

# High Availability, Load Balancing, and Replication

## What is High Availability?

This cluster structure aims to provide accessibility in any case, if an error occurs in any server, by taking over the related operation by another server in the cluster. In this case, the client performs its operations without being affected by anything.

### **failover**

If an error occurs in any of the servers connected to the clusters, a different server will take over the related process, and the other service will continue to run uninterrupted while dealing with the error. to this situation "**failover**It's called ".

## What is Load Balancing?

The primary purpose of Load Balancing (LB) clusters is to balance the loads of services operating under heavy load among more than one server, thereby increasing performance.

## What is replication?

With replication, a queryable copy of the live database is created. By running the queries for reporting purposes on this read-only server, the load on the master server is reduced.

# Replication setup on Docker

[From here](#) You can download the compose file or use a **docker-compose.yml** file we create it and write the following code in it. (If you download it, the name of the file **dockercompose.yml** change to .)

## docker-compose.yml

```
version: '2nd'

services:
  postgresql-master:
    image: 'bitnami/postgresql:latest'
    ports:
      - '5432'
    volumes:
      - 'postgresql_master_data:/bitnami/postgresql'
    environment:
      - POSTGRESQL_REPLICATION_MODE=master
      - POSTGRESQL_REPLICATION_USER=repl_user
      - POSTGRESQL_REPLICATION_PASSWORD=repl_password
      - POSTGRESQL_USERNAME=my_user
      - POSTGRESQL_PASSWORD=my_password
      - POSTGRESQL_DATABASE=my_database
  postgresql-slave:
    image: 'bitnami/postgresql:latest'
    ports:
      - '5432'
    depends_on:
      - postgresql-master
    environment:
      - POSTGRESQL_REPLICATION_MODE=slave
      - POSTGRESQL_REPLICATION_USER=repl_user
      - POSTGRESQL_REPLICATION_PASSWORD=repl_password
      - POSTGRESQL_MASTER_HOST=postgresql-master
      - POSTGRESQL_PASSWORD=my_password
      - POSTGRESQL_MASTER_PORT_NUMBER=5432

volumes:
  postgresql_master_data:
```

(You can change values such as username and password according to you.)

After this process, we run the following command at the location of this file.

```
docker-compose up
```

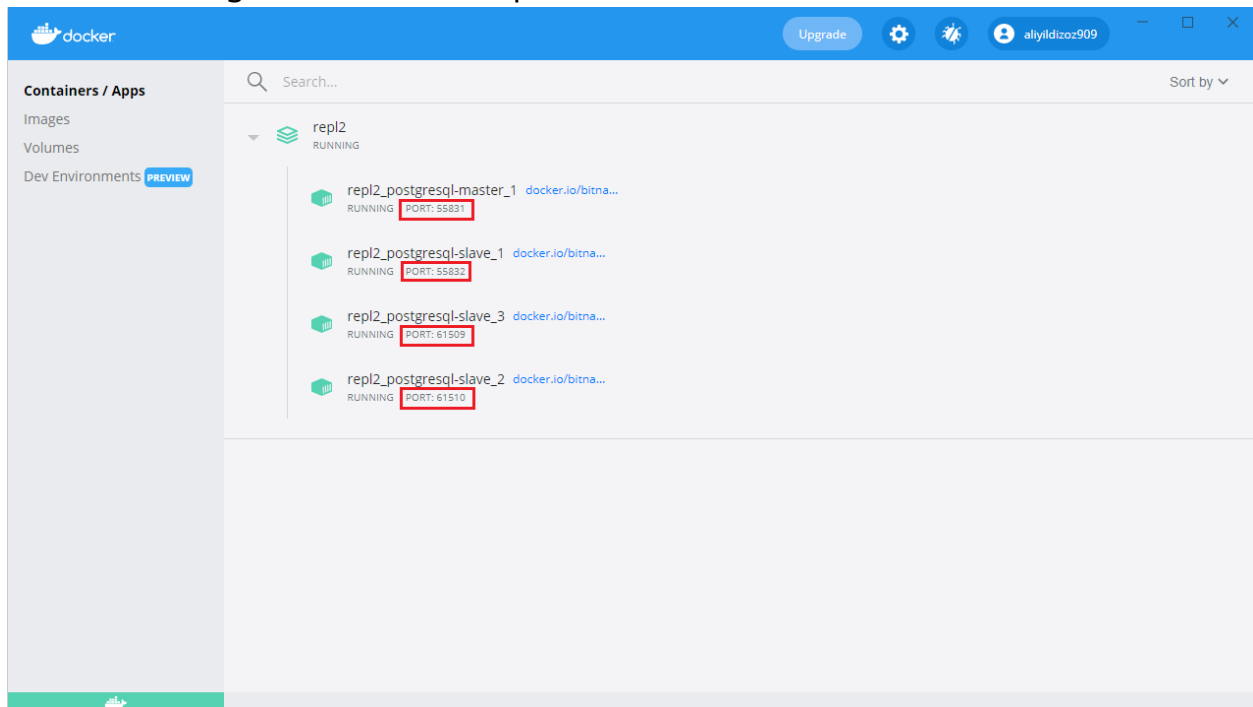
In this way, only 1 master and 1 slave server will be created.

**To scale by adding more slaves or masters, run the command below by entering the number of slaves and masters in the same folder.**

```
docker-compose up --detach --scale postgresql-master={master-number}-  
- scale postgresql-slave={slave-number}
```

## Connecting to servers

In order to connect to master or slave servers, we need to know the port numbers assigned to them on docker. Below are the port numbers of the servers running on docker desktop.



Server-Name	port number
master	55831
slave-1	55832
slave-2	61509
slave-3	61510

**Note:** Here I will connect through this example, you should do the same on which port it works.

### Connecting to master

- `psql -h localhost -U postgres -p55831`
- Password for user postgres:my\_password
- `postgres=#`

### Connecting to Slave-1

- `psql -h localhost -U postgres -p55832`
- Password for user postgres:my\_password
- `postgres=#`

### Connecting to Slave-2

- `psql -h localhost -U postgres -p61509`
- Password for user postgres:my\_password
- `postgres=#`

### Connecting to slave-3

- `psql -h localhost -U postgres -p61510`
- Password for user postgres:my\_password
- `postgres=#`

**Note:** Slave servers are read-only in these databases. **record, update** and **deletion** you can't just do **select** transactions can be made. If you do you will get the following error.

ERROR: cannot execute INSERT in a read-only transaction

## resources

- [bitnami/bitnami-docker-postgresql\(github.com\)](https://github.com/bitnami/bitnami-docker-postgresql)
- [AtifCeylan-ScalabilityAnd Redundancy In PostgreSQL.pdf](#)
- [Applied in PostgreSQL. \(Streaming Replication\)](#)
- [What is PostgreSQL Streaming Replication ? | Azbucuk](#)