

Nama : Mushlih Hakim ArifinHakim

NIM : 1227050098

---

1. Ketentuan service pertama (PostgreSQL)
  - a. Nama kontainer : postgres\_NAMA
  - b. Gunakan versi salah satu diantara 9-15, boleh custom
  - c. Gunakan username "postgres" dan password "ifunggul", tidak diperkenankan hardcode username dan password pada docker-compose.yml, sembunyikan kedua value tersebut dalam .env
  - d. Expose port untuk eksternal docker dengan ketentuan 22XXX (XXX diganti dengan tiga digit terakhir NIM masing-masing) (Skor: 40)
2. Ketentuan service kedua (PGADMIN4)
  - a. Nama kontainer : pgadmin\_NAMA
  - b. Gunakan PGADMIN4, boleh custom
  - c. Expose port untuk eksternal docker dengan ketentuan 44XXX (XXX diganti dengan tiga digit terakhir NIM masing-masing)
  - d. Buktikan PostgreSQL dapat diakses dari internal docker (PGADMIN4) dan eksternal docker (Dbeaver) (Skor: 20)

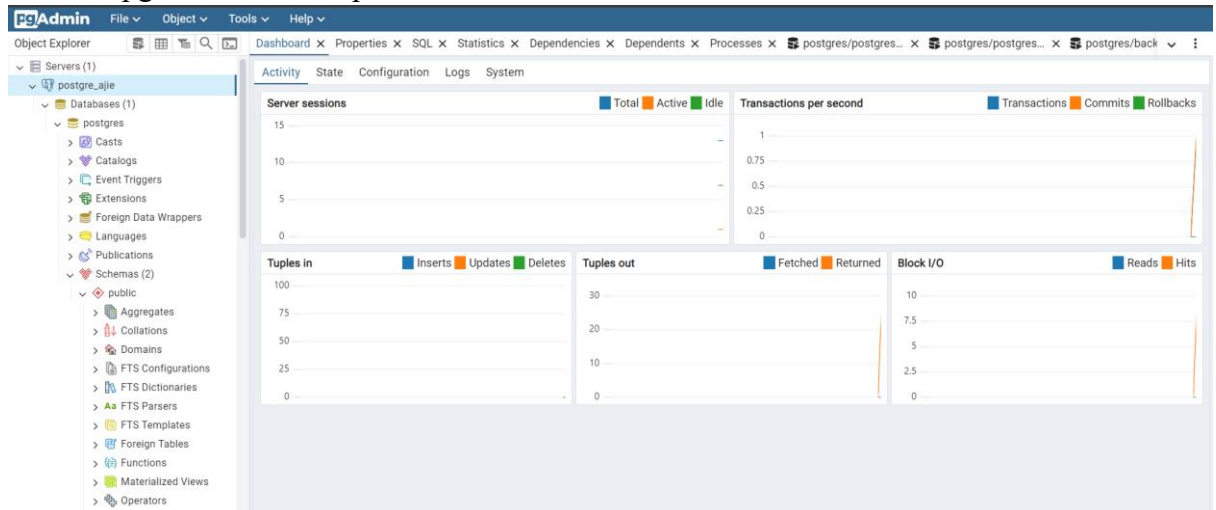
docker-composer.yml

```
docker-compose.yml
1  version: '3.7'
2
3  services:
4    postgres:
5      container_name: postgres_achmad_ajie_priyajie
6      image: postgres:latest
7      environment:
8        POSTGRES_USER: ${HOSTNAME}
9        POSTGRES_PASSWORD: ${PASSWORD}
10       PGDATA: /data/postgres
11      volumes:
12        - postgres:/data/postgres
13      ports:
14        - "${POSTGRES_PORT}:5432"
15      networks:
16        - postgres
17      restart: unless-stopped
18
19    pgadmin:
20      container_name: pgadmin_achmad_ajie_priyajie
21      image: dpage/pgadmin4
22      environment:
23        PGADMIN_DEFAULT_EMAIL: ${PGADMIN_DEFAULT_EMAIL}
24        PGADMIN_DEFAULT_PASSWORD: ${PGADMIN_DEFAULT_PASSWORD}
25        PGADMIN_CONFIG_SERVER_MODE: 'False'
26      volumes:
27        - pgadmin:/var/lib/pgadmin
28
29      ports:
30        - "${PGADMIN_PORT}:80"
31      networks:
32        - postgres
33      restart: unless-stopped
```

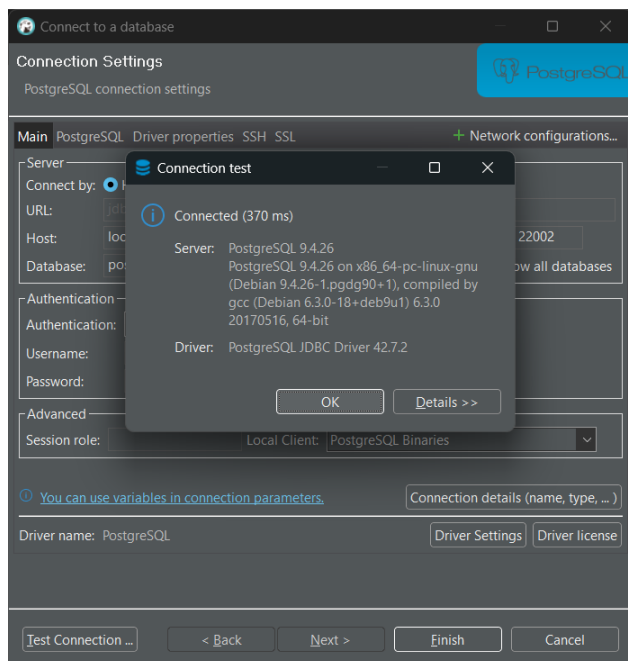
.env

```
HOSTNAME=postgres
PASSWORD=ifunggul
POSTGRES_PORT=22002
PGADMIN_DEFAULT_EMAIL=achmadajie74@gmail.com
PGADMIN_DEFAULT_PASSWORD=adminifunggul
PGADMIN_PORT=44002
```

## Koneksi pg admin dan tampilan dashboard:



## Koneksi dbEaver:



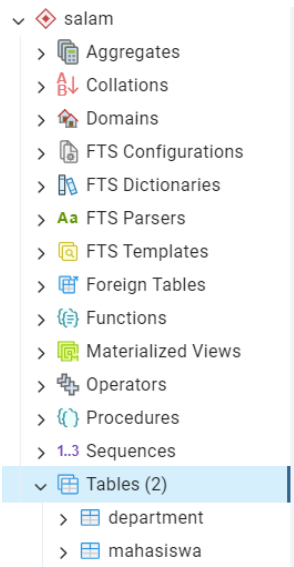
3. Buat skema dengan nama SALAM lalu didalamnya definisikan dan create tabel dengan nama *mahasiswa* paling tidak terdapat primary key, unique constraint, dan check constraint.

Buktikan bahwa constraintnya bekerja. (Skor: 10)

## Create table:

```
create table "salam".mahasiswa (  
    nim int primary key,  
    name varchar(64) not null,  
    department_id int,  
    score int check (score >= 80),  
    unique (nim)  
);  
  
create table "salam".department (  
    id int primary key,  
    name varchar(64) not null  
);  
  
ALTER TABLE "salam".mahasiswa  
ADD CONSTRAINT fk_department  
FOREIGN KEY (department_id) REFERENCES "salam".department(id);
```

## Schema:



## Insert table:

```
INSERT INTO "salam".department (id, name) VALUES
(1, 'Teknik Informatika'),
(2, 'Sistem Informasi'),
(3, 'Manajemen'),
(4, 'Ilmu Komunikasi');

INSERT INTO "salam".mahasiswa (nim, name, department_id, score) VALUES
(101, 'Ahmad Rahman', 1, 85),
(102, 'Siti Aisyah', 2, 90),
(103, 'Rizky Hidayat', 1, 88),
(104, 'Dian Permata', 3, 92),
(105, 'Fauzan Maulana', 4, 87);

SELECT * FROM "salam".mahasiswa;
SELECT * FROM "salam".department;
SELECT m.nim, m.name AS mahasiswa_name, d.name AS department_name, m.score
FROM "salam".mahasiswa m
JOIN "salam".department d
ON m.department_id = d.id;
```

## Output table:

	nim integer	mahasiswa_name character varying (64)	department_name character varying (64)	score integer
1	101	Ahmad Rahman	Teknik Informatika	85
2	102	Siti Aisyah	Sistem Informasi	90
3	103	Rizky Hidayat	Teknik Informatika	88
4	104	Dian Permata	Manajemen	92
5	105	Fauzan Maulana	Ilmu Komunikasi	87

## Primary Key:

[Query](#) [Query History](#)

```
1  INSERT INTO "salam".mahasiswa (nim, name, department_id, score) VALUES
2  (101, 'Ahmad Rahman', 1, 85),
3  (101, 'Ahmad Rahman', 1, 85);
```

[Data Output](#) [Messages](#) [Notifications](#)

ERROR: Key (nim)=(101) already exists.duplicate key value violates unique constraint "mahasiswa\_pkey"

ERROR: duplicate key value violates unique constraint "mahasiswa\_pkey"

SQL state: 23505

Detail: Key (nim)=(101) already exists.

## Not Null:

Query
Query History
Scratch Pa

1 **INSERT INTO "salam".mahasiswa** (nim, department\_id, score) **VALUES**  
2 (109, 1, 85);

Data Output
Messages
Notifications

ERROR: Failing row contains (109, null, 1, 85).null value in column "name" of relation "mahasiswa" violates not-null constraint  
  
ERROR: null value in column "name" of relation "mahasiswa" violates not-null constraint  
SQL state: 23502  
Detail: Failing row contains (109, null, 1, 85).

## Check Value:

Query
Query History
Scratch Pad x

1 **INSERT INTO "salam".mahasiswa** (nim, **name**, department\_id, score) **VALUES**  
2 (109, 'rafif sono aja', 1, 55);

Data Output
Messages
Notifications

ERROR: Failing row contains (109, rafif sono aja, 1, 55).new row for relation "mahasiswa" violates check constraint "mahasiswa\_score\_check"  
  
ERROR: new row for relation "mahasiswa" violates check constraint "mahasiswa\_score\_check"  
SQL state: 23514  
Detail: Failing row contains (109, rafif sono aja, 1, 55).

## 4. Buat user baru menggunakan SQL script sebagai berikut:

- Nama : backend\_dev, Role: CRUD semua table
- Nama : bi\_dev, Role: hanya read/select semua table/view
- Nama : data\_engineer, Role: CREATE, MODIFY, DROP semua objects, CRUD semua table

Buktikan bahwa aturannya bekerja (Skor: 30)

## Pembuatan User:

Query
Query History

1 **CREATE USER** backend\_dev **WITH PASSWORD** '123456';  
2 **GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA "salam" TO** backend\_dev;  
3 **ALTER DEFAULT PRIVILEGES IN SCHEMA "salam" GRANT SELECT, INSERT, UPDATE, DELETE ON TABLES TO** backend\_dev;  
4  
5 **CREATE USER** bi\_dev **WITH PASSWORD** '123456';  
6 **GRANT SELECT ON ALL TABLES IN SCHEMA "salam" TO** bi\_dev;  
7 **ALTER DEFAULT PRIVILEGES IN SCHEMA "salam" GRANT SELECT ON TABLES TO** bi\_dev;  
8  
9 **CREATE USER** data\_engineer **WITH PASSWORD** '123456';  
10 **GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA "salam" TO** data\_engineer;  
11 **GRANT CREATE ON SCHEMA "salam" TO** data\_engineer;  
12 **ALTER DEFAULT PRIVILEGES IN SCHEMA "salam" GRANT SELECT, INSERT, UPDATE, DELETE ON TABLES TO** data\_engineer;  
13  
14 **-- Memberikan hak akses pada schema salam**  
15 **GRANT USAGE ON SCHEMA "salam" TO** backend\_dev;  
16 **GRANT USAGE ON SCHEMA "salam" TO** bi\_dev;  
17 **GRANT USAGE ON SCHEMA "salam" TO** data\_engineer;  
18

## Backend\_dev:

Query
Query History

1 **-- Uji INSERT**  
2 **INSERT INTO "salam".mahasiswa** (nim, **name**, department\_id, score) **VALUES** (55, 'John Doe', 1, 85),  
3 (66, 'Sunu', 2, 88);  
4  
5  
6 **-- Uji UPDATE**  
7 **UPDATE "salam".mahasiswa SET** score = 90 **WHERE** nim = 1;  
8  
9 **-- Uji DELETE**  
10 **DELETE FROM "salam".mahasiswa WHERE** nim = 1;  
11  
12 **-- Uji SELECT**  
13 **SELECT \* FROM "salam".mahasiswa;**

8	55	John Doe	1	85
9	66	Sunu	2	88

Update score to 90:

8	66	Sunu	2	88
9	55	John Doe	1	90

Delete nim 66:

	nim [PK] integer	name character varying (64)	department_id integer	score integer
1	101	Ahmad Rahman	1	85
2	102	Siti Aisyah	2	90
3	103	Rizky Hidayat	1	88
4	104	Dian Permata	3	92
5	105	Fauzan Maulana	4	87
6	2	Jane Doe	1	85
7	3	Jane Does	1	87
8	55	John Doe	1	90

bi\_dev:

Query	Query History
1	-- Uji INSERT (akan gagal)
2	INSERT INTO "salam".mahasiswa (nim, name, department_id, score) VALUES (3, 'Jane Does', 1, 87);
3	
4	-- Uji SELECT
5	SELECT * FROM "salam".mahasiswa;

Insert -> gagal:

Query	Query History
1	-- Uji INSERT (akan gagal)
2	INSERT INTO "salam".mahasiswa (nim, name, department_id, score) VALUES (3, 'Jane Does', 1, 87);
3	
4	-- Uji SELECT
5	SELECT * FROM "salam".mahasiswa;

Data Output	Messages	Notifications
ERROR: permission denied for table mahasiswa		
SQL state: 42501		

View -> berhasil:

```
1 -- Uji INSERT (akan gagal)
2 INSERT INTO "salam".mahasiswa (nim, name, department_id, score) VALUES (3, 'Jane Does', 1, 87);
3
4 -- Uji SELECT
5 SELECT * FROM "salam".mahasiswa;
```

Data Output

Messages

Notifications

SQL

	nim [PK] integer	name character varying (64)	department_id integer	score integer
1	101	Ahmad Rahman	1	85
2	102	Siti Aisyah	2	90
3	103	Rizky Hidayat	1	88
4	104	Dian Permata	3	92

## Data\_engineer:

```
Query Query History
1 -- Uji INSERT
2 INSERT INTO "salam".mahasiswa (nim, name, department_id, score) VALUES (33, 'Alice', 1, 88);
3
4 -- Uji SELECT
5 SELECT * FROM "salam".mahasiswa;
6
7 -- Uji UPDATE
8 UPDATE "salam".mahasiswa SET score = 95 WHERE nim = 3;
9
10 -- Uji DELETE
11 DELETE FROM "salam".mahasiswa WHERE nim = 3;
12
13 -- Uji CREATE TABLE (akan berhasil)
14 CREATE TABLE "salam".new_table (
15     id SERIAL PRIMARY KEY,
16     description VARCHAR(255)
17 );
18
19 -- Uji DROP TABLE (akan berhasil)
20 DROP TABLE "salam".new_table;
```

## Insert:

```
Query Query History
1 -- Uji INSERT
2 INSERT INTO "salam".mahasiswa (nim, name, department_id, score) VALUES (33, 'Alice', 1, 88);
3
4 -- Uji SELECT
5 SELECT * FROM "salam".mahasiswa;
6
7 -- Uji UPDATE
8 UPDATE "salam".mahasiswa SET score = 95 WHERE nim = 3;
9
10 -- Uji DELETE
```

Data Output Messages Notifications

	nim [PK] integer	name character varying (64)	department_id integer	score integer
1	101	Ahmad Rahman	1	85
2	102	Siti Aisyah	2	90
3	103	Rizky Hidayat	1	88
4	104	Dian Permata	3	92
5	105	Fauzan Maulana	4	87
6	2	Jane Doe	1	85
7	3	Jane Does	1	87
8	55	John Doe	1	90
9	33	Alice	1	88

## Update nim 33:

```
7 -- Uji UPDATE
8 UPDATE "salam".mahasiswa SET name = 'ecila' WHERE nim = 33;
```

Data Output Messages Notifications

	nim [PK] integer	name character varying (64)	department_id integer	score integer
	101	Ahmad Rahman	1	85
:	102	Siti Aisyah	2	90
:	103	Rizky Hidayat	1	88
:	104	Dian Permata	3	92
:	105	Fauzan Maulana	4	87
:	2	Jane Doe	1	85
:	3	Jane Does	1	87
:	55	John Doe	1	90
:	33	ecila	1	88

## Delete nim 33:

```

9
10 -- Uji DELETE
11 DELETE FROM "salam".mahasiswa WHERE nim = 33;
12

```

Data Output Messages Notifications

	nim [PK] integer	name character varying (64)	department_id integer	score integer
1	101	Ahmad Rahman	1	85
2	102	Siti Aisyah	2	90
3	103	Rizky Hidayat	1	88
4	104	Dian Permata	3	92
5	105	Fauzan Maulana	4	87
6	2	Jane Doe	1	85
7	3	Jane Does	1	87
8	55	John Doe	1	90

Create table:

```

13 -- Uji CREATE TABLE (akan berhasil)
14 CREATE TABLE "salam".details (
15     id SERIAL PRIMARY KEY,
16     description VARCHAR(255)
17 );
18

```

Data Output Messages Notifications

CREATE TABLE

Query returned successfully in 78 msec.

Drop table:

```

19 -- Uji DROP TABLE (akan berhasil)
20 DROP TABLE "salam".details;

```

Data Output Messages Notifications

DROP TABLE

Query returned successfully in 77 msec.

```

22 SELECT * FROM "salam".details;

```

Data Output Messages Notifications

ERROR: relation "salam.details" does not exist  
 LINE 1: SELECT \* FROM "salam".details;  
 ^

SQL state: 42P01

Character: 15