

Cloud Data Centres

Lab 2

Jack Foley — C00274246

March 2025

Contents

1	Publishing and Exposing Ports	3
1.1	Publishing Ports	3
1.2	Ephemeral Ports	3
1.3	Publishing All Ports	3
1.4	Final step	4
1.5	Docker Compose	5
2	Overriding Container Defaults	6
2.1	Running Multiple Postgres Databases	6
2.2	Postgres with Networks	6
2.3	Managing the Resources	6
2.4	Override Defaults in Docker Compose	6
2.5	Override Defaults with docker run	7
3	Persistent Container Data	8
3.1	Using Volumes	8
3.2	View Volume Contents	8
3.3	Removing Volumes	9
4	Sharing Local Files with Containers	10

1 Publishing and Exposing Ports

1.1 Publishing Ports

```
+ ~ docker run -p 8080:80 nginx
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2025/03/05 12:26:57 [notice] 1#1: using the "epoll" event method
2025/03/05 12:26:57 [notice] 1#1: nginx/1.27.4
2025/03/05 12:26:57 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
2025/03/05 12:26:57 [notice] 1#1: OS: Linux 6.1.0-31-amd64
2025/03/05 12:26:57 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
2025/03/05 12:26:57 [notice] 1#1: start worker processes
2025/03/05 12:26:57 [notice] 1#1: start worker process 29
2025/03/05 12:26:57 [notice] 1#1: start worker process 30
2025/03/05 12:26:57 [notice] 1#1: start worker process 31
2025/03/05 12:26:57 [notice] 1#1: start worker process 32
2025/03/05 12:26:57 [notice] 1#1: start worker process 33
2025/03/05 12:26:57 [notice] 1#1: start worker process 34
2025/03/05 12:26:57 [notice] 1#1: start worker process 35
2025/03/05 12:26:57 [notice] 1#1: start worker process 36
2025/03/05 12:26:57 [notice] 1#1: start worker process 37
2025/03/05 12:26:57 [notice] 1#1: start worker process 38
2025/03/05 12:26:57 [notice] 1#1: start worker process 39
2025/03/05 12:26:57 [notice] 1#1: start worker process 40
```

1.2 Ephemeral Ports

Had some issues making it run in the background so I could check the port via `docker ps`. Adding the `-d` flag allowed it to run detached so I could view the port. Other than that, no issues.

```
+ CloudDataCentres git:(main) x docker run -d -p 80 nginx
0a51594e7eb79b6f613b98b952ecea3111e50797c236dbd6781332649109c8c
+ CloudDataCentres git:(main) x docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
0a51594e7eb7   nginx     "/docker-entrypoint..." 2 seconds ago  Up 2 seconds  0.0.0.0:32775->80/tcp, :::32775->80/tcp  elated_kare
+ CloudDataCentres git:(main) x tGPT4o
```

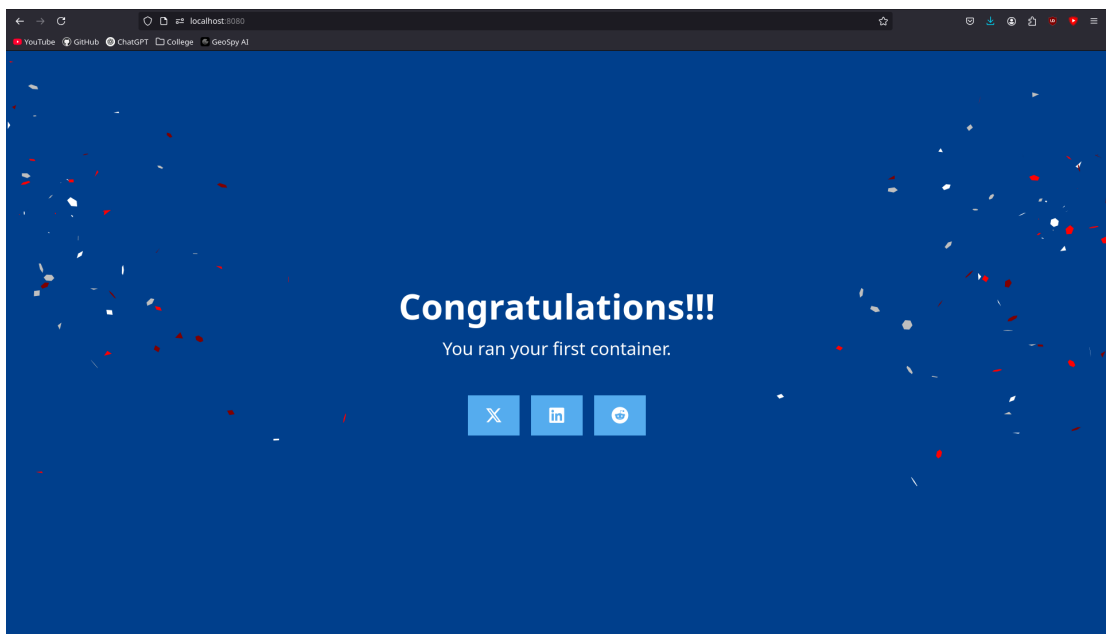
1.3 Publishing All Ports

```
+ CloudDataCentres git:(main) x docker run -d -P nginx
62cb051970f2eaeaf748d5c72fd73f2be08cf42f37e46766301f9931aac23afa
+ CloudDataCentres git:(main) x docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
62cb051970f2   nginx     "/docker-entrypoint..." 1 second ago   Up 1 second   0.0.0.0:32776->80/tcp, :::32776->80/tcp  dazzling_yalow
+ CloudDataCentres git:(main) x
```

1.4 Final step

Now it was time to actually run the `welcome-to-docker` container. It first downloaded the image since it was not found locally, then went and ran the container on host port 8080 and container port 80.

```
→ CloudDataCentres git:(main) x docker run -d -p 8080:80 docker/welcome-to-docker
Unable to find image 'docker/welcome-to-docker:latest' locally
latest: Pulling from docker/welcome-to-docker
96526aa774ef: Pull complete
740091335c74: Pull complete
da9c2e764c5b: Pull complete
ade17ad21ef4: Pull complete
4e6f462c8a69: Pull complete
1324d9977cd2: Pull complete
1b9b96da2c74: Pull complete
5d329b1e101a: Pull complete
Digest: sha256:eedaff45e3c78538087bdd9dc7afafac7e110061bbdd836af4104b10f10ab693
Status: Downloaded newer image for docker/welcome-to-docker:latest
3912876ffd33679f4f6051700f5417bff101ae37bf5fc240e93fa711576454cb
→ CloudDataCentres git:(main) x
```



1.5 Docker Compose

```
Lab2 git:(main) docker compose up
[+] Running 2/2
 ✓ Network lab2_default Created
 ✓ Container lab2-app-1 Created
Attaching to app-1
app-1 | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
app-1 | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
app-1 | /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
app-1 | 10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
app-1 | 10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
app-1 | /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
app-1 | /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
app-1 | /docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
app-1 | /docker-entrypoint.sh: Configuration complete; ready for start up
app-1 | 2025/03/05 12:45:28 [notice] 1#1: using the "epoll" event method
app-1 | 2025/03/05 12:45:28 [notice] 1#1: nginx/1.25.3
app-1 | 2025/03/05 12:45:28 [notice] 1#1: built by gcc 12.2.1 20220924 (Alpine 12.2.1_git20220924-r10)
app-1 | 2025/03/05 12:45:28 [notice] 1#1: OS: Linux 6.1.0-31-amd64
app-1 | 2025/03/05 12:45:28 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker processes
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 30
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 31
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 32
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 33
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 34
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 35
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 36
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 37
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 38
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 39
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 40
app-1 | 2025/03/05 12:45:28 [notice] 1#1: start worker process 41
```

2 Overriding Container Defaults

2.1 Running Multiple Postgres Databases

```
+ ~ docker run -d -e POSTGRES_PASSWORD=secret -p 5432:5432 postgres
995ce5b05ae4b507fb5b1f9ab3512845598ad52510bbf8635bd3598b783dd92
+ ~ docker run -d -e POSTGRES_PASSWORD=secret -p 5433:5432 postgres
f185d1a385e2f125d181a1f1550b67b1cf2d9d8161cf77a6ba6a9083c788ce7
+ ~ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
f185d1a385e2   postgres "docker-entrypoint.s..." 8 seconds ago Up 6 seconds  0.0.0.0:5433->5432/tcp, :::5433->5432/tcp   tender_wescoff
995ce5b05ae4   postgres "docker-entrypoint.s..." 21 seconds ago Up 20 seconds  0.0.0.0:5432->5432/tcp, :::5432->5432/tcp   recurring_franklin
```

2.2 Postgres with Networks

```
+ ~ docker network create mynetwork
cdf6d5783401971f8f4983f97e8a6bac4d8ff49f46a93aaa6f5fd53a17b78a00
+ ~ docker network ls
NETWORK ID     NAME      DRIVER    SCOPE
2509aec7b76    bridge    bridge    local
395c3436221d    host      host      local
14328201155d    lab2_default bridge    local
cdf6d5783401    mynetwork bridge    local
89779ea5da3e    none      null      local
+ ~ docker run -d -e POSTGRES_PASSWORD=secret -p 5434:5432 --network mynetwork postgres
c77b87803d5d0bfff3bdf6c3dbcd9d2a0a74925516c95de1a730c7712b8775ed
```

2.3 Managing the Resources

```
+ ~ docker run -d -e POSTGRES_PASSWORD=secret --memory="512m" --cpus=".5" postgres
921a131ba54ada93a6a4b8708f6364ff31d3ac18fc531b3204f30cf33d4d0762c4df5dfdb44d0c59
```

2.4 Override Defaults in Docker Compose

```
Lab2 git:(main) x cat docker-compose.yml
services:
  postgres:
    image: postgres
    entrypoint: ["docker-entrypoint.sh", "postgres"]
    command: ["-h", "localhost", "-p", "5432"]
    environment:
      POSTGRES_PASSWORD: secret
Lab2 git:(main) x docker compose up -d
Running 1/1
Container lab2-postgres-1 Started
Lab2 git:(main) x docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                               NAMES
ee3b5e2af7fc   postgres "docker-entrypoint.s..." 15 seconds ago Up 15 seconds  0.0.0.0:5432->5432/tcp               lab2-postgres-1
921a131ba54a   postgres "docker-entrypoint.s..." 2 minutes ago  Up 2 minutes  5432/tcp                           jovial_hawking
c77b87803d5d   postgres "docker-entrypoint.s..." 3 minutes ago  Up 3 minutes  0.0.0.0:5434->5432/tcp, :::5434->5432/tcp   vigorous_mclean
Lab2 git:(main) x docker exec -it ee3b5e2af7fc /bin/bash
root@ee3b5e2af7fc:/# psql -U postgres
psql (17.4 (Debian 17.4-1.pgdg120+2))
Type "help" for help.

postgres=#
```

2.5 Override Defaults with docker run

```
lab2@git:(main) ✗ docker run -e POSTGRES_PASSWORD=secret postgres docker-entrypoint.sh -h localhost -p 5432
The files belonging to this database system will be owned by user "postgres".
This user must also own the server process.

The database cluster will be initialized with locale "en_US.utf8".
The default database encoding has accordingly been set to "UTF8".
The default text search configuration will be set to "english".
Data page checksums are disabled.

fixing permissions on existing directory /var/lib/postgresql/data ... ok
creating subdirectories ... ok
selecting dynamic shared memory implementation ... posix is run in the background so I could check the
selecting default "max_connections" ... 100
selecting default "shared_buffers" ... 128MB ok as I could view the port. Other than that, no issues.
selecting default time zone ... Etc/UTC
creating configuration files ... ok
running bootstrap script ... ok
performing post-bootstrap initialization ... ok
syncing data to disk ... ok

Success. You can now start the database server using:
    pg_ctl -D /var/lib/postgresql/data -l logfile start

initdb: warning: enabling "trust" authentication for local connections
initdb: hint: You can change this by editing pg_hba.conf or using the option -A, or --auth-local and --auth-host, the next time you run initdb.
waiting for server to start... 2025-03-05 14:41:41.363 UTC [48] LOG: starting PostgreSQL 17.4 (Debian 17.4-1.pgdg120+2) on x86_64-pc-linux-gnu, compiled by gcc (Debian 12.2.0-14) 12.2.0, 64-bit
2025-03-05 14:41:41.364 UTC [48] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
2025-03-05 14:41:41.366 UTC [51] LOG: database system was shut down at 2025-03-05 14:41:41 UTC
2025-03-05 14:41:41.368 UTC [48] LOG: database system is ready to accept connections
done
server started

/usr/local/bin/docker-entrypoint.sh: ignoring /docker-entrypoint-initdb.d/*

2025-03-05 14:41:41.485 UTC [48] LOG: received fast shutdown request
waiting for server to shut down... 2025-03-05 14:41:41.488 UTC [48] LOG: aborting any active transactions
2025-03-05 14:41:41.489 UTC [48] LOG: background worker "logical replication launcher" (PID 54) exited with exit code 1
2025-03-05 14:41:41.489 UTC [49] LOG: shutting down
2025-03-05 14:41:41.490 UTC [49] LOG: checkpoint starting: shutdown immediate
2025-03-05 14:41:41.492 UTC [49] LOG: checkpoint complete: write 2 buffers (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=0.001 s, sync=0.001 s, total=0.004 s; sync files=2, longest=0.001 s, average=0.001 s; distance=0 kB, estimate=0 kB; lsn=0/14E4FA0, redo lsn=0/14E4FA0
2025-03-05 14:41:41.496 UTC [48] LOG: database system is shut down
done
server stopped

PostgreSQL init process complete; ready for start up.

2025-03-05 14:41:41.609 UTC [1] LOG: starting PostgreSQL 17.4 (Debian 17.4-1.pgdg120+2) on x86_64-pc-linux-gnu, compiled by gcc (Debian 12.2.0-14) 12.2.0, 64-bit
2025-03-05 14:41:41.609 UTC [1] LOG: listening on IPv4 address "127.0.0.1", port 5432
2025-03-05 14:41:41.609 UTC [1] LOG: could not bind IPv6 address ":::1": Cannot assign requested address
2025-03-05 14:41:41.610 UTC [1] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
2025-03-05 14:41:41.612 UTC [62] LOG: database system was shut down at 2025-03-05 14:41:41 UTC
2025-03-05 14:41:41.615 UTC [1] LOG: database system is ready to accept connections
```

3 Persistent Container Data

3.1 Using Volumes

```
+ Lab2 git:(main) * docker run --name=db -e POSTGRES_PASSWORD=secret -d -v postgres_data:/var/lib/postgresql/data postgres
b42e31e4323573fc40921ac37dcd64db7bcacb33fabf284376a5831e30c08b6a
+ Lab2 git:(main) * ls
1.png /docker-compose.yml /doc.odt
+ Lab2 git:(main) * docker exec -ti db psql -U postgres
psql (17.4 (Debian 17.4-1.pgdgl20+2))
Type "help" for help.

postgres=# CREATE TABLE tasks (
      id SERIAL PRIMARY KEY,
      description VARCHAR(100)
);
INSERT INTO tasks (description) VALUES ('Finish work'), ('Have fun');
CREATE TABLE
INSERT 0 2
postgres=# SELECT * FROM tasks;
 id | description
----+-----
  1 | Finish work
  2 | Have fun
(2 rows)

postgres=# \q
+ Lab2 git:(main) * docker stop db
docker rm db
+ Lab2 git:(main) * docker run --name=new-db -d -v postgres_data:/var/lib/postgresql/data postgres
c4be2d1f93a4c3b8770b6bd4f2df85598907578759a16c7d291e351e6609f983
+ Lab2 git:(main) * docker exec -ti new-db psql -U postgres -c "SELECT * FROM tasks"
 id | description
----+-----
  1 | Finish work
  2 | Have fun
(2 rows)
```

3.2 View Volume Contents

I do not have docker desktop, therefore I cannot view the volume contents as expect in the documentation I was following. Instead, I ran docker volume inspect in the CLI.


```

→ Lab2 git:(main) x docker volume inspect postgres_data
[
  {
    "CreatedAt": "2025-03-07T14:02:00Z",
    "Driver": "local",
    "Labels": null,
    "Mountpoint": "/var/lib/docker/volumes/postgres_data/_data",
    "Name": "postgres_data",
    "Options": null,
    "Scope": "local"
  }
]
→ Lab2 git:(main) x |

```

3.3 Removing Volumes

```

→ Lab2 git:(main) x docker rm -f new-db
new-db
→ Lab2 git:(main) x docker volume rm postgres_data
postgres_data
→ Lab2 git:(main) x docker volume prune
WARNING! This will remove all local volumes not used by at least one container.
Are you sure you want to continue? [y/N] y
Deleted Volumes:
527e019891ba7160f53cbef5fefbbd66a31d223fda97ca38d0937df03366632f
778838fad24ead281dcec9c74e65cf4a617217d12a1397587a010d97a0c072c5
f35859a568a9c6382c5391e6a8900caa2905ba3fc238c1584a87e9528e9fa59d
3188512df8f7a1c75768abe9d7025e7e7314547a633fec3873c6bb4a58b1a90e
7606232a2267ccc205af6550acdecbb18c090d80ad7ad40a26ae5284caee9e99d
83de92343be9f1ca65a215d64a827242bac31d295fee1f1aa2750f8302df615e
mcserver_mc_data
Total reclaimed space: 441.2MB
→ Lab2 git:(main) x |

```

4 Sharing Local Files with Containers