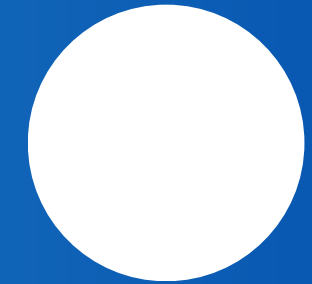
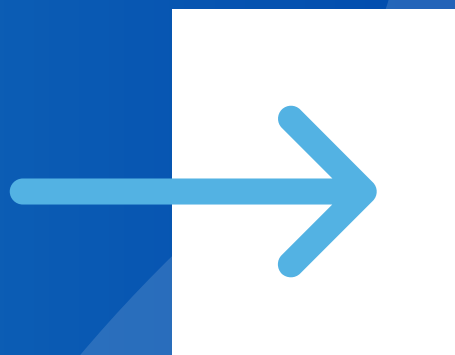


IMPLEMENTASI BASIS DATA (STORE PROCEDURE)



Kelompok 8

- Dimas Rizky Fajar Pamungkas / 24.N1.0026
- Adonai Shan Matthew / 24.N1.0043
- Josephine Cornelia Kristanto / 24.N4.0001



PROCEDURE 1

procedure ini mengatur proses input data Bruder ke tabel databruder dengan validasi. Untuk flow-nya gini:

1. Mengecek apakah id_user yang dikirim ada di tabel users.
2. Kalau ada, maka data Bruder (nama, foto/gambar, TTL, alamat, tahun masuk/prasetia/kaul) dimasukkan ke tabel databruder.
3. Kalau nggak ada, prosedur langsung berhenti dan kasih error custom: "id_user tidak ditemukan, insert dibatalkan".

Intinya:

- Mencegah data nyasar (tidak mungkin ada Bruder dengan id_user yang belum terdaftar).
- Otomatisasi supaya tiap kali input, aturan validasinya konsisten.

```
DELIMITER $$

CREATE PROCEDURE insertBruder(
    IN p_id_user INT,
    IN p_nama VARCHAR(100),
    IN p_gambar LONGBLOB,
    IN p_ttl VARCHAR(200),
    IN p_alamat VARCHAR(255),
    IN p_tahun_masuk YEAR,
    IN p_tahun_prasetia YEAR,
    IN p_tahun_kaul YEAR
)
BEGIN
    DECLARE id_exists INT;

    -- cek apakah user ada di tabel users
    SELECT COUNT(*) INTO id_exists
    FROM users
    WHERE id_user = p_id_user;

    IF id_exists > 0 THEN
        INSERT INTO databruder (
            id_user,
            nama_bruder,
            gambar_bruder,
            ttl_bruder,
            alamat_bruder,
            tahun_masuk_postulan,
            tahun_prasetia_pertama,
            tahun_kaul_kekal
        ) VALUES (
            p_id_user,
            p_nama,
            p_gambar,
            p_ttl,
            p_alamat,
            p_tahun_masuk,
            p_tahun_prasetia,
            p_tahun_kaul
        );
    ELSE
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'id_user tidak ditemukan, insert dibatalkan';
    END IF;
END $$

DELIMITER ;
```



HASIL

```
CALL insertBruder(  
  1, -- p_id_user  
  'Bruder Antonius', -- p_nama  
  NULL, -- p_gambar  
  'Semarang, 01-01-1990', -- p_ttl  
  'Jl. Imam Bonjol No. 5', -- p_alamat  
  2009, -- p_tahun_masuk  
  2011, -- p_tahun_prasetia  
  2018 -- p_tahun_kaul  
);
```

```
CALL insertBruder( 1, 'Bruder Antonius', NULL, 'Semarang, 01-01-1990', 'Jl. Imam Bonjol No. 5', 2009, 2011, 2018 );
```

PROCEDURE 2

Pada procedure ini menampilkan daftar pengeluaran terbesar seorang bruder (user) dalam satu tahun tertentu, dibatasi sejumlah baris (Top N) sesuai kebutuhan. Misalnya:

- “Pengeluaran apa aja yang paling banyak dikeluarkan oleh Bruder A di tahun 2025?”
- Maka prosedur ini langsung kasih daftar pos pengeluaran dari yang terbesar sampai terkecil, misalnya Top 3, Top 5, atau Top 10.

Intinya:

buat laporan analisis keuangan → fokus ke pengeluaran terbesar per bruder per tahun.

```
DELIMITER $$

CREATE PROCEDURE top_pengeluaran_opt (
    IN p_id_user INT,
    IN p_tahun INT,
    IN p_limit INT
)
BEGIN
    -- safety untuk limit
    IF p_limit IS NULL OR p_limit <= 0 THEN
        SET p_limit = 10;
    END IF;

    SELECT
        kh.id_perkiraan AS id_pos,
        SUM(kh.jumlah) AS total_pengeluaran
    FROM kas_harian kh
    WHERE kh.id_status = 0
        AND kh.id_user = p_id_user
        AND kh.tanggal >= MAKEDATE(p_tahun, 1)
        AND kh.tanggal < MAKEDATE(p_tahun + 1, 1)
    GROUP BY kh.id_perkiraan
    ORDER BY total_pengeluaran DESC
    LIMIT p_limit;
END$$

DELIMITER ;
```



HASIL

```
CALL top_pengeluaran_opt(1, 2024, 5);
```

id_perkiraan	total_pengeluaran ▾ 1
101	500000.00
102	350000.00
103	200000.00

PROCEDURE 3

Procedure yang ini untuk mencari dan menampilkan data Bruder berdasarkan kata kunci yang cocok dengan nama atau alamat. Jadi:

- Kalau kita masukan keyword "Yohanes" → keluar semua Bruder yang namanya mengandung kata Yohanes.
- Kalau keyword "Semarang" → keluar semua Bruder yang alamatnya mengandung kata Semarang.

Intinya:

mempermudah pencarian data Bruder tanpa harus ingat persis ID-nya, cukup pakai nama atau alamat.

```
DELIMITER $$

CREATE PROCEDURE cari_bruder (
    IN p_keyword VARCHAR(100)
)
BEGIN
    SELECT
        id_databruder,
        id_user,
        nama_bruder,
        gambar_bruder,
        ttl_bruder,
        alamat_bruder,
        tahun_masuk_postulan,
        tahun_prasetia_pertama,
        tahun_kaul_kekal
    FROM databruder
    WHERE nama_bruder LIKE CONCAT('%', p_keyword, '%')
        OR alamat_bruder LIKE CONCAT('%', p_keyword, '%');
END$$

DELIMITER ;
```



HASIL

```
CALL cari_bruder('Bruder Yosef');
```



id_databruder	id_user	nama_bruder	gambar_bruder	ttl_bruder	alamat_bruder	tahun_masuk_postulan	tahun_prasetia_pertama	tahun_kaul_kekal
2	1	Bruder Yosef	NULL	Magelang, 12-05-1992	Jl. Pandanaran No. 10	2010	2012	2019

PROCEDURE 4

Procedure yang terakhir ini menampilkan daftar riwayat tugas seorang Bruder berdasarkan ID-nya.

- Input: id_bruder (siapa yang mau dicari).
- Output: semua tugas yang pernah dijalani bruder tersebut, lengkap dengan tahun dan deskripsinya.
- Data ditampilkan berurutan dari yang terbaru ke yang lama.

Intinya:

dipakai buat melihat sejarah penugasan setiap bruder secara rapi dan cepat.

```
DELIMITER $$

CREATE PROCEDURE lihat_riwayat_tugas (
    IN p_id_bruder INT
)
BEGIN
    SELECT
        rt.id_tugas,
        rt.id_bruder,
        db.nama_bruder,
        rt.tahun,
        rt.deskripsi_tugas
    FROM riwayat_tugas_bruder rt
    JOIN databruder db ON rt.id_bruder = db.id_databruder
    WHERE rt.id_bruder = p_id_bruder
    ORDER BY rt.tahun DESC;
END$$

DELIMITER ;
```




HASIL

```
CALL lihat_riwayat_tugas(11);
```

id_tugas	id_bruder	nama_bruder	tahun	deskripsi_tugas
9	11	Bruder Antonius	2023	Pembimbing rohani di seminari
8	11	Bruder Antonius	2020	Menjadi bendahara komunitas
7	11	Bruder Antonius	2017	Mendampingi retreat siswa