

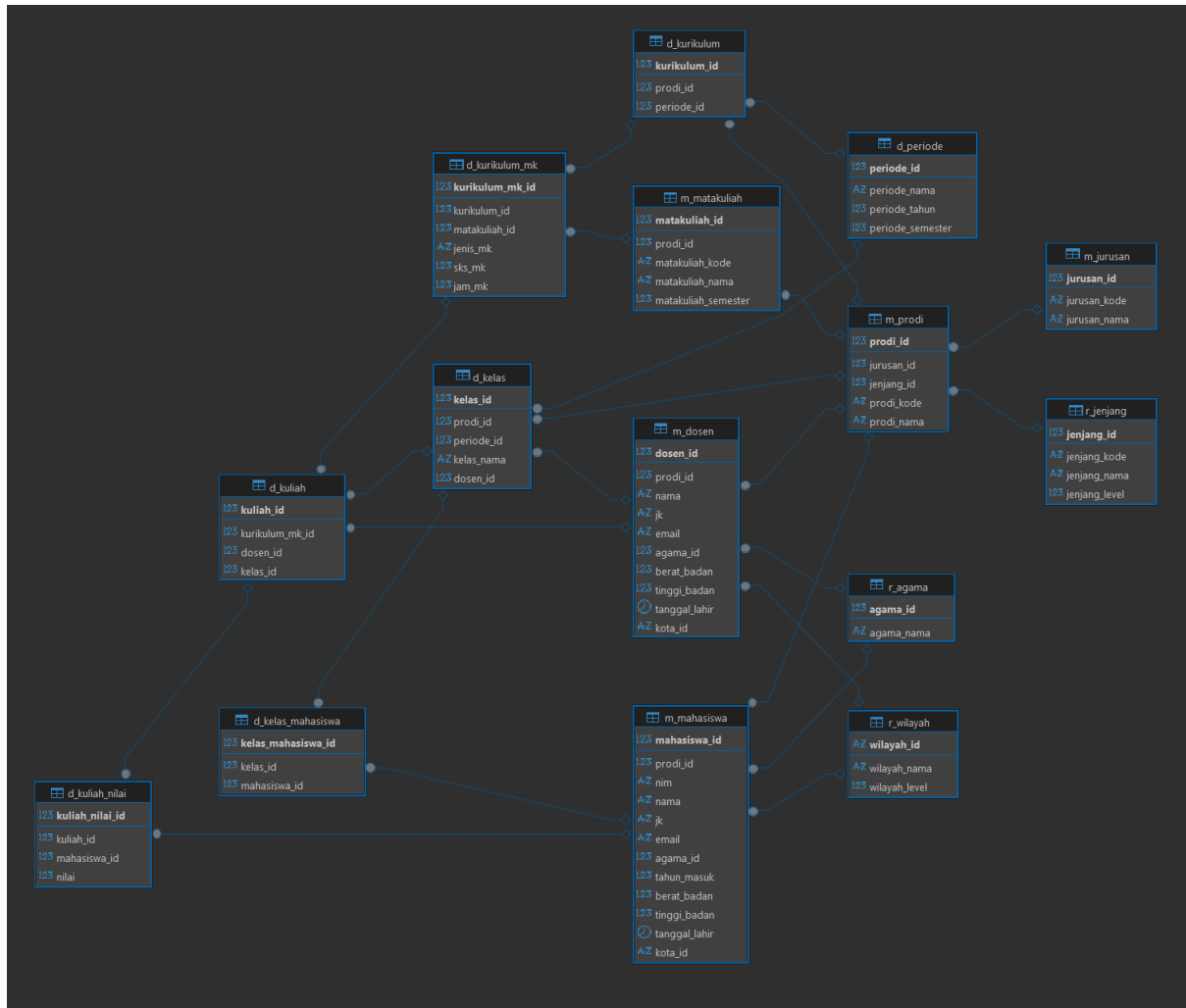


NAMA : Narendra Daniswara Alamsyah
NIM : 244107060095
KELAS : SIB-2G
MATKUL : Basis Data Lanjut

Praktikum 0:

Pertanyaan 1:

a.



Database ini memiliki struktur yang terorganisir dengan baik dan mengikuti prinsip normalisasi. Terdapat beberapa kelompok tabel utama:

1. Master Data (Tabel Referensi)

- r_agama - Data master agama
- r_jenjang - Data master jenjang pendidikan
- r_wilayah - Data master wilayah
- m_jurusan - Master jurusan
- m_prodi - Master program studi

2. Tabel Entitas Utama

- a. m_mahasiswa - Data mahasiswa (termasuk atribut fisik seperti tinggi_badan, berat_badan)
- b. m_dosen - Data dosen
- c. d_kelas - Data kelas
- d. m_matakuliah - Data mata kuliah

3. Tabel Relasi/Transaksi

- a. d_kuliah - Detail kuliah (menghubungkan kurikulum, dosen, kelas)
- b. d_kelas_mahasiswa - Relasi mahasiswa dengan kelas
- c. d_kurikulum - Data kurikulum
- d. d_kurikulum_mk - Detail kurikulum mata kuliah
- e. d_periode - Data periode akademik
- f. d_kuliah_nilai - Data nilai mahasiswa

b.

1. Relasi Hirarki Organisasi

m_jurusan (1) -----> (N) m_prodi

- Satu jurusan memiliki banyak program studi.
- Relasi melalui foreign key jurusan_id di tabel m_prodi.

2. Relasi Program Studi dengan Jenjang

r_jenjang (1) -----> (N) m_prodi

- Setiap prodi memiliki satu jenjang pendidikan (D3/D4/S1).
- Relasi melalui foreign key jenjang_id di tabel m_prodi.

3. Relasi Mahasiswa

m_mahasiswa (N) -----> (1) m_prodi

m_mahasiswa (N) -----> (1) r_agama

m_mahasiswa (N) -----> (1) r_wilayah

- Setiap mahasiswa terdaftar di satu program studi
- Setiap mahasiswa memiliki satu agama dan berasal dari satu wilayah (kota).
- Relasi melalui foreign key: prodi_id, agama_id, kota_id.

4. Relasi Dosen

m_dosen (N) -----> (1) m_prodi

m_dosen (N) -----> (1) r_agama

m_dosen (N) -----> (1) r_wilayah

- Dosen tergabung dalam satu program studi.
- Struktur serupa dengan tabel mahasiswa.

5. Relasi Kelas dan Mahasiswa

d_kelas (1) -----> (N) d_kelas_mahasiswa (N) -----> (1) m_mahasiswa

d_kelas (N) -----> (1) m_prodi

d_kelas (N) -----> (1) d_periode

d_kelas (N) -----> (1) m_dosen (sebagai DPA/Dosen Wali)

- Kelas dimiliki oleh satu prodi dan periode tertentu.
- Satu kelas memiliki banyak mahasiswa (relasi many-to-many melalui tabel junction d_kelas_mahasiswa).

6. Relasi Kurikulum

d_kurikulum (N) -----> (1) m_prodi

d_kurikulum (N) -----> (1) d_periode

d_kurikulum (1) -----> (N) d_kurikulum_mk (N) -----> (1) m_matakuliah

- Kurikulum dibuat per prodi dan periode
- Satu kurikulum memiliki banyak mata kuliah (relasi many-to-many melalui d_kurikulum_mk).
- Tabel d_kurikulum_mk menyimpan detail seperti SKS, jenis mata kuliah, dan jam mata kuliah.

7. Relasi Mata Kuliah dengan Program Studi

m_matakuliah (N) -----> (1) m_prodi

m_matakuliah (N) -----> (1) d_periode

Mata kuliah terkait dengan satu prodi.

Mata kuliah memiliki semester tertentu.

8. Relasi Perkuliahan (d_kuliah)

d_kuliah (N) -----> (1) d_kurikulum_mk

d_kuliah (N) -----> (1) m_dosen

d_kuliah (N) -----> (1) d_kelas

- Menghubungkan mata kuliah dari kurikulum, dosen pengampu, dan kelas.
- Tabel ini merepresentasikan "jadwal kuliah" atau "kelas mata kuliah".

9. Relasi Nilai

d_kuliah_nilai (N) -----> (1) d_kuliah

d_kuliah_nilai (N) -----> (1) m_mahasiswa

- Menyimpan nilai mahasiswa untuk setiap mata kuliah yang diambil.
- Menghubungkan mahasiswa dengan kuliah yang diikuti.

Praktikum 1:

Pertanyaan 2:

a.

Cek Tinggi Badan

```
CREATE OR REPLACE FUNCTION cek_tinggi_badan(tinggi numeric)
RETURNS text AS $$
    SELECT case when tinggi < 160 then 'Pendek'
                when tinggi between 160 and 170 then 'Sedang'
                else 'Tinggi' end;
$$ LANGUAGE sql;

SELECT nim, nama, tinggi_badan, cek_tinggi_badan(tinggi_badan) as kategori_tinggi
FROM m_mahasiswa;
```

Output Messages Notifications

Showing rows: 1 to 1000				
nim	nama	tinggi_badan	kategori_tinggi	
character varying (12)	character varying (100)	smallint	text	
2101100001	Dimas Dirja Samosir	146	Pendek	
2101220002	Sabri Jamal Mahendra M.Kom.	145	Pendek	
2101780003	Maida Usada	153	Pendek	
2101600004	Febi Hassanah S.Ked	145	Pendek	
2101280005	Zulfa Kuswandari	169	Sedang	
2101910006	Dimas Setiawan S.Pd	153	Pendek	
2101930007	Chelsea Vanesa Prastuti	163	Sedang	
2101520008	Laras Vicky Purwanti	159	Pendek	
2101810009	Widya Hariyah M.Ak	177	Tinggi	

Function `cek_tinggi_badan` mengkategorikan tinggi badan mahasiswa menjadi 3 kategori:

- "Pendek": tinggi < 160 cm
- "Sedang": tinggi antara 160 - 170 cm
- "Tinggi": tinggi > 170 cm

Cek Berat Badan

```
CREATE OR REPLACE FUNCTION hitung_bmi(berat numeric, tinggi numeric)
RETURNS text AS $$
DECLARE
    hasil text;
    bmi numeric;
BEGIN
    tinggi := tinggi / 100; -- konversi tinggi dari cm ke m
    bmi := berat / (tinggi * tinggi); -- perhitungan BMI

    IF (bmi < 16) THEN
        hasil := 'Kurus Parah';
    ELSIF (bmi >= 16 and bmi < 17) THEN
        hasil := 'Kurus Sedang';
    ELSIF (bmi >= 17 and bmi < 18.5) THEN
        hasil := 'Kurus';
    ELSIF (bmi >= 18.5 and bmi < 25) THEN
        hasil := 'Normal';
    ELSIF (bmi >= 25 and bmi < 30) THEN
        hasil := 'Gemuk';
    ELSIF (bmi >= 30 and bmi < 35) THEN
        hasil := 'Obesitas Kelas 1';
    ELSIF (bmi >= 35 and bmi < 40) THEN
        hasil := 'Obesitas Kelas 2';
    ELSE
        hasil := 'Obesitas Kelas 3';
    END IF;

    RETURN hasil;
END;
$$ LANGUAGE plpgsql;
```

```
SELECT nim, nama, tinggi_badan, berat_badan,
       hitung_bmi(berat_badan, tinggi_badan) as status_bmi
FROM m_mahasiswa;
```

	nim character varying (12)	nama character varying (100)	tinggi_badan smallint	berat_badan smallint	status_bmi text
1	2101100001	Dimas Dirja Samosir	146	59	Gemuk
2	2101220002	Sabri Jamal Mahendra M.Kom.	145	72	Obesitas Kelas 1
3	2101780003	Maida Usada	153	72	Obesitas Kelas 1
4	2101600004	Febi Hassanah S.Ked	145	80	Obesitas Kelas 2
5	2101280005	Zulfa Kuswandari	169	59	Normal
6	2101910006	Dimas Setiawan S.Pd	153	68	Gemuk
7	2101930007	Chelsea Vanesa Prastuti	163	79	Gemuk
8	2101520008	Laras Vicky Purwanti	159	68	Gemuk
9	2101810009	Widya Hariyah M.Ak	177	61	Normal
10	2101570010	Ulva Winarsih S.Sos	145	84	Obesitas Kelas 2

Function `hitung_bmi` menghitung Body Mass Index dengan rumus: **BMI = Berat (kg) / (Tinggi (m))²**

Function ini lebih detail dengan 8 kategori BMI berdasarkan standar WHO:

- BMI < 16: Kurus Parah
- BMI 16-17: Kurus Sedang
- BMI 17-18.5: Kurus
- BMI 18.5-25: Normal (ideal)
- BMI 25-30: Gemuk
- BMI 30-35: Obesitas Kelas 1
- BMI 35-40: Obesitas Kelas 2
- BMI ≥ 40: Obesitas Kelas 3

b.

1. Struktur Kode

`cek_tinggi_badan`: Menggunakan struktur SQL sederhana dengan CASE WHEN. Tidak memerlukan variabel atau blok procedural.

`hitung_bmi`: Menggunakan struktur procedural lengkap dengan DECLARE, BEGIN-END, dan assignment statements.

2. Logika Bisnis

`cek_tinggi_badan`: Evaluasi langsung berdasarkan 1 parameter dengan 3 kondisi sederhana.

`hitung_bmi`: Melakukan perhitungan matematis terlebih dahulu, baru evaluasi dengan 8 kondisi bertingkat.

3. Penggunaan Variabel

`cek_tinggi_badan`: Tidak menggunakan variabel, hasil langsung dikembalikan dari CASE statement.

`hitung_bmi`: Menggunakan 2 variabel lokal:

- `bmi` untuk menyimpan hasil perhitungan.
- `hasil` untuk menyimpan kategori yang akan dikembalikan.

c. Kapan Menggunakan SQL Function:

- Transformasi data sederhana (UPPER, LOWER, TRIM).
- Perhitungan matematis dasar.
- CASE statement untuk kategorisasi.
- Wrapper untuk query SELECT sederhana.
- Ketika performa adalah prioritas utama.
- Operasi yang sering dipanggil (hot path).

Praktikum 2:

Pertanyaan 3:

a.

```
1 CREATE OR REPLACE VIEW vw_mahasiswa AS
2 SELECT m.mahasiswa_id,
3        m.nim,
4        m.nama,
5        p.prodi_nama as prodi,
6        j.jurusan_nama as jurusan,
7        a.agama_nama as agama,
8        w.wilayah_nama as kota_lahir
9 FROM   m_mahasiswa m
10 JOIN   m_prodi p on p.prodi_id = m.prodi_id
11 JOIN   m_jurusan j on j.jurusan_id = p.jurusan_id
12 JOIN   r_agama a on a.agama_id = m.agama_id
13 JOIN   r_wilayah w on w.wilayah_id = m.kota_id;
14
15 SELECT * FROM vw_mahasiswa;
```

Data Output							
Showing rows: 1 to 1000							
Page No: 1 of 4							
mahasiswa_id	nim	nama	prodi	jurusan	agama	kota_lahir	
integer	character varying (12)	character varying (100)	character varying (80)	character varying (50)	character varying (50)	character varying (50)	ch
1	2101100001	Dimas Dirja Samosir	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Islam	K	
2	2101220002	Sabri Jamal Mahendra M.Kom.	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Katolik	K	
3	2101780003	Maida Usada	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Katolik	K	
4	2101600004	Febi Hassanah S.Ked	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Kristen Protestan	K	
5	2101280005	Zulfa Kuswandari	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Kristen Protestan	K	
6	2101910006	Dimas Setiawan S.Pd	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Kristen Protestan	K	
7	2101930007	Chelsea Vanesa Prastuti	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Budha	K	
8	2101520008	Laras Vicky Purwanti	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Budha	K	
9	2101810009	Widya Hariyah M.Ak	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Hindu	K	

Tujuan View adalah:

- Menyederhanakan query kompleks yang melibatkan multiple JOIN.
- Menggabungkan data mahasiswa dengan informasi dari tabel referensi (prodi, jurusan, agama, wilayah).
- Menampilkan nama/deskripsi yang readable daripada ID.

b.

Query

Query History

```
1 CREATE VIEW vw_mahasiswa AS
2 SELECT m.mahasiswa_id,
3        m.nim,
4        m.nama,
5        m.prodi_id,
6        p.prodi_nama AS prodi,
7        j.jurusan_id,
8        j.jurusan_nama AS jurusan,
9        m.agama_id,
10       a.agama_nama AS agama,
11       w.wilayah_nama AS kota_lahir
12 FROM m_mahasiswa m
13 JOIN m_prodi p ON p.prodi_id = m.prodi_id
14 JOIN m_jurusan j ON j.jurusan_id = p.jurusan_id
15 JOIN r_agama a ON a.agama_id = m.agama_id
16 JOIN r_wilayah w ON w.wilayah_id = m.kota_id;
17
18 SELECT * FROM vw_mahasiswa WHERE prodi_id = 1;
```

Data Output

Messages

Notifications

Showing rows: 1 to 210

Page No: 1 of 1

	mahasiswa_id integer	nim character varying (12)	nama character varying (100)	prodi_id bigint	prodi character varying (80)	jurusan_id integer	jurusan character varying (50)	agama_id bigint	agama character varying (50)
1	1	2101100001	Dimas Dirja Samosir	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	1	Islam
2	2	2101220002	Sabri Jamal Mahendra M.Kom.	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	3	Katolik
3	3	2101780003	Maida Usada	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	3	Katolik
4	4	2101600004	Febi Hassanah S.Ked	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	2	Kristen Protestan
5	5	2101280005	Zulfa Kuswandari	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	2	Kristen Protestan
6	6	2101910006	Dimas Setiawan S.Pd	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	2	Kristen Protestan

Praktikum 3:

Pertanyaan 4:

a.

```
CREATE MATERIALIZED VIEW mv_mahasiswa AS
SELECT m.mahasiswa_id,
       m.nim,
       m.nama,
       p.prodi_nama AS prodi,
       j.jurusan_nama AS jurusan,
       a.agama_nama AS agama,
       w.wilayah_nama AS kota_lahir
FROM m_mahasiswa m
JOIN m_prodi p ON p.prodi_id = m.prodi_id
JOIN m_jurusan j ON j.jurusan_id = p.jurusan_id
JOIN r_agama a ON a.agama_id = m.agama_id
JOIN r_wilayah w ON w.wilayah_id = m.kota_id;

SELECT * FROM mv_mahasiswa;
```

Output

Messages

Notifications

Showing rows: 1 to 1000

Page No: 1 of 4

mahasiswa_id integer	nim character varying (12)	nama character varying (100)	prodi character varying (80)	jurusan character varying (50)	agama character varying (50)	kota_lahir character varying (50)
1	2101100001	Dimas Dirja Samosir	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Islam	KAB. Bogor
2	2101220002	Sabri Jamal Mahendra M.Kom.	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Katolik	KAB. Bogor
3	2101780003	Maida Usada	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Katolik	KAB. Bogor
4	2101600004	Febi Hassanah S.Ked	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Kristen Protestan	KAB. Bogor
5	2101280005	Zulfa Kuswandari	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Kristen Protestan	KAB. Bogor
6	2101910006	Dimas Setiawan S.Pd	D3 - Manajemen Informasi	Jurusan Teknologi Informasi	Kristen Protestan	KAB. Bogor

Karakteristik Materialized View:

1. Menyimpan Data Secara Fisik

Berbeda dengan view biasa yang hanya menyimpan query definition.

2. Tidak Auto-Update

Ketika data di tabel sumber (m_mahasiswa, m_prodi, dll) berubah, materialized view tidak otomatis update.

3. Performa Query Sangat Cepat

Karena data sudah tersimpan, tidak perlu JOIN ulang saat query.

b.

```
CREATE MATERIALIZED VIEW mv_mahasiswa AS
SELECT  m.mahasiswa_id,
        m.nim,
        m.nama,
        m.prodi_id,
        p.prodi_nama as prodi,
        j.jurusan_id,
        j.jurusan_nama as jurusan,
        m.agama_id,
        a.agama_nama as agama,
        w.wilayah_nama as kota_lahir
FROM    m_mahasiswa m
JOIN    m_prodi p on p.prodi_id = m.prodi_id
JOIN    m_jurusan j on j.jurusan_id = p.jurusan_id
JOIN    r_agama a on a.agama_id = m.agama_id
JOIN    r_wilayah w on w.wilayah_id = m.kota_id;

SELECT * FROM mv_mahasiswa WHERE jurusan_id = 1;
```

Output Messages Notifications

Showing rows: 1 to 750 Page No: 1 of 1							
mahasiswa_id integer	nim character varying (12)	nama character varying (100)	prodi_id bigint	prodi character varying (80)	jurusan_id integer	jurusan character varying (50)	
1	2101100001	Dimas Dirja Samosir	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	
2	2101220002	Sabri Jamal Mahendra M.Kom.	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	
3	2101780003	Maida Usada	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	
4	2101600004	Febi Hassanah S.Ked	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	
5	2101280005	Zulfa Kuswandari	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	
6	2101910006	Dimas Setiawan S.Pd	1	D3 - Manajemen Informasi	1	Jurusan Teknologi Informasi	

c.

```
EXPLAIN ANALYZE
SELECT * FROM vw_mahasiswa WHERE prodi_id = 1;

EXPLAIN ANALYZE
SELECT * FROM mv_mahasiswa WHERE prodi_id = 1;
```

Output Messages Notifications

Showing rows: 1 to 750 Page No: 1 of 1	
QUERY PLAN	
text	
Seq Scan on mv_mahasiswa (cost=0.00..117.38 rows=210 width=128) (actual time=0.027..0.795 rows=210 loops=...	
Filter: (prodi_id = 1)	
Rows Removed by Filter: 3260	
Planning Time: 1.794 ms	
Execution Time: 0.836 ms	

Regular View:

- Melakukan 4 JOIN operations setiap kali query dijalankan.
- PostgreSQL harus:
 1. Scan tabel m_mahasiswa
 2. Join dengan m_prodi
 3. Join dengan m_jurusan
 4. Join dengan r_agama
 5. Join dengan r_wilayah
- Setiap JOIN membutuhkan Hash Table atau Nested Loop.
- Total cost tinggi karena operasi kompleks.

Materialized View:

- Hanya melakukan **Sequential Scan** pada satu tabel fisik.
- Data sudah di-denormalize dan tersimpan.
- Tidak ada overhead JOIN.
- Seperti query tabel biasa.

Praktikum 4:**Pertanyaan 5:**

- a. SP diatas membuat storage baru untuk membuat storage tambah_agama.
- b. Stored procedure ini memastikan hanya data unik yang tersimpan, dengan validasi yang robust terhadap variasi penulisan input.

Soal Latihan Jobsheet 06

1.

```
CREATE OR REPLACE FUNCTION cek_berat_ideal(  
    tinggi INTEGER,  
    berat INTEGER,  
    jenis_kelamin CHAR(1)  
)  
RETURNS TEXT AS $$  
DECLARE  
    berat_ideal NUMERIC(5,2);  
    selisih NUMERIC(5,2);  
    status TEXT;  
BEGIN  
    IF tinggi IS NULL OR berat IS NULL OR jenis_kelamin IS NULL THEN  
        RETURN 'Data tidak lengkap';  
    END IF;  
  
    IF jenis_kelamin NOT IN ('L', 'P') THEN  
        RETURN 'Jenis kelamin harus L atau P';  
    END IF;  
  
    IF jenis_kelamin = 'L' THEN  
        berat_ideal := (tinggi - 100) - ((tinggi - 100) * 0.1);  
    ELSE  
        berat_ideal := (tinggi - 100) - ((tinggi - 100) * 0.15);  
    END IF;  
  
    selisih := ABS(berat_ideal - berat);  
  
    IF selisih <= 2 THEN  
        status := 'Ideal (Berat Ideal: ' || ROUND(berat_ideal, 2) || ' kg, Selisih: ' || ROUND(selisih, 2) || ' kg)';  
    ELSIF berat < berat_ideal THEN  
        status := 'Kurang (Berat Ideal: ' || ROUND(berat_ideal, 2) || ' kg, Selisih: ' || ROUND(selisih, 2) || ' kg)';  
    ELSE  
        status := 'Berlebih (Berat Ideal: ' || ROUND(berat_ideal, 2) || ' kg, Selisih: ' || ROUND(selisih, 2) || ' kg)';  
    END IF;  
  
    RETURN status;  
END;  
$$ LANGUAGE plpgsql;  
  
SELECT  
    nim,  
    nama,  
    jk,  
    tinggi_badan,  
    berat_badan,  
    cek_berat_ideal(tinggi_badan, berat_badan, jk) as status_berat  
FROM m_mahasiswa  
LIMIT 10;
```

Output Messages Notifications

Showing rows: 1 to 10 of 1						
nim	nama	jk	tinggi_badan	berat_badan	status_berat	
character varying (12)	character varying (100)	character varying (1)	smallint	smallint	text	
2101100001	Dimas Dirja Samosir	P	146	59	Berlebih (Berat Ideal: 39.10 kg, Selisih: 19.90 kg)	
2101220002	Sabri Jamal Mahendra M.Kom.	P	145	72	Berlebih (Berat Ideal: 38.25 kg, Selisih: 33.75 kg)	
2101780003	Maida Usada	P	153	72	Berlebih (Berat Ideal: 45.05 kg, Selisih: 26.95 kg)	
2101600004	Febi Hassanah S.Ked	P	145	80	Berlebih (Berat Ideal: 38.25 kg, Selisih: 41.75 kg)	
2101280005	Zulfa Kuswandari	L	169	59	Kurang (Berat Ideal: 62.10 kg, Selisih: 3.10 kg)	
2101910006	Dimas Setiawan S.Pd	L	153	68	Berlebih (Berat Ideal: 47.70 kg, Selisih: 20.30 kg)	
2101930007	Chelsea Vanesa Prastuti	P	163	79	Berlebih (Berat Ideal: 53.55 kg, Selisih: 25.45 kg)	
2101520008	Laras Vicky Purwanti	P	159	68	Berlebih (Berat Ideal: 50.15 kg, Selisih: 17.85 kg)	
2101810009	Widya Hariyah M.Ak	P	177	61	Kurang (Berat Ideal: 65.45 kg, Selisih: 4.45 kg)	
2101570010	Ulva Winarsih S.Sos	L	145	84	Berlebih (Berat Ideal: 40.50 kg, Selisih: 43.50 kg)	

2.

a.

```
CREATE OR REPLACE VIEW vw_mahasiswa_kelas_dpa AS
SELECT
    m.mahasiswa_id,
    m.nim,
    m.nama AS nama_mahasiswa,
    m.jk,
    p.prodi_nama,
    k.kelas_nama,
    d.nama AS nama_dosen_wali,
    d.jk AS jk_dosen
FROM m_mahasiswa m
JOIN d_kelas_mahasiswa dkm ON dkm.mahasiswa_id = m.mahasiswa_id
JOIN d_kelas k ON k.kelas_id = dkm.kelas_id
JOIN m_dosen d ON d.dosen_id = k.dosen_id
JOIN m_prodi p ON p.prodi_id = m.prodi_id
ORDER BY k.kelas_nama, m.nama;

SELECT * FROM vw_mahasiswa_kelas_dpa;
```

Output Messages Notifications						
Showing rows: 1 to 1000 Page No: 1 of 4						
mahasiswa_id integer	nim character varying (12)	nama_mahasiswa character varying (100)	jk character varying (1)	prodi_nama character varying (80)	kelas_nama character varying (100)	
2804	2129670024	Abyasa Raden Waskita S.Pt	P	D4 - Akuntansi Manajemen	1A	
2781	2129870001	Ade Gantar Wijaya	P	D4 - Akuntansi Manajemen	1A	
1785	2117470015	Ade Nabila Oktaviani S.E.	L	S2 - Rekayasa Teknologi Manufaktur	1A	
2221	2123170011	Ade Vanya Yulianti	L	D4 - Manajemen Rekayasa Konstruksi	1A	
1010	2109410020	Ade Yulianti	L	D4 - Teknik Elektronika	1A	
1359	2112160009	Adhjarja Gunawan	P	S2 - Teknik Elektro	1A	

b.

```
CREATE OR REPLACE VIEW vw_kurikulum_matakuliah AS
SELECT
    k.kurikulum_id,
    p.prodi_id,
    p.prodi_nama,
    j.jurusan_nama,
    pe.periode_nama AS periode_kurikulum,
    mk.matakuliah_id,
    mk.matakuliah_kode,
    mk.matakuliah_nama,
    dkm.sks_mk,
    dkm.jenis_mk,
    dkm.jam_mk,
    mk.matakuliah_semester
FROM d_kurikulum k
JOIN m_prodi p ON p.prodi_id = k.prodi_id
JOIN m_jurusan j ON j.jurusan_id = p.jurusan_id
JOIN d_periode pe ON pe.periode_id = k.periode_id
JOIN d_kurikulum_mk dkm ON dkm.kurikulum_id = k.kurikulum_id
JOIN m_matakuliah mk ON mk.matakuliah_id = dkm.matakuliah_id
ORDER BY p.prodi_nama, mk.matakuliah_semester, mk.matakuliah_nama;

SELECT * FROM vw_kurikulum_matakuliah;
```

Output Messages Notifications						
Showing rows: 1 to 222 Page No: 1 of 1						
kurikulum_id integer	prodi_id integer	prodi_nama character varying (80)	jurusan_nama character varying (50)	periode_kurikulum character varying (30)	matakuliah_id integer	matakuliah_kode character varying (10)
4	4	D2 - Pengembangan Piranti Lunak Situs	Jurusan Teknologi Informasi	2022/2023 Ganjil	21	PLS403
4	4	D2 - Pengembangan Piranti Lunak Situs	Jurusan Teknologi Informasi	2022/2023 Ganjil	19	PLS401

3.

a.

```
CREATE MATERIALIZED VIEW mv_dosen_mengajar AS
SELECT
    d.dosen_id,
    d.nama AS nama_dosen,
    d.jk AS jk_dosen,
    d.email AS email_dosen,
    p.prodi_nama AS prodi_dosen,
    mk.matakuliah_id,
    mk.matakuliah_kode,
    mk.matakuliah_nama,
    dkm.sks_mk,
    k.kelas_nama,
    pe.periode_nama,
    pe.periode_tahun,
    pe.periode_semester
FROM m_dosen d
JOIN d_kuliah dk ON dk.dosen_id = d.dosen_id
JOIN d_kurikulum_mk dkm ON dkm.kurikulum_mk_id = dk.kurikulum_mk_id
JOIN m_matakuliah mk ON mk.matakuliah_id = dkm.matakuliah_id
JOIN d_kelas k ON k.kelas_id = dk.kelas_id
JOIN d_periode pe ON pe.periode_id = k.periode_id
JOIN m_prodi p ON p.prodi_id = d.prodi_id
ORDER BY d.nama, mk.matakuliah_nama;

CREATE INDEX idx_mv_dosen_mengajar_dosen ON mv_dosen_mengajar(dosen_id);
CREATE INDEX idx_mv_dosen_mengajar_mk ON mv_dosen_mengajar(matakuliah_id);

SELECT * FROM mv_dosen_mengajar;
```

Output Messages Notifications

Showing rows: 1 to 348 Page No: 1 of 1

dosen_id integer	nama_dosen character varying (50)	jk_dosen character varying (1)	email_dosen character varying (50)	prodi_dosen character varying (80)	matakuliah_id integer
138	Ade Ganjaran Irawan	L	oliva.pangestu@example.net	D3 - Teknologi Konstruksi Jalan, Jembatan, dan Bangunan ...	
98	Aditya Prasasta S.E.I	P	bkuswandari@example.net	D3 - Teknologi Pemeliharaan Pesawat Udara	
98	Aditya Prasasta S.E.I	P	bkuswandari@example.net	D3 - Teknologi Pemeliharaan Pesawat Udara	
205	Ajeng Wijayanti	P	zmahendra@example.net	D4 - Keuangan	
205	Ajeng Wijayanti	P	zmahendra@example.net	D4 - Keuangan	
261	Amelia Anggraini S.Farm	L	iwaluyo@example.net	D4 - Bahasa Inggris untuk Industri Pariwisata	
250	Ami Ajeng Prastuti	P	tlaksita@example.com	D4 - Manajemen Pemasaran	
199	Ani Keisha Nurdianti	L	bwidiastuti@example.org	D3 - Akuntansi	

b.

```
CREATE MATERIALIZED VIEW mv_mahasiswa_nilai AS
SELECT
    m.mahasiswa_id,
    m.nim,
    m.nama as nama_mahasiswa,
    p.prodi_nama,
    k.kelas_nama,
    mk.matakuliah_kode,
    mk.matakuliah_nama,
    dkm.sks_mk,
    pe.periode_nama,
    pe.periode_tahun,
    pe.periode_semester,
    dkn.nilai as nilai_mahasiswa,
    CASE
        WHEN dkn.nilai >= 80 THEN 'A'
        WHEN dkn.nilai >= 75 THEN 'AB'
        WHEN dkn.nilai >= 70 THEN 'B'
        WHEN dkn.nilai >= 65 THEN 'BC'
        WHEN dkn.nilai >= 60 THEN 'C'
        WHEN dkn.nilai >= 50 THEN 'D'
        ELSE 'E'
    END as grade,
    CASE
        WHEN dkn.nilai >= 60 THEN 'Lulus'
        ELSE 'Tidak Lulus'
    END as status_lulus
FROM m_mahasiswa m
JOIN d_kelas_mahasiswa dkm_rel ON dkm_rel.mahasiswa_id = m.mahasiswa_id
JOIN d_kelas k ON k.kelas_id = dkm_rel.kelas_id
JOIN d_kuliah_nilai dkn ON dkn.mahasiswa_id = m.mahasiswa_id
JOIN d_kuliah dk ON dk.kuliah_id = dkn.kuliah_id
JOIN d_kurikulum_mk dkm ON dkm.kurikulum_mk_id = dk.kurikulum_mk_id
JOIN m_matakuliah mk ON mk.matakuliah_id = dkm.matakuliah_id
JOIN d_periode pe ON pe.periode_id = k.periode_id
JOIN m_prodi p ON p.prodi_id = m.prodi_id
ORDER BY m.nim, pe.periode_tahun, pe.periode_semester, mk.matakuliah_nama;

CREATE INDEX idx_mv_mahasiswa_nilai_mhs ON mv_mahasiswa_nilai(mahasiswa_id);
CREATE INDEX idx_mv_mahasiswa_nilai_nim ON mv_mahasiswa_nilai(nim);

SELECT * FROM mv_mahasiswa_nilai;
```

Output Messages Notifications

Showing rows: 1 to 1000 Page No: 1 of 11									
matakuliah_kode character varying (12)	matakuliah_nama character varying (50)	sks_mk smallint	periode_nama character varying (30)	periode_tahun integer	periode_semester smallint	nilai_mahasiswa smallint	grade text	status_lulus text	
MI102	Analisis Sistem Informasi	2	2022/2023 Ganjil	2022	1	46	E	Tidak Lulus	
MI105	Audit Sistem Informasi	3	2022/2023 Ganjil	2022	1	61	C	Lulus	
MI103	Manajemen Proyek TI	2	2022/2023 Ganjil	2022	1	88	A	Lulus	
MI102	Analisis Sistem Informasi	2	2022/2023 Ganjil	2022	1	85	A	Lulus	
MI105	Audit Sistem Informasi	3	2022/2023 Ganjil	2022	1	67	BC	Lulus	
MI103	Manajemen Proyek TI	2	2022/2023 Ganjil	2022	1	78	AB	Lulus	
MI102	Analisis Sistem Informasi	2	2022/2023 Ganjil	2022	1	44	E	Tidak Lulus	
MI105	Audit Sistem Informasi	3	2022/2023 Ganjil	2022	1	58	D	Tidak Lulus	
MI103	Manajemen Proyek TI	2	2022/2023 Ganjil	2022	1	59	D	Tidak Lulus	
MI102	Analisis Sistem Informasi	2	2022/2023 Ganjil	2022	1	50	D	Tidak Lulus	
MI105	Audit Sistem Informasi	3	2022/2023 Ganjil	2022	1	50	D	Tidak Lulus	
MI103	Manajemen Proyek TI	2	2022/2023 Ganjil	2022	1	56	D	Tidak Lulus	
MI102	Analisis Sistem Informasi	2	2022/2023 Ganjil	2022	1	87	A	Lulus	
MI105	Audit Sistem Informasi	3	2022/2023 Ganjil	2022	1	46	E	Tidak Lulus	
MI103	Manajemen Proyek TI	2	2022/2023 Ganjil	2022	1	56	D	Tidak Lulus	
MI102	Analisis Sistem Informasi	2	2022/2023 Ganjil	2022	1	77	AB	Lulus	

4.

```
CREATE OR REPLACE PROCEDURE insert_mahasiswa_validasi(  
    p_nim VARCHAR,  
    p_nama VARCHAR,  
    p_jk CHAR(1),  
    p_prodi_id INTEGER,  
    p_agama_id INTEGER,  
    p_kota_id INTEGER,  
    p_tanggal_lahir DATE,  
    p_tahun_masuk INTEGER,  
    p_tinggi_badan INTEGER,  
    p_berat_badan INTEGER,  
    p_email VARCHAR DEFAULT NULL  
)  
LANGUAGE plpgsql
```