-compose-v1 ‹ docker-compose -f docker-compo...

```
[zacharyzordo@MacBook-Pro-3 ~ % docker-compose -f docker-compose-pg-only.yml up  ]
Starting zacharyzordo_db_1 ... done
Attaching to zacharyzordo_db_1
db_1  | The files belonging to this database system will be owned by user "postg
res".
db_1  | This user must also own the server process.
db_1  |
db_1  | The database cluster will be initialized with locale "en_US.utf8".
db_1  | The default database encoding has accordingly been set to "UTF8".
db_1  | The default text search configuration will be set to "english".
db_1  |
db_1  | Data page checksums are disabled.
db_1  |
db_1  | fixing permissions on existing directory /var/lib/postgresql/data ... ok
db_1  | creating subdirectories ... ok
db_1  | selecting dynamic shared memory implementation ... posix
db_1  | selecting default max_connections ... 100
db_1  | selecting default shared_buffers ... 128MB
db_1  | selecting default time zone ... Etc/UTC
db_1  | creating configuration files ... ok
db_1  | running bootstrap script ... ok
db_1  | performing post-bootstrap initialization ... ok
db_1  | syncing data to disk ... ok
db_1  |
db_1  |
db_1  | Success. You can now start the database server using:
db_1  |
db_1  |     pg_ctl -D /var/lib/postgresql/data -l logfile start
db_1  |
db_1  | initdb: warning: enabling "trust" authentication for local connections
db_1  | You can change this by editing pg_hba.conf or using the option -A, or
db_1  | --auth-local and --auth-host, the next time you run initdb.
db_1  | waiting for server to start....2022-03-30 17:35:59.528 UTC [49] LOG:  st
arting PostgreSQL 12.10 (Debian 12.10-1.pgdg110+1) on x86_64-pc-linux-gnu, compi
led by gcc (Debian 10.2.1-6) 10.2.1 20210110, 64-bit
db_1  | 2022-03-30 17:35:59.529 UTC [49] LOG:  listening on Unix socket "/var/ru
n/postgresql/.s.PGSQL.5432"
db_1  | 2022-03-30 17:35:59.541 UTC [50] LOG:  database system was shut down at
2022-03-30 17:35:59 UTC
db_1  | 2022-03-30 17:35:59.545 UTC [49] LOG:  database system is ready to accep
t connections
db_1  |  done
db_1  | server started
db_1  |
db_1  | /usr/local/bin/docker-entrypoint.sh: ignoring /docker-entrypoint-initdb.
d/*
db_1  |
db_1  | 2022-03-30 17:35:59.686 UTC [49] LOG:  received fast shutdown request
db_1  | waiting for server to shut down....2022-03-30 17:35:59.688 UTC [49] LOG:
  aborting any active transactions
db_1  | 2022-03-30 17:35:59.689 UTC [49] LOG:  background worker "logical replic
ation launcher" (PID 56) exited with exit code 1
db_1  | 2022-03-30 17:35:59.689 UTC [51] LOG:  shutting down
db_1  | 2022-03-30 17:35:59.701 UTC [49] LOG:  database system is shut down
db_1  |  done
db_1  | server stopped
db_1  |
db_1  | PostgreSQL init process complete; ready for start up.
db_1  |
db_1  | 2022-03-30 17:35:59.800 UTC [1] LOG:  starting PostgreSQL 12.10 (Debian
12.10-1.pgdg110+1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 10.2.1-6) 10.
2.1 20210110, 64-bit
db_1  | 2022-03-30 17:35:59.801 UTC [1] LOG:  listening on IPv4 address "0.0.0.0
", port 5432
db_1  | 2022-03-30 17:35:59.801 UTC [1] LOG:  listening on IPv6 address "::", po
rt 5432
db_1  | 2022-03-30 17:35:59.803 UTC [1] LOG:  listening on Unix socket "/var/run
/postgresql/.s.PGSQL.5432"
db_1  | 2022-03-30 17:35:59.818 UTC [68] LOG:  database system was shut down at
2022-03-30 17:35:59 UTC
db_1  | 2022-03-30 17:35:59.822 UTC [1] LOG:  database system is ready to accept
 connections
```

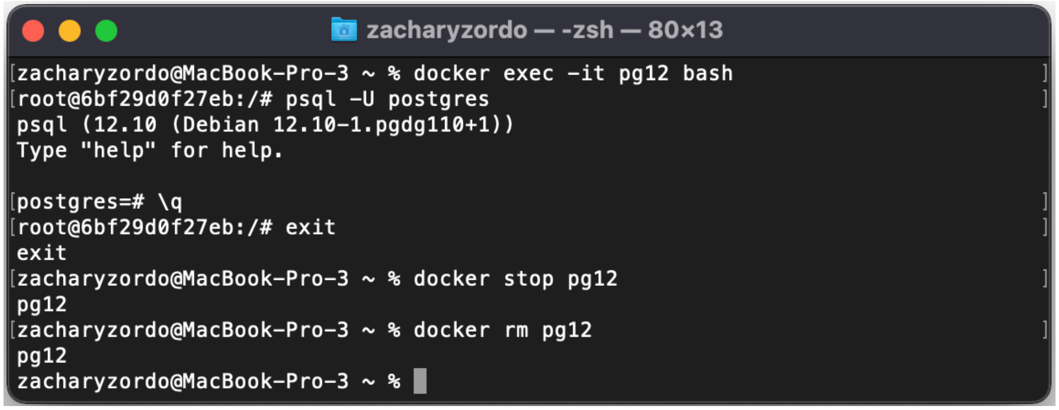
4. Check Docker Volume Location



```
zacharyzordo@MacBook-Pro-3 ~ % docker volume ls
DRIVER    VOLUME NAME
local     68f88cf66834eaa9dbac09519e8722dd288a3b3816dc3f01fdf8c886f509a4ec
zacharyzordo@MacBook-Pro-3 ~ % docker volume inspect 68f88cf66834eaa9dbac09519e8
722dd288a3b3816dc3f01fdf8c886f509a4ec
[
    {
        "CreatedAt": "2022-03-30T17:14:42Z",
        "Driver": "local",
        "Labels": null,
        "Mountpoint": "/var/lib/docker/volumes/68f88cf66834eaa9dbac09519e8722dd2
88a3b3816dc3f01fdf8c886f509a4ec/_data",
        "Name": "68f88cf66834eaa9dbac09519e8722dd288a3b3816dc3f01fdf8c886f509a4e
c",
        "Options": null,
        "Scope": "local"
    }
]
zacharyzordo@MacBook-Pro-3 ~ %
```

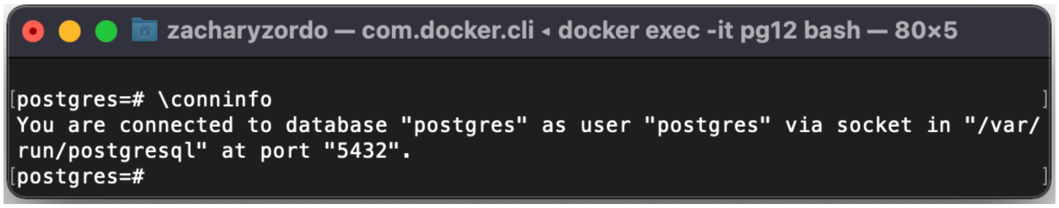
5. Stop the instance
   (See screenshot for Step 6)

6. Destroy any trace of Docker instances



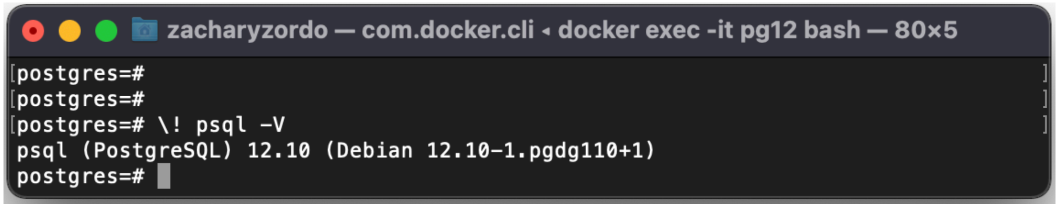
```
zacharyzordo@MacBook-Pro-3 ~ % docker exec -it pg12 bash
root@6bf29d0f27eb:/# psql -U postgres
psql (12.10 (Debian 12.10-1.pgdg110+1))
Type "help" for help.

postgres=# \q
root@6bf29d0f27eb:/# exit
exit
zacharyzordo@MacBook-Pro-3 ~ % docker stop pg12
pg12
zacharyzordo@MacBook-Pro-3 ~ % docker rm pg12
pg12
zacharyzordo@MacBook-Pro-3 ~ %
```

7. Check postgres port


zacharyzordo — com.docker.cli ◂ docker exec -it pg12 bash — 80×5

```
postgres=# \conninfo
You are connected to database "postgres" as user "postgres" via socket in "/var/
run/postgresql" at port "5432".
postgres=#
```

8. Check postgresql version



zacharyzordo — com.docker.cli ◂ docker exec -it pg12 bash — 80×5

```
[postgres=#
[postgres=#
[postgres=# \! psql -V
psql (PostgreSQL) 12.10 (Debian 12.10-1.pgdg110+1)
postgres=#
```