

Dokumentasi Ujian Tengah Semester 5
Implementasi Manajemen Basis Data dengan PostgreSQL dan
PGADMIN4

Nama : Dimas Rizqia Hidayat	Dosen Pengampu : Wildan Budiawan Z,S.T.,M.Kom
NIM : 1237050073	Kelas : 5/A (Ganjil)
Prodi : Teknik Informatika	Tahun Akademik : 2025/2026
Matakuliah : Manajemen Basis Data	Tanggal Ujian : 23 Oktober 2025 (tanggal pengumpulan)

Daftar Isi

Dokumentasi Ujian Tengah Semester 5 Implementasi Manajemen Basis Data dengan PostgreSQL dan PGADMIN4	1
Daftar Isi	2
A. Set Up Projek	3
1. Instalasi Docker	3
2. Instalasi dan Konfigurasi PostgreSQL (Pgadmin dan DBeaver)	3
a. Instalasi PostgreSQL via Docker + connect ke pgadmin	3
b. Konfigurasi Akses PostgreSQL ke DBeaver (Eksternal)	6
B. Buat Skema SALAM, Table Mahasiswa dan Constraint Test	7
a. Membuat skema SALAM	7
b. Membuat <i>Table</i> mahasiswa	7
c. Memasukan data pada <i>Table</i> mahasiswa	7
d. Constraint Test pada Table mahasiswa	9
e. Full Script (01_buat_skema.sql)	13
C. Membuat User/Role (backend_dev, bi_dev, dan data_engineer), Konfigurasi Akses dan Tes Akses.....	15
a. Membuat User/Role dan Memberikan Akses umum ke Skema.....	15
b. Memberikan Akses CRUD kepada backend_dev	15
c. Memberikan Akses Select (Read-Only) kepada bi_dev.....	15
d. Memberikan Akses Full kepada data_engineer.....	16
e. Tes Akses masing masing User	16
f. Full Script 02_buat_user.sql.....	20
g. Full Script 02.1_backend_dev_test.sql	21
h. Full Script 02.2_bi_dev_test.sql.....	22
i. Full Script 02.3_data_engineer_test.sql	22
D. Fungsi Query Tambahan + Tes Fungsi.....	23
a. Fungsi cek status mahasiswa.....	23
b. Fungsi Hitung mahasiswa sesuai prodi	24
c. Fungsi Kategori IPK	25
d. Fungsi Pencarian sederhana (Nim atau nama)	26
e. Tes 2 : Sample data mahasiswa dengan fungsi Ambil 5 mahasiswa aktif dengan IPK tertinggi beserta kategorinya.....	28
f. Full Script 03_buat_fungsi.sql	28
g. Full Script 03.1_quick_test_fungsi.sql.....	30

A. Set Up Projek

1. Instalasi Docker

Langkah-langkah Instalasi Docker;

- a. Download Installer Docker Desktop dari situs resmi :
<https://www.docker.com/products/docker-desktop/>
- b. Install Docker Desktop menggunakan docker installer, Lalu Restart PC/Laptop setelah proses instalasi selesai.
- c. Verifikasi Instalasi
 - Buka Docker Desktop > Terminal
 - Jalankan perintah berikut :

```
docker --version  
docker-compose --version
```

- Output seharusnya sebagai berikut:

```
PS C:\Users\MyBook Hype> docker --version  
>> docker-compose --version  
>>  
Docker version 28.4.0, build d8eb465  
Docker Compose version v2.39.4-desktop.1
```

2. Instalasi dan Konfigurasi PostgreSQL (Pgadmin dan DBeaver)

- a. Instalasi PostgreSQL via Docker + connect ke pgadmin

- Pertama, buat struktur direktori projek

```
mkdir uts-postgre-pgadmin4  
cd uts-postgre-pgadmin4
```

- Kemudian, Buat file docker-compose.yml seperti berikut:

```
docker-compose.yml  
version: '3.8'  
  
services:  
  postgres_dimasrizqiahidayat:  
    image: postgres:15  
    container_name: postgres_dimasrizqiahidayat  
    restart: always  
    env_file:  
      - .env  
    ports:  
      - "22073:5432"  
    volumes:  
      - postgres_data:/var/lib/postgresql/data  
    networks:  
      - exam-network  
  
  pgadmin_dimasrizqiahidayat:  
    image: dpage/pgadmin4  
    container_name: pgadmin_dimasrizqiahidayat
```

```

    restart: always
    env_file:
      - .env
    ports:
      - "44073:80"
    depends_on:
      - postgres_dimasrizqiahidayat
    volumes:
      - pgadmin_data:/var/lib/pgadmin
    networks:
      - exam-network

  volumes:
    postgres_data:
    pgadmin_data:

  networks:
    exam-network:
      driver: bridge

```

- Buat file .env seperti berikut:

```

.env
POSTGRES_USER=postgres
POSTGRES_PASSWORD=ifunggul
POSTGRES_DB=mahasiswa_db
PGADMIN_DEFAULT_EMAIL=admin@admin.com
PGADMIN_DEFAULT_PASSWORD=admin123

```

- Jalankan docker-compose.yml
di Terminal jalankan command berikut;

```
docker-compose up -d
```

```

PS E:\.01-MyDocument\.01-Kuliah\Kuliah-sm5\Management Basis Data\uts-postgre-pgadmin> docker-compose up -d
time="2025-10-23T11:31:33+07:00" level=warning msg="E:\.01-MyDocument\.01-Kuliah\Kuliah-sm5\Management B
-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential c
[+] Running 3/3
✓ Network uts-postgre-pgadmin_exam-network  Created
✓ Container postgres_dimasrizqiahidayat  Started
✓ Container pgadmin_dimasrizqiahidayat  Started
PS E:\.01-MyDocument\.01-Kuliah\Kuliah-sm5\Management Basis Data\uts-postgre-pgadmin>

```

Lalu cek apakah Container sudah di jalankan

```
docker ps
```

Harusnya muncul seperti ini

```

PS E:\.01-MyDocument\.01-Kuliah\Kuliah-sm5\Management Basis Data\uts-postgre-pgadmin> docker ps
CONTAINER ID   IMAGE          COMMAND       CREATED      STATUS      PORTS          NAMES
95cc9bd7b170   dpage/pgadmin4   "/entrypoint.sh"   31 minutes ago   Up 31 minutes   0.0.0.0:44073->80/tcp, [::]:44073->80/tcp   pgadmin_
dimasrizqiahidayat
bcb41729fe10   postgres:15     "docker-entrypoint.s..."  31 minutes ago   Up 31 minutes   0.0.0.0:22073->5432/tcp, [::]:22073->5432/tcp   postgres
dimasrizqiahidayat
PS E:\.01-MyDocument\.01-Kuliah\Kuliah-sm5\Management Basis Data\uts-postgre-pgadmin>

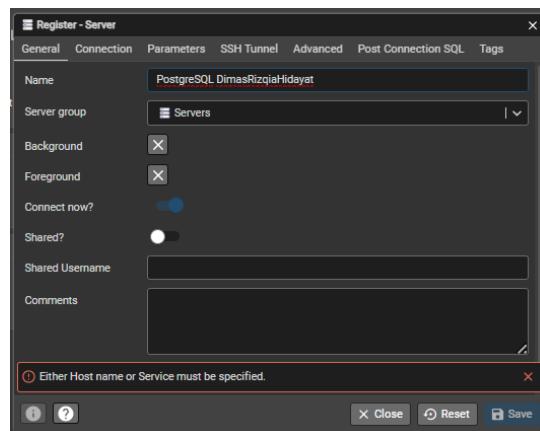
```

- Akses pgadmin via browser, lalu buka -> <http://localhost:44073>
- Lalu login dengan akun admin
Email : admin@admin.com

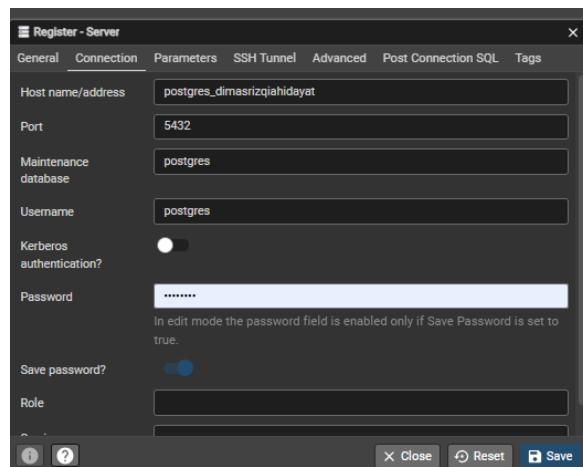
Password : admin123



- Hubungkan ke PostgreSQL (Container Internal)
 - Di sidebar kiri pgAdmin/Quick link Add new Server → Register → Server
 - Tab General:

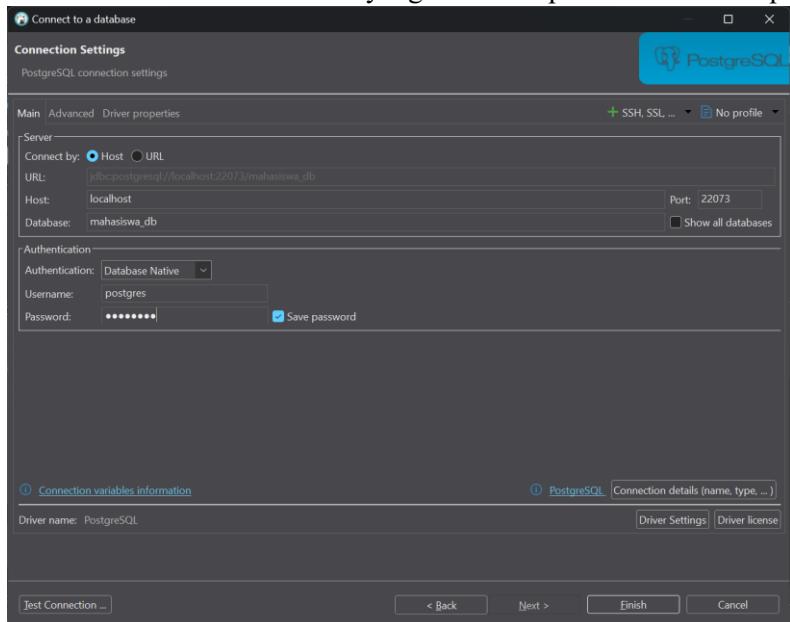


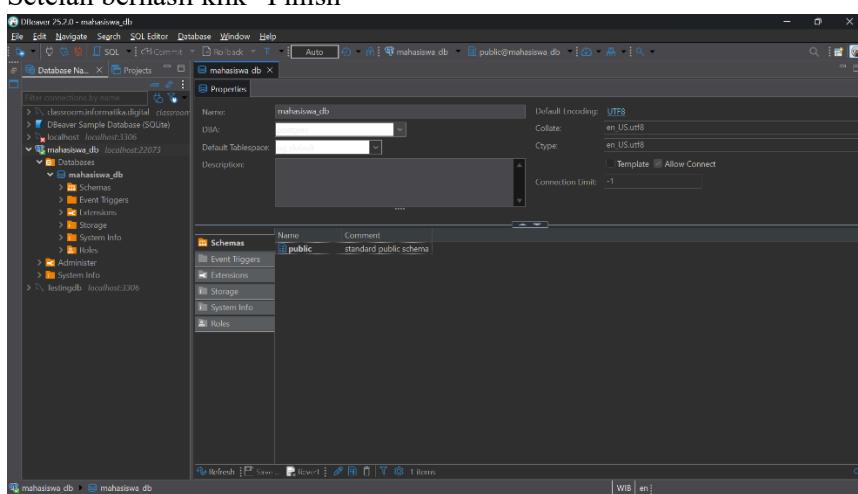
- Name: PostgreSQL DimasRizqiaHidayat
- Tab Connection:



- Host name/address: postgres_dimasrizqiahidayat
(ini sesuai dengan nama service di docker-compose.yml)
- Port: 5432
- Username: postgres

- Password: ifunggul
 - Klik Save

- b. Konfigurasi Akses PostgreSQL ke DBeaver (Eksternal)
 - Pastikan Container berjalan. Jalankan `docker ps` di Terminal dan pastikan `postgres_dimasrizqiahidayat` (port `22073:5432`) berjalan.
Jika belum, jalankan `docker-compose up -d`
 - Buka DBeaver, Lalu;
 - Klik ikon “New Database Connection” (ikon colokan listrik )
 - Pilih PostgreSQL
 - Klik Next
 - Isi detail koneksi sesuai yang tertera pada `docker-compose.yml` dan `.env`

 - Host : localhost
 - Port : 22073
 - Database : mahasiswa_db
 - Username : postgres
 - Password : ifunggul
 - Centang “Save password locally”, Lalu klik “Test Connection”
 - Setelah berhasil klik “Finish”



B. Buat Skema SALAM, Table Mahasiswa dan Constraint Test

Untuk tahap ini kita akan menggunakan Query Tools di pgadmin

- Membuat skema SALAM

```
CREATE SCHEMA SALAM;
```

- Membuat *Table* mahasiswa

```
CREATE TABLE SALAM.mahasiswa (
    nim CHAR(10) PRIMARY KEY,
    nama VARCHAR(100) NOT NULL,
    prodi VARCHAR(100) NOT NULL,
    fakultas VARCHAR(100) NOT NULL,
    semester INT CHECK (semester BETWEEN 1 AND 14),
    angkatan INT CHECK (angkatan BETWEEN 2000 AND EXTRACT(YEAR FROM CURRENT_DATE)),
    email VARCHAR(100) UNIQUE,
    usia INT CHECK (usia >= 16),
    jenis_kelamin CHAR(1) CHECK (jenis_kelamin IN ('L', 'P')),
    ipk NUMERIC(3,2) CHECK (ipk BETWEEN 0.00 AND 4.00) DEFAULT NULL,
    status_aktif VARCHAR(20) CHECK (status_aktif IN ('Aktif', 'Cuti',
'Lulus', 'Drop Out')),

    -- Jika IPK masih NULL (mahasiswa baru), maka semester HARUS 1
    CONSTRAINT check_semester_mahasiswa_baru CHECK (
        (ipk IS NULL AND semester = 1) OR (ipk IS NOT NULL)
    )
);
```

- Memasukan data pada *Table* mahasiswa

```
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia,
jenis_kelamin, ipk, status_aktif)
VALUES
('1237150001', 'Ahmad Taufik', 'Informatika', 'Sains dan
Teknologi', 5, 2023, 'ahmad112@mail.com', 20, 'L', 3.45, 'Aktif');
```

Atau bisa memasukan data dummy yang sudah saya siapkan pada file **01.1_insert_dummy.sql**

```
-- 40 data mahasiswa (semester disesuaikan dengan angkatan)

INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk, status_aktif)
VALUES
-- Informatika (70-50)
('1227050001', 'Ahmad Fauzi', 'Informatika', 'Sains & Teknologi', 7, 2022, '1227050001@student.uinsgd.ac.id', 22, 'L', 3.45, 'Aktif'),
('1227050002', 'Siti Rahma', 'Informatika', 'Sains & Teknologi', 7, 2022, '1227050002@student.uinsgd.ac.id', 21, 'P', 3.12, 'Aktif'),
```

```

('1237050003','Rafi Ramadhan','Informatika','Sains & Teknologi',5,2023,'1237050003@student.uinsgd.ac.id',20,'L',3.15,'Aktif'),
('1237050004','Dewi Lestari','Informatika','Sains & Teknologi',5,2023,'1237050004@student.uinsgd.ac.id',22,'P',3.88,'Aktif'),
('1247050005','Fahri Hidayat','Informatika','Sains & Teknologi',3,2024,'1247050005@student.uinsgd.ac.id',19,'L',3.20,'Aktif'),
('1247050006','Nadia Aulia','Informatika','Sains & Teknologi',3,2024,'1247050006@student.uinsgd.ac.id',18,'P',3.55,'Aktif'),
('1257050007','Bima Prasetyo','Informatika','Sains & Teknologi',1,2025,'1257050007@student.uinsgd.ac.id',18,'L',NULL,'Aktif'),
('1257050008','Lina Marlina','Informatika','Sains & Teknologi',1,2025,'1257050008@student.uinsgd.ac.id',18,'P',NULL,'Aktif'),
-- Biologi (70-51)

('1227051001','Rina Amalia','Biologi','Sains & Teknologi',7,2022,'1227051001@student.uinsgd.ac.id',22,'P',3.55,'Aktif'),
('1227051002','Bayu Firmansyah','Biologi','Sains & Teknologi',7,2022,'1227051002@student.uinsgd.ac.id',21,'L',3.10,'Aktif'),
('1237051003','Nia Rosita','Biologi','Sains & Teknologi',5,2023,'1237051003@student.uinsgd.ac.id',21,'P',3.78,'Aktif'),
('1237051004','Andi Setiawan','Biologi','Sains & Teknologi',5,2023,'1237051004@student.uinsgd.ac.id',22,'L',3.33,'Aktif'),
('1247051005','Citra Permata','Biologi','Sains & Teknologi',3,2024,'1247051005@student.uinsgd.ac.id',18,'P',2.90,'Aktif'),
('1247051006','Rizky Maulana','Biologi','Sains & Teknologi',3,2024,'1247051006@student.uinsgd.ac.id',19,'L',3.0,'Aktif'),
('1257051007','Ayu Wulandari','Biologi','Sains & Teknologi',1,2025,'1257051007@student.uinsgd.ac.id',18,'P',NULL,'Aktif'),
('1257051008','Teguh Ramadhan','Biologi','Sains & Teknologi',1,2025,'1257051008@student.uinsgd.ac.id',18,'L',NULL,'Aktif'),
-- Aqidah & Filsafat Islam (71-52)

('1227152001','Hasan Basri','Aqidah & Filsafat Islam','Ushuluddin',7,2022,'1227152001@student.uinsgd.ac.id',23,'L',3.60,'Aktif'),
('1227152002','Nurul Hikmah','Aqidah & Filsafat Islam','Ushuluddin',7,2022,'1227152002@student.uinsgd.ac.id',21,'P',3.41,'Aktif'),
('1237152003','Ridwan Kamil','Aqidah & Filsafat Islam','Ushuluddin',5,2023,'1237152003@student.uinsgd.ac.id',20,'L',3.10,'Aktif'),
('1237152004','Maya Salsabila','Aqidah & Filsafat Islam','Ushuluddin',5,2023,'1237152004@student.uinsgd.ac.id',20,'P',3.25,'Aktif'),
('1247152005','Hana Sari','Aqidah & Filsafat Islam','Ushuluddin',3,2024,'1247152005@student.uinsgd.ac.id',18,'P',3.35,'Aktif'),
('1247152006','Faris Rahman','Aqidah & Filsafat Islam','Ushuluddin',3,2024,'1247152006@student.uinsgd.ac.id',18,'L',3.20,'Aktif'),
('1257152007','Naila Qolbi','Aqidah & Filsafat Islam','Ushuluddin',1,2025,'1257152007@student.uinsgd.ac.id',18,'P',NULL,'Aktif'),
('1257152008','Zain Fadhill','Aqidah & Filsafat Islam','Ushuluddin',1,2025,'1257152008@student.uinsgd.ac.id',18,'L',NULL,'Aktif'),
-- Ekonomi Syariah (72-53)

('1227253001','Eka Nurhaliza','Ekonomi Syariah','Ekonomi & Bisnis Islam',7,2022,'1227253001@student.uinsgd.ac.id',23,'P',3.85,'Aktif'),
('1227253002','Rangga Aditya','Ekonomi Syariah','Ekonomi & Bisnis Islam',7,2022,'1227253002@student.uinsgd.ac.id',21,'L',3.15,'Aktif'),
('1237253003','Dini Larasati','Ekonomi Syariah','Ekonomi & Bisnis Islam',5,2023,'1237253003@student.uinsgd.ac.id',21,'P',3.75,'Aktif'),
('1237253004','Taufik Hidayat','Ekonomi Syariah','Ekonomi & Bisnis Islam',5,2023,'1237253004@student.uinsgd.ac.id',20,'L',3.40,'Aktif'),
('1247253005','Sarah Fadilah','Ekonomi Syariah','Ekonomi & Bisnis Islam',3,2024,'1247253005@student.uinsgd.ac.id',18,'P',3.9,'Aktif'),
('1247253006','Reza Maulana','Ekonomi Syariah','Ekonomi & Bisnis Islam',3,2024,'1247253006@student.uinsgd.ac.id',19,'L',3.75,'Aktif'),
('1257253007','Intan Rahayu','Ekonomi Syariah','Ekonomi & Bisnis Islam',1,2025,'1257253007@student.uinsgd.ac.id',18,'P',NULL,'Aktif'),
('1257253008','Arif Wicaksono','Ekonomi Syariah','Ekonomi & Bisnis Islam',1,2025,'1257253008@student.uinsgd.ac.id',18,'L',NULL,'Aktif'),
-- Komunikasi & Penyiaran Islam (73-54)

('1227354001','Dita Amelia','Komunikasi & Penyiaran Islam','Dakwah & Komunikasi',7,2022,'1227354001@student.uinsgd.ac.id',23,'P',3.50,'Aktif'),
('1227354002','Fajar Nugraha','Komunikasi & Penyiaran Islam','Dakwah & Komunikasi',7,2022,'1227354002@student.uinsgd.ac.id',21,'L',3.20,'Aktif'),
('1237354003','Silvia Rahmadani','Komunikasi & Penyiaran Islam','Dakwah & Komunikasi',5,2023,'1237354003@student.uinsgd.ac.id',20,'P',3.33,'Aktif'),
('1237354004','Hafidz Fadlan','Komunikasi & Penyiaran Islam','Dakwah & Komunikasi',5,2023,'1237354004@student.uinsgd.ac.id',20,'L',3.12,'Aktif'),
('1247354005','Aisyah Putri','Komunikasi & Penyiaran Islam','Dakwah & Komunikasi',3,2024,'1247354005@student.uinsgd.ac.id',18,'P',3.3,'Aktif'),
('1247354006','Yoga Saputra','Komunikasi & Penyiaran Islam','Dakwah & Komunikasi',3,2024,'1247354006@student.uinsgd.ac.id',19,'L',3.55,'Aktif'),
('1257354007','Luthfi Azhari','Komunikasi & Penyiaran Islam','Dakwah & Komunikasi',1,2025,'1257354007@student.uinsgd.ac.id',18,'L',NULL,'Aktif'),

```

```
('1257354008', 'Nadia Khairunnisa', 'Komunikasi & Penyiaran Islam', 'Dakwah &  
Komunikasi', 1, 2025, '1257354008@student.uinsgd.ac.id', 18, 'P', NULL, 'Aktif');
```

d. Constraint Test pada Table mahasiswa

- Uji Primary Key

```
-- Uji Primary Key  
-- mencoba insert nim yang sudah ada  
INSERT INTO SALAM.mahasiswa  
(nim, nama, prodi, fakultas, semester, angkatan, email, usia,  
jenis_kelamin, ipk, status_aktif)  
VALUES  
('1237150001', 'Tes Duplikat', 'Informatika', 'Sains &  
Teknologi', 5, 2023, 'tes@mail.com', 20, 'L', 3.0, 'Aktif');
```

Hasil :

```
20/10/2025 10:59:48  
Date Rows affected Duration  
Copy Copy to Query Editor  
  
-- Uji Primary Key  
-- mencoba insert nim yang sudah ada  
INSERT INTO SALAM.mahasiswa  
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin  
VALUES  
('1237050001', 'Tes Duplikat', 'Informatika', 'Sains & Teknologi', 5, 2023,  
  
Messages  
ERROR: duplicate key value violates unique constraint "mahasiswa_pkey"  
Key (nim)=(1237050001) already exists.  
  
SQL state: 23505  
Detail: Key (nim)=(1237050001) already exists.
```

- Uji Unique Key

```
-- Uji Unique key  
-- mencoba pakai email yang sudah ada  
INSERT INTO SALAM.mahasiswa  
(nim, nama, prodi, fakultas, semester, angkatan, email, usia,  
jenis_kelamin, ipk, status_aktif)  
VALUES  
('1257050999', 'Email Duplikat', 'Informatika', 'Sains &  
Teknologi', 1, 2025, 'ahmad112@mail.com', 18, 'L', NULL,  
'Aktif');
```

Hasil:

```

20/10/2025 11.04.16           -592 msec
Date                  Rows affected      Duration
Copy | Copy to Query Editor

-- Uji Unique key
-- mencoba pakai email yang sudah ada
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin
VALUES
('1257050999', 'Email Duplikat', 'Informatika', 'Sains & Teknologi', 1, 202

Messages
ERROR: duplicate key value violates unique constraint "mahasiswa_email_key"
Key (email)=(ahmad@mail.com) already exists.

SQL state: 23505
Detail: Key (email)=(ahmad@mail.com) already exists.

```

- Uji CHECK usia kurang dari 16 tahun

```

-- Uji CHECK Usia
-- usia terlalu muda (usia < 16 tahun)
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia,
jenis_kelamin, ipk, status_aktif)
VALUES
('1257050998', 'Usia Tidak Valid', 'Informatika', 'Sains &
Teknologi', 1, 2025, 'invalid_usia@student.uinsgd.ac.id', 15,
'L', NULL, 'Aktif');

```

Hasil :

```

20/10/2025 11.06.15           -526 msec
Date                  Rows affected      Duration
Copy | Copy to Query Editor

-- Uji Usia
-- usia terlalu muda (usia < 16 tahun)
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin
VALUES
('1257050998', 'Usia Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025,
invalid_usia@student.uinsgd.ac.id, 15, L, null, Aktif)

Messages
ERROR: new row for relation "mahasiswa" violates check constraint "mahasiswa_usia_check"
Failing row contains (1257050998, Usia Tidak Valid, Informatika, Sains & Teknologi, 1, 2025,
invalid_usia@student.uinsgd.ac.id, 15, L, null, Aktif).

SQL state: 23514
Detail: Failing row contains (1257050998, Usia Tidak Valid, Informatika, Sains & Teknologi, 1, 2025,
invalid_usia@student.uinsgd.ac.id, 15, L, null, Aktif).

```

- Uji CHECK ipk diluar batas

```

-- Uji CHECK ipk
-- ipk di luar batas
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia,
jenis_kelamin, ipk, status_aktif)
VALUES
('1257050997', 'IPK Tidak Valid', 'Informatika', 'Sains &
Teknologi', 1, 2025, 'invalid_ipk@student.uinsgd.ac.id', 18,
'L', 4.5, 'Aktif');

```

Hasil :

```

20/10/2025 11.08.14           -265 msec
Date                          Rows affected          Duration
Copy  Copy to Query Editor
-- Uji CHECK ipk
-- ipk di luar batas
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk, status_aktif)
VALUES
('1257050997', 'IPK Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025, 'invalid_ipk@student.uinsgd.ac.id', 18, L, 4.50, Aktif);

Messages
ERROR: new row for relation "mahasiswa" violates check constraint "mahasiswa_ipk_check"
Failing row contains (1257050997, IPK Tidak Valid, Informatika, Sains & Teknologi, 1, 2025, invalid_ipk@student.uinsgd.ac.id, 18, L, 4.50, Aktif).

SQL state: 23514
Detail: Failing row contains (1257050997, IPK Tidak Valid, Informatika, Sains & Teknologi, 1, 2025, invalid_ipk@student.uinsgd.ac.id, 18, L, 4.50, Aktif).

```

- Uji CHECK semester diluar batas

```

-- Uji CHECK semester
-- semester di luar batas
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk, status_aktif)
VALUES
('1257050996', 'Semester Tidak Valid', 'Informatika', 'Sains & Teknologi', 20, 2025, 'invalid_semester@student.uinsgd.ac.id', 18, 'L', NULL, 'Aktif');

```

Hasil :

```

20/10/2025 11.12.42           -335 msec
Date                          Rows affected          Duration
Copy  Copy to Query Editor
-- Uji CHECK semester
-- semester di luar batas
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk, status_aktif)
VALUES
('1257050996', 'Semester Tidak Valid', 'Informatika', 'Sains & Teknologi', 20, 2025, 'invalid_semester@student.uinsgd.ac.id', 18, 'L', NULL, 'Aktif');

Messages
ERROR: new row for relation "mahasiswa" violates check constraint "check_semester_mahasiswa_baru"
Failing row contains (1257050996, Semester Tidak Valid, Informatika, Sains & Teknologi, 20, 2025, invalid_semester@student.uinsgd.ac.id, 18, 'L', NULL, 'Aktif').

SQL state: 23514
Detail: Failing row contains (1257050996, Semester Tidak Valid, Informatika, Sains & Teknologi, 20, 2025, invalid_semester@student.uinsgd.ac.id, 18, 'L', NULL, 'Aktif').

```

- Uji CHECK jenis kelamin (L/P)

```

-- Uji CHECK jenis kelamin (L/P)
-- jenis kelamin tidak valid
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk, status_aktif)
VALUES
('1257050995', 'Gender Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025, 'invalid_gender@student.uinsgd.ac.id', 18, 'X', NULL, 'Aktif');

```

Hasil :

```

20/10/2025 11.14.17
Date          Rows affected      -944 msec
Duration
Copy | Copy to Query Editor

-- Uji CHECK jenis kelamin (L/P)
-- jenis kelamin tidak valid
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk, status_aktif)
VALUES
('1257858995', 'Gender Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025, 'invalid_gender@student.uinsgd.ac.id', 18, X, null, Aktif)

Messages
ERROR: new row for relation 'mahasiswa' violates check constraint 'mahasiswa_jenis_kelamin_check'
Failing row contains (1257858995, Gender Tidak Valid, Informatika, Sains & Teknologi, 1, 2025, invalid_gender@student.uinsgd.ac.id, 18, X, null, Aktif).

SQL state: 23514
Detail: Failing row contains (1257858995, Gender Tidak Valid, Informatika, Sains & Teknologi, 1, 2025, invalid_gender@student.uinsgd.ac.id, 18, X, null, Aktif).

```

- Uji CHECK status aktif

```

-- Uji CHECK status aktif
-- status aktif tidak sesuai
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia,
jenis_kelamin, ipk, status_aktif)
VALUES
('1257050994', 'Status Tidak Valid', 'Informatika', 'Sains &
Teknologi', 1, 2025, 'invalid_status@student.uinsgd.ac.id', 18,
'L', NULL, 'Keluar');

```

Hasil :

```

20/10/2025 11.15.45
Date          Rows affected      -66 msec
Duration
Copy | Copy to Query Editor

-- Uji CHECK status aktif
-- status aktif tidak sesuai
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk, status_aktif)
VALUES
('1257050994', 'Status Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025, 'invalid_status@student.uinsgd.ac.id', 18, L, null, Keluar)

Messages
ERROR: new row for relation 'mahasiswa' violates check constraint 'mahasiswa_status_aktif_check'
Failing row contains (1257050994, Status Tidak Valid, Informatika, Sains & Teknologi, 1, 2025, invalid_status@student.uinsgd.ac.id, 18, L, null, Keluar).

SQL state: 23514
Detail: Failing row contains (1257050994, Status Tidak Valid, Informatika, Sains & Teknologi, 1, 2025, invalid_status@student.uinsgd.ac.id, 18, L, null, Keluar).

```

- Uji CHECK NULL Value

```

-- Uji CHECK NULL
-- nama kosong
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia,
jenis_kelamin, ipk, status_aktif)
VALUES
('1257050993', NULL, 'Informatika', 'Sains & Teknologi', 1,
2025, 'invalid_null@student.uinsgd.ac.id', 18, 'L', NULL,
'Aktif');

```

Hasil :

20/10/2025 11.15.45		-66 msec
Date	Rows affected	Duration
Copy Copy to Query Editor		
<pre>-- Uji CHECK status aktif -- status aktif tidak sesuai INSERT INTO SALAM.mahasiswa (nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk, status_aktif) VALUES ('1257050994', 'Status Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025, 'invalid_status@student.uinsgd.ac.id', 18, 'L', null, 'Keluar'); Messages ERROR: new row for relation "mahasiswa" violates check constraint "mahasiswa_status_aktif_ck" Failing row contains (1257050994, Status Tidak Valid, Informatika, Sains & Teknologi, 1, 2025, invalid_status@student.uinsgd.ac.id, 18, L, null, Keluar). SQL state: 23514 Detail: Failing row contains (1257050994, Status Tidak Valid, Informatika, Sains & Teknologi, 1, 2025, invalid_status@student.uinsgd.ac.id, 18, L, null, Keluar).</pre>		

e. Full Script (01_buat_skema.sql)

```
-- Script untuk membuat skema dan tabel mahasiswa dengan berbagai constraint
CREATE SCHEMA SALAM;

GO

CREATE TABLE SALAM.mahasiswa (
    nim CHAR(10) PRIMARY KEY,
    nama VARCHAR(100) NOT NULL,
    prodi VARCHAR(100) NOT NULL,
    fakultas VARCHAR(100) NOT NULL,
    semester INT CHECK (semester BETWEEN 1 AND 14),
    angkatan INT CHECK (angkatan BETWEEN 2000 AND EXTRACT(YEAR FROM CURRENT_DATE)),
    email VARCHAR(100) UNIQUE,
    usia INT CHECK (usia >= 16),
    jenis_kelamin CHAR(1) CHECK (jenis_kelamin IN ('L', 'P')),
    ipk NUMERIC(3,2) CHECK (ipk BETWEEN 0.00 AND 4.00) DEFAULT NULL,
    status_aktif VARCHAR(20) CHECK (status_aktif IN ('Aktif', 'Cuti', 'Lulus', 'Drop Out')),
    CONSTRAINT check_semester_mahasiswa_baru CHECK (
        (ipk IS NULL AND semester = 1) OR (ipk IS NOT NULL)
    )
);

INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk, status_aktif)
VALUES
('1237150001', 'Ahmad Taufik', 'Informatika', 'Sains dan Teknologi', 5, 2023,
'ahmad112@mail.com', 20, 'L', 3.45, 'Aktif');

-- Uji Primary Key
-- mencoba insert nim yang sudah ada
INSERT INTO SALAM.mahasiswa
```

```
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk,
status_aktif)
VALUES
('1237050001', 'Tes Duplikat', 'Informatika', 'Sains & Teknologi', 5, 2023,
'tes@mail.com', 20, 'L', 3.0, 'Aktif');

-- Uji Unique key
-- mencoba pakai email yang sudah ada
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk,
status_aktif)
VALUES
('1257050999', 'Email Duplikat', 'Informatika', 'Sains & Teknologi', 1, 2025,
'ahmad@mail.com', 18, 'L', NULL, 'Aktif');

-- Uji CHECK Usia
-- usia terlalu muda (usia < 16 tahun)
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk,
status_aktif)
VALUES
('1257050998', 'Usia Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025,
'invalid_usia@student.uinsgd.ac.id', 15, 'L', NULL, 'Aktif');

-- Uji CHECK ipk
-- ipk di luar batas
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk,
status_aktif)
VALUES
('1257050997', 'IPK Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025,
'invalid_ipk@student.uinsgd.ac.id', 18, 'L', 4.5, 'Aktif');

-- Uji CHECK semester
-- semester di luar batas
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk,
status_aktif)
VALUES
('1257050996', 'Semester Tidak Valid', 'Informatika', 'Sains & Teknologi', 20, 2025,
'invalid_semester@student.uinsgd.ac.id', 18, 'L', NULL, 'Aktif');

-- Uji CHECK jenis kelamin (L/P)
-- jenis kelamin tidak valid
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk,
status_aktif)
VALUES
```

```

('1257050995', 'Gender Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025,
'invalid_gender@student.uinsgd.ac.id', 18, 'X', NULL, 'Aktif');

-- Uji CHECK status aktif
-- status aktif tidak sesuai
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk,
status_aktif)
VALUES
('1257050994', 'Status Tidak Valid', 'Informatika', 'Sains & Teknologi', 1, 2025,
'invalid_status@student.uinsgd.ac.id', 18, 'L', NULL, 'Keluar');

-- Uji CHECK NULL
-- nama kosong
INSERT INTO SALAM.mahasiswa
(nim, nama, prodi, fakultas, semester, angkatan, email, usia, jenis_kelamin, ipk,
status_aktif)
VALUES
('1257050993', NULL, 'Informatika', 'Sains & Teknologi', 1, 2025,
'invalid_null@student.uinsgd.ac.id', 18, 'L', NULL, 'Aktif');

```

C. Membuat User/Role (backend_dev, bi_dev, dan data_engineer), Konfigurasi Akses dan Tes Akses

- Membuat User/Role dan Memberikan Akses umum ke Skema

```

-- Pastikan kamu login sebagai superuser (misal: postgres)
CREATE USER backend_dev WITH PASSWORD 'backend123';
CREATE USER bi_dev WITH PASSWORD 'bidev123';
CREATE USER data_engineer WITH PASSWORD 'dataeng123';

-- Berikan akses umum ke skema
GRANT USAGE ON SCHEMA SALAM TO backend_dev, bi_dev, data_engineer;

```

- Memberikan Akses CRUD kepada backend_dev

```

-- backend_dev access account
-- CRUD Access : dapat membaca dan memodifikasi data di semua
tabel
GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA SALAM
TO backend_dev;

```

- Memberikan Akses Select (Read-Only) kepada bi_dev

```

-- bi_dev access account
-- Hanya dapat membaca (SELECT)
GRANT SELECT ON ALL TABLES IN SCHEMA SALAM TO bi_dev;

-- Termasuk view
GRANT SELECT ON ALL SEQUENCES IN SCHEMA SALAM TO bi_dev;

```

```
ALTER DEFAULT PRIVILEGES IN SCHEMA SALAM
GRANT SELECT ON TABLES TO bi_dev;
```

- d. Memberikan Akses Full kepada data_engineer

```
-- data_engineer account (Full Access)
-- Dapat membuat, mengubah, dan menghapus objek dalam skema
GRANT CREATE, USAGE ON SCHEMA SALAM TO data_engineer;

-- CRUD ke seluruh tabel
GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA SALAM
TO data_engineer;

-- Hak tambahan
ALTER DEFAULT PRIVILEGES IN SCHEMA SALAM
GRANT ALL PRIVILEGES ON TABLES TO data_engineer;
```

- e. Tes Akses masing masing User

Jangan lupa untuk ubah ke role yang ingin di test!

```
SET ROLE backend_dev;
```

Atau

```
SET ROLE bi_dev;
```

Atau

```
SET ROLE data_engineer;
```

- CRUD Akses backend_dev
 - Uji SELECT

```
-- Uji SELECT
SELECT * FROM SALAM.mahasiswa;
```

Hasil :

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL code:

```
-- Uji SELECT
SELECT * FROM SALAM.mahasiswa;
```

The results pane displays the following data:

id	nama	prodi	fakultas	semester	angkatan	email	telepon	jenis_kelamin	lkk	status_akif
1	Ahmad Faizi	Informatika	Sains & Teknologi	7	2022	1227050001@student.unspg.ac...	22	L	3.45	Aktif
2	Siti Rahma	Informatika	Sains & Teknologi	7	2022	1227050002@student.unspg.ac...	21	P	3.12	Aktif
3	Rafi Ramadhan	Informatika	Sains & Teknologi	5	2023	1227050003@student.unspg.ac...	26	L	3.15	Aktif
4	Dewi Lestari	Informatika	Sains & Teknologi	5	2023	1227050004@student.unspg.ac...	22	P	2.68	Aktif
5	Fahrin Indrayani	Informatika	Sains & Teknologi	3	2024	1227050005@student.unspg.ac...	19	L	3.20	Aktif
6	Nadia Aulia	Informatika	Sains & Teknologi	3	2024	1227050006@student.unspg.ac...	18	P	3.55	Aktif
7	Rima Pratiyoga	Informatika	Sains & Teknologi	1	2025	1227050007@student.unspg.ac...	18	[redacted]	[redacted]	[redacted]
8	Liaqatul Huda	Informatika	Sains & Teknologi	1	2025	1227050008@student.unspg.ac...	18	P	[redacted]	[redacted]
9	Rico Amrizal	Biologi	Sains & Teknologi	7	2023	1227050009@student.unspg.ac...	23	P	3.65	Aktif
10	Baya Firmansyah	Biologi	Sains & Teknologi	7	2023	1227050010@student.unspg.ac...	21	L	3.10	Aktif
11	Nis Rosdiani	Biologi	Sains & Teknologi	5	2023	1227050011@student.unspg.ac...	21	P	3.78	Aktif
12	Andi Setyawan	Biologi	Sains & Teknologi	5	2023	1227050012@student.unspg.ac...	22	L	3.33	Aktif
13	Citra Permatasari	Biologi	Sains & Teknologi	3	2024	1227050013@student.unspg.ac...	18	P	2.90	Aktif
14	Riday Maulana	Biologi	Sains & Teknologi	3	2024	1227050014@student.unspg.ac...	19	L	3.00	Aktif
15	Ayu Walandari	Biologi	Sains & Teknologi	1	2025	1227050015@student.unspg.ac...	18	P	[redacted]	[redacted]
16	Tegar Ramadhan	Biologi	Sains & Teknologi	1	2025	1227050016@student.unspg.ac...	18	L	[redacted]	[redacted]
17	Hasan Besri	Aqiqah & Fitrah Islami	Ushrauddin	7	2022	1227152001@student.unspg.ac...	23	L	3.60	Aktif
18	Nurul Hirmah	Aqiqah & Fitrah Islami	Ushrauddin	7	2022	1227152002@student.unspg.ac...	21	P	3.41	Aktif
19	Ridwan Kamili	Aqiqah & Fitrah Islami	Ushrauddin	5	2023	1227152003@student.unspg.ac...	20	L	3.10	Aktif

- Uji INSERT

```
-- Uji INSERT
INSERT INTO
SALAM.mahasiswa(nim,nama,prodi,fakultas,semester,angkatan,email,usia,jenis_kelamin,ipk,status_aktif)
VALUES('1237050200','Test Backend.dev','Informatika','Sains dan
Teknologi',3,2025,'testbackend@mail.com',20,'L',3.5,'Aktif');
```

Hasil :

```
22/10/2025 15.45.59
Date                                     Rows affected          197 msec
Duration
Copy | Copy to Query Editor
-- Uji INSERT
INSERT INTO SALAM.mahasiswa(nim,nama,prodi,fakultas,semester,angkatan,email,usia,jenis_kelamin,ipk,status_aktif)
VALUES('1237050200','Test Backend.dev','Informatika','Sains dan Teknologi',3,2025,'testbackend@mail.com',20,'L',3.5,'Aktif');

Messages
Query returned successfully in 197 msec.
```

- Uji UPDATE

```
-- Uji UPDATE
UPDATE SALAM.mahasiswa SET ipk=3.6 WHERE
nim='1237050200';
```

Hasil :

```
22/10/2025 15.35.31
Date                                     Rows affected          124 msec
Duration
Copy | Copy to Query Editor
-- Uji UPDATE
UPDATE SALAM.mahasiswa SET ipk=3.6 WHERE nim='1237050200';

Messages
Query returned successfully in 124 msec.
```

- Uji DELETE

```
-- Uji DELETE
DELETE FROM SALAM.mahasiswa WHERE nim='1237050200';
```

Hasil :

```

22/10/2025 15.40.06
Date                                     Rows affected          Duration
Copy | Copy to Query Editor

-- Uji DELETE
DELETE FROM SALAM.mahasiswa WHERE nim='1237050200';

Messages
Query returned successfully in 125 msec.

```

- Uji Buat Table (Harus gagal)

```

-- Uji buat Table (Harus gagal)
CREATE TABLE SALAM.test_backend(id INT);

Hasil :
16
17 -- Uji buat Table (Harus gagal)
18 CREATE TABLE SALAM.test_backend(id INT);

Data Output Messages Notifications

ERROR: permission denied for schema salam
LINE 2: CREATE TABLE SALAM.test_backend(id INT);
^

SQL state: 42501
Character: 46

```

- Read-Only Akses bi_dev
 - Uji SELECT

```

-- Uji SELECT
SELECT * FROM SALAM.mahasiswa;

```

Hasil :

nim	name	prodi	kelulusan	semester	angkatan	email	seks	jenis_kelamin	blk	status_sks	character varying (25)
1227050001	Ahmad Faizi	Informatika	Sains & Teknologi	7	2822	1227050001@student.unsp.ac...	22	L	3.45	Akif	
1227050002	Siti Rahma	Informatika	Sains & Teknologi	7	2822	1227050002@student.unsp.ac...	21	P	3.12	Akif	
1227050003	Raf Ramadhan	Informatika	Sains & Teknologi	5	2823	1227050003@student.unsp.ac...	20	L	3.15	Akif	
1227050004	Dewi Lestari	Informatika	Sains & Teknologi	5	2823	1227050004@student.unsp.ac...	22	P	3.48	Akif	
1227050005	Fahril Hidayat	Informatika	Sains & Teknologi	3	2824	1227050005@student.unsp.ac...	19	L	3.20	Akif	
1227050006	Nadia Aulia	Informatika	Sains & Teknologi	3	2824	1227050006@student.unsp.ac...	18	P	3.55	Akif	
1227050007	Rima Prastyo	Informatika	Sains & Teknologi	1	2825	1227050007@student.unsp.ac...	18	L	[outof]	Akif	
1227050008	Lilia Mariana	Informatika	Sains & Teknologi	1	2825	1227050008@student.unsp.ac...	16	P	[outof]	Akif	
1227050009	Kino Analia	Bioleg	Sains & Teknologi	7	2822	1227050009@student.unsp.ac...	22	P	3.55	Akif	
1227050010	Baw Firmatayah	Bioleg	Sains & Teknologi	7	2822	1227050010@student.unsp.ac...	21	L	3.10	Akif	
1227050011	Nia Rizka	Bioleg	Sains & Teknologi	9	2823	1227050011@student.unsp.ac...	21	P	3.95	Akif	
1227050012	Aldi Purnamasari	Bioleg	Sains & Teknologi	5	2823	1227050012@student.unsp.ac...	22	L	3.25	Akif	
1227050013	Gita Permata	Bioleg	Sains & Teknologi	9	2824	1227050013@student.unsp.ac...	18	P	2.80	Akif	
1227050014	Ricky Mulyadi	Bioleg	Sains & Teknologi	3	2824	1227050014@student.unsp.ac...	19	L	3.00	Akif	
1227050015	Aky Waleedini	Bioleg	Sains & Teknologi	1	2825	1227050015@student.unsp.ac...	18	P	[outof]	Akif	
1227050016	Teguh Ramadhan	Bioleg	Sains & Teknologi	1	2825	1227050016@student.unsp.ac...	18	L	[outof]	Akif	
1227150001	Hesan Basit	Ajgah & Fisafat Islam	Ushuluddin	7	2822	1227150001@student.unsp.ac...	23	L	3.60	Akif	
1227150002	Nurul Hikmah	Ajgah & Fisafat Islam	Ushuluddin	7	2822	1227150002@student.unsp.ac...	21	P	3.41	Akif	
1227150003	Ridwan Kanili	Ajgah & Fisafat Islam	Ushuluddin	5	2823	1227150003@student.unsp.ac...	20	L	3.10	Akif	

- Uji INSERT (Harus gagal)

```
-- Uji INSERT (Harus gagal)
INSERT INTO
SALAM.mahasiswa(nim,nama,prodi,fakultas,semester,angkatan,email,usia,jenis_kelamin,ipk,status_aktif)
VALUES('1237050201','Test BI','Informatika','Sains dan Teknologi',3,2025,'testbi@mail.com',20,'P',3.7,'Aktif');
```

Hasil :

```
7  -- Uji INSERT (Harus gagal)
8  INSERT INTO SALAM.mahasiswa(nim,nama,prodi,fakultas,semester,angkatan,email,usia,jenis_kelamin,ipk,status_aktif)
9  VALUES('1237050201','Test BI','Informatika','Sains dan Teknologi',3,2025,'testbi@mail.com',20,'P',3.7,'Aktif');
Data Output Messages Notifications
ERROR: permission denied for table mahasiswa
```

- Full Akses data_engineer
 - Uji CREATE TABLE

```
-- Uji CREATE TABLE
CREATE TABLE SALAM.test_dataeng(id SERIAL PRIMARY KEY,
info TEXT);
```

Hasil :

Date	Rows affected	Duration
22/10/2025 16.09.26		202 msec
		Duration

```
-- Uji CREATE TABLE
CREATE TABLE SALAM.test_dataeng(id SERIAL PRIMARY KEY, info TEXT);
Messages
Query returned successfully in 202 msec.
```

- Uji INSERT

```
-- Uji INSERT
INSERT INTO SALAM.test_dataeng(info) VALUES('Percobaan Data Engineer');

-- Lihat Hasil
SELECT * FROM SALAM.test_dataeng;
```

Hasil :

id	info
1	Percobaan Data Engin...

- Uji UPDATE

```
-- Uji UPDATE
UPDATE SALAM.test_dataeng SET info='Update Berhasil'
WHERE id=1;
SELECT * FROM SALAM.test_dataeng;
```

Hasil :

The screenshot shows a PostgreSQL client interface. The SQL tab contains the executed query. The Data Output tab displays a table with one row:

	id [PK] integer	info text
1	1	Update Berhasil

- Uji DELETE

```
-- Uji DELETE
DELETE FROM SALAM.test_dataeng WHERE id=1;
```

Hasil :

The screenshot shows a PostgreSQL client interface. The SQL tab contains the executed query. The Messages tab shows a success message: "Query returned successfully in 186 msec." The table header includes Date, Rows affected, and Duration.

Date	Rows affected	Duration
22/10/2025 16.12.52	1	186 msec

Messages

Query returned successfully in 186 msec.

- Uji DROP TABLE

```
-- Uji DROP TABLE
DROP TABLE SALAM.test_dataeng;
```

Hasil :

The screenshot shows a PostgreSQL client interface. The SQL tab contains the executed query. The Messages tab shows a success message: "Query returned successfully in 105 msec." The table header includes Date, Rows affected, and Duration.

Date	Rows affected	Duration
22/10/2025 16.13.27	1	105 msec

Messages

Query returned successfully in 105 msec.

f. Full Script 02_buat_user.sql

```
-- Script new user etc --
-- Pastikan kamu login sebagai superuser (misal: postgres)
CREATE USER backend_dev WITH PASSWORD 'backend123';
CREATE USER bi_dev WITH PASSWORD 'bidev123';
```

```

CREATE USER data_engineer WITH PASSWORD 'dataeng123';

-- =====
-- Berikan akses umum ke skema
GRANT USAGE ON SCHEMA SALAM TO backend_dev, bi_dev, data_engineer;

-- backend_dev access account
-- CRUD Access : dapat membaca dan memodifikasi data di semua tabel
GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA SALAM TO backend_dev;

-- Agar hak akses ini otomatis berlaku untuk tabel baru
ALTER DEFAULT PRIVILEGES IN SCHEMA SALAM
GRANT SELECT, INSERT, UPDATE, DELETE ON TABLES TO backend_dev;

-- bi_dev access account
-- Hanya dapat membaca (SELECT)
GRANT SELECT ON ALL TABLES IN SCHEMA SALAM TO bi_dev;

-- Termasuk view
GRANT SELECT ON ALL SEQUENCES IN SCHEMA SALAM TO bi_dev;

ALTER DEFAULT PRIVILEGES IN SCHEMA SALAM
GRANT SELECT ON TABLES TO bi_dev;

-- data_engineer account (Full Access)
-- Dapat membuat, mengubah, dan menghapus objek dalam skema
GRANT CREATE, USAGE ON SCHEMA SALAM TO data_engineer;

-- CRUD ke seluruh tabel
GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA SALAM TO data_engineer;

-- Hak tambahan
ALTER DEFAULT PRIVILEGES IN SCHEMA SALAM
GRANT ALL PRIVILEGES ON TABLES TO data_engineer;

```

g. Full Script 02.1_backend_dev_test.sql

```

-- Uji Testing user backend_dev
SET ROLE backend_dev;

-- Uji SELECT
SELECT * FROM SALAM.mahasiswa;

-- Uji INSERT
INSERT INTO
SALAM.mahasiswa(nim,nama,prodi,fakultas,semester,angkatan,email,usia,jenis_kelamin,ipk
,status_aktif)

```

```

VALUES('1237050200','Test Backend.dev','Informatika','Sains dan
Teknologi',3,2025,'testbackend@mail.com',20,'L',3.5,'Aktif');

-- Uji UPDATE
UPDATE SALAM.mahasiswa SET ipk=3.6 WHERE nim='1237050200';

-- Uji DELETE
DELETE FROM SALAM.mahasiswa WHERE nim='1237050200';

-- Uji buat Table (Harus gagal)
CREATE TABLE SALAM.test_backend(id INT);

-- Kembali ke superuser
SET ROLE postgres;

```

h. Full Script 02.2_bi_dev_test.sql

```

-- Uji akses untuk role bi_dev
SET ROLE bi_dev;

-- Uji SELECT
SELECT * FROM SALAM.mahasiswa;

-- Uji INSERT (Harus gagal)
INSERT INTO
SALAM.mahasiswa(nim,nama,prodi,fakultas,semester,angkatan,email,usia,jenis_kelamin,ipk
,status_aktif)
VALUES('1237050201','Test BI','Informatika','Sains dan
Teknologi',3,2025,'testbi@mail.com',20,'P',3.7,'Aktif');

-- kembali ke superuser
SET ROLE postgres;

```

i. Full Script 02.3_data_engineer_test.sql

```

-- Uji akses untuk role data_engineer
SET ROLE data_engineer;

-- Uji CREATE TABLE
CREATE TABLE SALAM.test_dataeng(id SERIAL PRIMARY KEY, info TEXT);

-- Uji INSERT
INSERT INTO SALAM.test_dataeng(info) VALUES('Percobaan Data Engineer');
SELECT * FROM SALAM.test_dataeng;

-- Uji UPDATE
UPDATE SALAM.test_dataeng SET info='Update Berhasil' WHERE id=1;
SELECT * FROM SALAM.test_dataeng;

```

```
-- Uji DELETE
DELETE FROM SALAM.test_dataeng WHERE id=1;

-- Uji DROP TABLE
DROP TABLE SALAM.test_dataeng;

-- kembali ke superuser
SET ROLE postgres;
```

D. Fungsi Query Tambahan + Tes Fungsi

a. Fungsi cek status mahasiswa

- Fungsi :

```
-- 01. Fungsi sederhana untuk cek status mahasiswa
CREATE OR REPLACE FUNCTION SALAM.cek_status_aktif(p_nim CHAR(10))
RETURNS VARCHAR
LANGUAGE plpgsql
AS $$
DECLARE
    status_mhs VARCHAR(20);
BEGIN
    SELECT status_aktif INTO status_mhs
    FROM SALAM.mahasiswa
    WHERE nim = p_nim;

    IF status_mhs IS NULL THEN
        RETURN 'MAHASISWA TIDAK DITEMUKAN';
    ELSE
        RETURN status_mhs;
    END IF;
END;
$$;
```

- Testing :

```
-- 01. Test cek_status_aktif
-- test dengan data ada
SELECT
    nim AS test_nim,
    SALAM.cek_status_aktif(nim) AS status_dicek,
    '✓' AS hasil
FROM SALAM.mahasiswa
WHERE nim = '1247050005'

UNION ALL

-- Test dengan data tidak ada
```

```

SELECT
    '0000000000' as test_nim,
    SALAM.cek_status_aktif('0000000000') as status_dicek,
    '✓' as hasil;

```

Hasil :

- Tes data ada :

```

21 -- 01. Test cek_status_aktif
22 -- test dengan data ada
23
24     nim as test_nim,
25     SALAM.cek_status_aktif(nim) as status_dicek,
26     '✓' as hasil
27 FROM SALAM.mahasiswa
28 WHERE nim = '1247050005'
29

```

Data Output Messages Notifications

	test_nim	status_dicek	hasil
1	1247050005	Aktif	✓

- Tes data tidak ada :

```

31 -- Test dengan data tidak ada
32
33
34     '0000000000' as test_nim,
35     SALAM.cek_status_aktif('0000000000') as status_dicek,
36     '✗' as hasil;

```

Data Output Messages Notifications

	test_nim	status_dicek	hasil
1	00000000	MAHASISWA TIDAK DITEMUK...	✗

b. Fungsi Hitung mahasiswa sesuai prodi

- Fungsi :

```

-- 02. Fungsi hitung mahasiswa per prodi
CREATE OR REPLACE FUNCTION SALAM.hitung_mahasiswa_per_prodi(p_prodi VARCHAR)
RETURNS INTEGER
LANGUAGE plpgsql
AS $$

DECLARE
    total INTEGER;
BEGIN
    SELECT COUNT(*) INTO total
    FROM SALAM.mahasiswa
    WHERE prodi = p_prodi
    AND status_aktif = 'Aktif';

    RETURN total;
END;
$$;

```

- Testing

```

-- 02. Test hitung_mahasiswa_per_prodi
SELECT

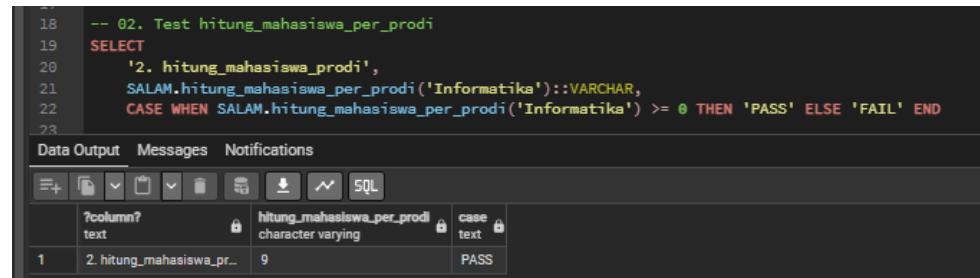
```

```

    '2. hitung_mahasiswa_prodi',
    SALAM.hitung_mahasiswa_per_prodi('Informatika')::VARCHAR,
    CASE WHEN SALAM.hitung_mahasiswa_per_prodi('Informatika') >= 0 THEN 'PASS'
ELSE 'FAIL' END

```

Hasil :



?column?	hitung_mahasiswa_per_prodi	case
	text	character varying
1	2. hitung_mahasiswa_pr...	PASS

c. Fungsi Kategori IPK

- Fungsi :

```

-- 03. Fungsi kategorisasi IPK
CREATE OR REPLACE FUNCTION SALAM.kategori_ipk(p_ipk NUMERIC)
RETURNS VARCHAR
LANGUAGE plpgsql
AS $$

BEGIN
    IF p_ipk IS NULL THEN
        RETURN 'BELUM ADA IPK';
    ELSIF p_ipk >= 3.5 THEN
        RETURN 'SANGAT MEMUASKAN';
    ELSIF p_ipk >= 3.0 THEN
        RETURN 'MEMUASKAN';
    ELSIF p_ipk >= 2.5 THEN
        RETURN 'CUKUP';
    ELSIF p_ipk >= 2.0 THEN
        RETURN 'KURANG';
    ELSE
        RETURN 'SANGAT KURANG';
    END IF;
END;
$$;

```

- Testing

```

-- 03. Test kategori_ipk
SELECT
    ipk_test AS nilai_ipk,
    SALAM.kategori_ipk(ipk_test) AS kategori,
    '✓' AS hasil
FROM (VALUES
    (NULL), (1.75), (2.25), (2.75), (3.25), (3.75)
) AS test_ipk(ipk_test);

```

```
-- Ambil 3 mahasiswa dengan IPK tertinggi beserta kategorinya
SELECT
    nim,
    nama,
    ipk,
    SALAM.kategori_ipk(ipk) as kategori_ipk,
    '✓' as hasil
FROM SALAM.mahasiswa
WHERE ipk IS NOT NULL
ORDER BY ipk DESC
LIMIT 3;
```

Hasil :

	nilai_ipk	kategori	hasil
1	[null]	BELUM ADA IPK	✓
2	1.75	SANGAT KURANG	✓
3	2.25	KURANG	✓
4	2.75	CUKUP	✓
5	3.25	MEMUASKAN	✓
6	3.75	SANGAT MEMUASK...	✓

- Ambil 3 IPK teratas

	nim	nama	ipk	kategori_ipk	hasil
1	1247253005	Sarah Fadilah	3.90	SANGAT MEMUASK...	✓
2	1237050004	Dewi Lestari	3.88	SANGAT MEMUASK...	✓
3	1227253001	Eka Nurhaliza	3.85	SANGAT MEMUASK...	✓

d. Fungsi Pencarian sederhana (Nim atau nama)

- Fungsi :

```
-- 04. Fungsi pencarian sederhana
CREATE OR REPLACE FUNCTION SALAM.cari_mahasiswa_sederhana(p_keyword VARCHAR)
RETURNS TABLE(
    nim CHAR(10),
    nama VARCHAR(100),
    prodi VARCHAR(100),
    semester INTEGER,
    ipk NUMERIC,
```

```

        status_aktif VARCHAR(20)
    )
LANGUAGE plpgsql
AS $$

BEGIN
    RETURN QUERY
SELECT
    m.nim, m.nama, m.prodi, m.semester, m.ipk, m.status_aktif
FROM SALAM.mahasiswa m
WHERE m.nim LIKE '%' || p_keyword || '%'
    OR m.nama ILIKE '%' || p_keyword || '%'
    ORDER BY m.nama;
END;
$$;

```

- Testing

```

-- 04. Test cari_mahasiswa_sederhana
-- Test pencarian dengan NIM
SELECT * FROM SALAM.cari_mahasiswa_sederhana('123');

-- Test pencarian dengan nama
SELECT * FROM SALAM.cari_mahasiswa_sederhana('Ahmad');

```

Hasil :

- Cari dengan NIM

	nim	name	prodi	semester	ipk	status_aktif
	character	character varying	character varying	integer	numeric	character varying
1	12370510...	Andi Setiawan	Biologi	5	3.33	Aktif
2	12370500...	Dewi Lestari	Informatika	5	3.88	Aktif
3	12372530...	Dini Larasati	Ekonomi Syariah	5	3.75	Aktif
4	12373540...	Hafidz Fadlan	Komunikasi & Penyiaran Isl...	5	3.12	Aktif
5	12371520...	Maya Salsabila	Aqidah & Filsafat Islam	5	3.25	Aktif
6	12370510...	Nia Rosita	Biologi	5	3.78	Aktif
7	12371500...	Refi Ramadhan	Informatika	5	3.15	Aktif
8	12371520...	Ridwan Kamil	Aqidah & Filsafat Islam	5	3.10	Aktif
9	12373540...	Silvia Rahmadani	Komunikasi & Penyiaran Isl...	5	3.33	Aktif
10	12372530...	Taufik Hideyat	Ekonomi Syariah	5	3.40	Aktif
11	12370502...	Test Backend.dev	Informatika	3	3.50	Aktif

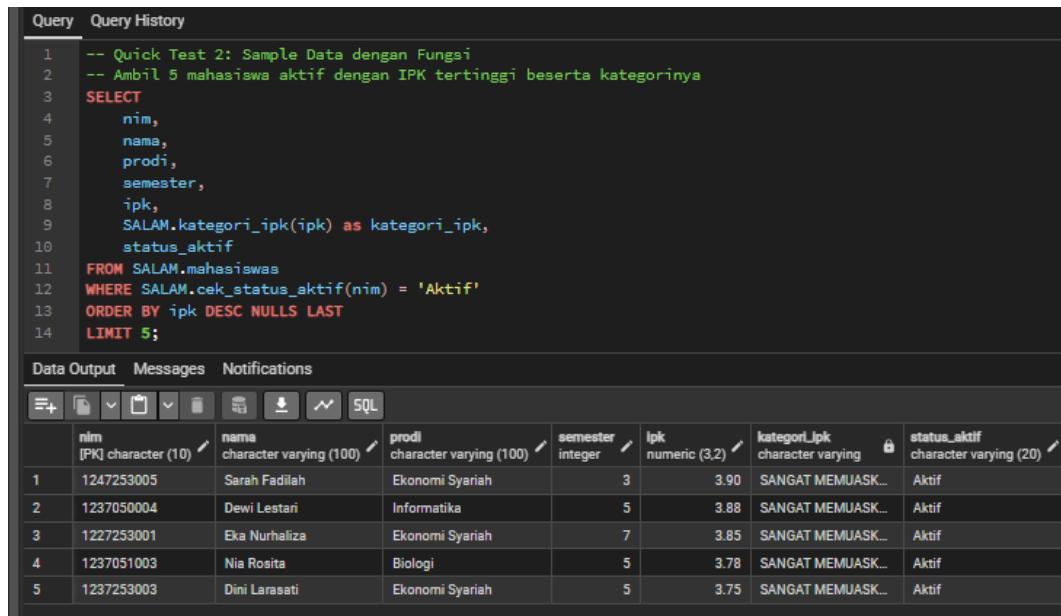
- Cari dengan Nama. Silvia terpanggil karena ada “ahmad” dalam nama nya

	nim	name	prodi	semester	ipk	status_aktif
	character	character varying	character varying	integer	numeric	character varying
1	12270500...	Ahmad Fauzi	Informatika	7	3.45	Aktif
2	12373540...	Silvia Rahmadani	Komunikasi & Penyiaran Isl...	5	3.33	Aktif

- e. Tes 2 : Sample data mahasiswa dengan fungsi
 Ambil 5 mahasiswa aktif dengan IPK tertinggi beserta kategorinya

```
-- Quick Test 2: Sample Data dengan Fungsi
-- Ambil 5 mahasiswa aktif dengan IPK tertinggi beserta kategorinya
SELECT
    nim,
    nama,
    prodi,
    semester,
    ipk,
    SALAM.kategori_ipk(ipk) as kategori_ipk,
    status_aktif
FROM SALAM.mahasiswa
WHERE SALAM.cek_status_aktif(nim) = 'Aktif'
ORDER BY ipk DESC NULLS LAST
LIMIT 5;
```

Hasil :



The screenshot shows a PostgreSQL query editor interface. The top section is the Query History, displaying the executed SQL script. The bottom section is the Data Output, showing a table with five rows of student data.

	nim	nama	prodi	semester	ipk	kategori_ipk	status_aktif
1	1247253005	Sarah Fadilah	Ekonomi Syariah	3	3.90	SANGAT MEMUASK...	Aktif
2	1237050004	Dewi Lestari	Informatika	5	3.88	SANGAT MEMUASK...	Aktif
3	1227253001	Eka Nurhaliza	Ekonomi Syariah	7	3.85	SANGAT MEMUASK...	Aktif
4	1237051003	Nia Rosita	Biologi	5	3.78	SANGAT MEMUASK...	Aktif
5	1237253003	Dini Larasati	Ekonomi Syariah	5	3.75	SANGAT MEMUASK...	Aktif

- f. Full Script 03_buat_fungsi.sql

```
-- 01. Fungsi sederhana untuk cek status mahasiswa
CREATE OR REPLACE FUNCTION SALAM.cek_status_aktif(p_nim CHAR(10))
RETURNS VARCHAR
LANGUAGE plpgsql
AS $$

DECLARE
    status_mhs VARCHAR(20);

BEGIN
    SELECT status_aktif INTO status_mhs
    FROM SALAM.mahasiswa
    WHERE nim = p_nim;
```

```

    IF status_mhs IS NULL THEN
        RETURN 'MAHASISWA TIDAK DITEMUKAN';
    ELSE
        RETURN status_mhs;
    END IF;
END;
$$;

-- 02. Fungsi hitung mahasiswa per prodi
CREATE OR REPLACE FUNCTION SALAM.hitung_mahasiswa_per_prodi(p_prodi VARCHAR)
RETURNS INTEGER
LANGUAGE plpgsql
AS $$

DECLARE
    total INTEGER;
BEGIN
    SELECT COUNT(*) INTO total
    FROM SALAM.mahasiswa
    WHERE prodi = p_prodi
    AND status_aktif = 'Aktif';

    RETURN total;
END;
$$;

-- 03. Fungsi kategorisasi IPK
CREATE OR REPLACE FUNCTION SALAM.kategori_ipk(p_ipk NUMERIC)
RETURNS VARCHAR
LANGUAGE plpgsql
AS $$

BEGIN
    IF p_ipk IS NULL THEN
        RETURN 'BELUM ADA IPK';
    ELSIF p_ipk >= 3.5 THEN
        RETURN 'SANGAT MEMUASKAN';
    ELSIF p_ipk >= 3.0 THEN
        RETURN 'MEMUASKAN';
    ELSIF p_ipk >= 2.5 THEN
        RETURN 'CUKUP';
    ELSIF p_ipk >= 2.0 THEN
        RETURN 'KURANG';
    ELSE
        RETURN 'SANGAT KURANG';
    END IF;
END;
$$;

-- 04. Fungsi pencarian sederhana

```

```

CREATE OR REPLACE FUNCTION SALAM.cari_mahasiswa_sederhana(p_keyword VARCHAR)
RETURNS TABLE(
    nim CHAR(10),
    nama VARCHAR(100),
    prodi VARCHAR(100),
    semester INTEGER,
    ipk NUMERIC,
    status_aktif VARCHAR(20)
)
LANGUAGE plpgsql
AS $$

BEGIN
    RETURN QUERY
    SELECT
        m.nim, m.nama, m.prodi, m.semester, m.ipk, m.status_aktif
    FROM SALAM.mahasiswa m
    WHERE m.nim ILIKE '%' || p_keyword || '%'
        OR m.nama ILIKE '%' || p_keyword || '%'
    ORDER BY m.nama;
END;
$$;

```

g. Full Script 03.1_quick_test_fungsi.sql

```

-- =====
-- QUICK TEST - FUNGSI SEDERHANA
-- =====

SET search_path TO SALAM;

-- Quick Test 1: Fungsi Dasar
SELECT 'QUICK TEST RESULTS:' as test_header;

-- 01. Test cek_status_aktif
-- test dengan data ada
SELECT
    nim as test_nim,
    SALAM.cek_status_aktif(nim) as status_dicek,
    '✓' as hasil
FROM SALAM.mahasiswa
WHERE nim = '1247050005'

UNION ALL

-- Test dengan data tidak ada
SELECT
    '0000000000' as test_nim,
    SALAM.cek_status_aktif('0000000000') as status_dicek,
    '✗' as hasil

```

```
'✓' as hasil;

UNION ALL

-- 02. Test hitung_mahasiswa_per_prodi
SELECT
    '2. hitung_mahasiswa_prodi',
    SALAM.hitung_mahasiswa_per_prodi('Informatika')::VARCHAR,
    CASE WHEN SALAM.hitung_mahasiswa_per_prodi('Informatika') >= 0 THEN 'PASS' ELSE
'FAIL' END

UNION ALL

-- 03. Test kategori_ipk
SELECT
    ipk_test as nilai_ipk,
    SALAM.kategori_ipk(ipk_test) as kategori,
    '✓' as hasil
FROM (VALUES
    (NULL), (1.75), (2.25), (2.75), (3.25), (3.75)
) AS test_ipk(ipk_test);

-- Ambil 3 mahasiswa dengan IPK tertinggi beserta kategorinya
SELECT
    nim,
    nama,
    ipk,
    SALAM.kategori_ipk(ipk) as kategori_ipk,
    '✓' as hasil
FROM SALAM.mahasiswas
WHERE ipk IS NOT NULL
ORDER BY ipk DESC
LIMIT 3;

UNION ALL

-- 04. Test cari_mahasiswa_sederhana
-- Test pencarian dengan NIM
SELECT * FROM SALAM.cari_mahasiswa_sederhana('123');

-- Test pencarian dengan nama
SELECT * FROM SALAM.cari_mahasiswa_sederhana('Ahmad');

-- Quick Test 2: Sample Data dengan Fungsi
-- Ambil 5 mahasiswa aktif dengan IPK tertinggi beserta kategorinya
SELECT
    nim,
    nama,
```

```
prodi,
semester,
ipk,
SALAM.kategori_ipk(ipk) as kategori_ipk,
status_aktif
FROM SALAM.mahasiswa
WHERE SALAM.cek_status_aktif(nim) = 'Aktif'
ORDER BY ipk DESC NULLS LAST
LIMIT 5;
```