

SISTEM PERPUSTAKAAN SEKOLAH

1. Tabel Buku

| ID Buku | Judul Buku | Penulis | Kategori | Stok |
|---------|-----------------------------|-----------------|--------------|------|
| 1 | Algoritma dan Pemrograman | Andi Wijaya | Teknologi | 5 |
| 2 | Dasar-dasar Database | Budi Santoso | Teknologi | 7 |
| 3 | Matematika Diskrit | Rina Sari | Matematika | 4 |
| 4 | Sejarah Dunia | John Smith | Sejarah | 3 |
| 5 | Pemrograman Web dengan PHP | Eko Prasetyo | Teknologi | 8 |
| 6 | Sistem Operasi | Dian Kurniawan | Teknologi | 6 |
| 7 | Jaringan Komputer | Ahmad Fauzi | Teknologi | 5 |
| 8 | Cerita Rakyat Nusantara | Lestari Dewi | Sastra | 9 |
| 9 | Bahasa Inggris untuk Pemula | Jane Doe | Bahasa | 10 |
| 10 | Biologi Dasar | Budi Rahman | Sains | 7 |
| 11 | Kimia Organik | Siti Aminah | Sains | 5 |
| 12 | Teknik Elektro | Ridwan Hakim | Teknik | 6 |
| 13 | Fisika Modern | Albert Einstein | Sains | 4 |
| 14 | Manajemen Waktu | Steven Covey | Pengembangan | 8 |
| 15 | Strategi Belajar Efektif | Tony Buzan | Pendidikan | 6 |

2. Tabel Siswa

| ID Siswa | Nama | Kelas |
|----------|-----------------|---------|
| 1 | Andi Saputra | X-RPL |
| 2 | Budi Wijaya | X-TKJ |
| 3 | Citra Lestari | XI-RPL |
| 4 | Dewi Kurniawan | XI-TKJ |
| 5 | Eko Prasetyo | XII-RPL |
| 6 | Farhan Maulana | XII-TKJ |
| 7 | Gita Permata | X-RPL |
| 8 | Hadi Sucipto | X-TKJ |
| 9 | Intan Permadi | XI-RPL |
| 10 | Joko Santoso | XI-TKJ |
| 11 | Kartika Sari | XII-RPL |
| 12 | Lintang Putri | XII-TKJ |
| 13 | Muhammad Rizky | X-RPL |
| 14 | Novi Andriana | X-TKJ |
| 15 | Olivia Hernanda | XI-RPL |

3. Tabel Peminjaman

| ID Peminjaman | ID Siswa | ID Buku | Tanggal Pinjam | Tanggal Kembali | Status |
|---------------|----------|---------|----------------|-----------------|--------------|
| 1 | 11 | 2 | 2025-02-01 | 2025-02-08 | Dipinjam |
| 2 | 2 | 5 | 2025-01-28 | 2025-02-04 | Dikembalikan |
| 3 | 3 | 8 | 2025-02-02 | 2025-02-09 | Dipinjam |
| 4 | 4 | 10 | 2025-01-30 | 2025-02-06 | Dikembalikan |
| 5 | 5 | 3 | 2025-01-25 | 2025-02-01 | Dikembalikan |
| 6 | 15 | 7 | 2025-02-01 | 2025-02-08 | Dipinjam |
| 7 | 7 | 1 | 2025-01-29 | 2025-02-05 | Dikembalikan |
| 8 | 8 | 9 | 2025-02-03 | 2025-02-10 | Dipinjam |
| 9 | 13 | 4 | 2025-01-27 | 2025-02-03 | Dikembalikan |
| 10 | 10 | 11 | 2025-02-01 | 2025-02-08 | Dipinjam |

TUGAS

1. Buatlah database dengan nama db_perpus.

```
CREATE DATABASE db_perpus;
```

2. Buatlah table buku, siswa dan peminjaman.

Buku

```
CREATE TABLE Buku (ID_Buku int PRIMARY KEY AUTO_INCREMENT, Judul_Buku  
VARCHAR(50), Penulis VARCHAR(50), Kategori VARCHAR(30), Stok INT);
```

Siswa

```
CREATE TABLE Siswa (ID_Siswa int PRIMARY KEY AUTO_INCREMENT, Nama VARCHAR(50),  
Kelas VARCHAR(50));
```

Peminjaman

```
CREATE TABLE Peminjaman (ID_Peminjaman int PRIMARY KEY AUTO_INCREMENT, ID_Siswa  
INT, ID_Buku INT, Tanggal_Pinjam DATE, Tanggal_Kembali DATE, Status VARCHAR(50));
```

3. Input 5 record di setiap table menggunakan query INSERT, UPDATE, DELETE.

```
INSERT INTO buku (Judul_Buku, Penulis, Kategori, Stok) VALUES  
( 'Algoritma dan Pemrograman', 'Andi Wijaya', 'Teknologi', 5),  
( 'Dasar-dasar Database', 'Budi Santoso', 'Teknologi', 7),  
( 'Matematika Diskrit', 'Rina Sari', 'Matematika', 4),  
( 'Sejarah Dunia', 'John Smith', 'Sejarah', 3),  
( 'Pemrograman Web dengan PHP', 'Eko Prasetyo', 'Teknologi', 8);
```

```
INSERT INTO siswa (Nama, Kelas) VALUES  
( 'Andi Saputra', 'X-RPL'),  
( 'Budi Wijaya', 'X-TKJ'),
```

```
('Citra Lestari', 'XI-RPL'),  
( 'Dewi Kurniawan', 'XI-TKJ'),  
( 'Eko Prasetyo', 'XII-RPL');
```

```
INSERT INTO peminjaman (ID_Siswa, ID_Buku, Tanggal_Pinjam, Tanggal_Kembali, Status)  
VALUES  
(11, 2, '2025-02-01', '2025-02-08', 'Dipinjam'),  
(2, 5, '2025-01-28', '2025-02-04', 'Dikembalikan'),  
(3, 8, '2025-02-02', '2025-02-09', 'Dipinjam'),  
(4, 10, '2025-01-30', '2025-02-06', 'Dikembalikan'),  
(5, 3, '2025-01-25', '2025-02-01', 'Dikembalikan');
```

4. Input 10 record di setiap table menggunakan stored procedure INSERT.

Buku

DELIMITER //

```
CREATE PROCEDURE insertBuku(  
    IN p_judul VARCHAR(50),  
    IN p_penulis VARCHAR(50),  
    IN p_kategori VARCHAR(30),  
    IN p_stok INT  
)  
BEGIN  
    INSERT INTO buku (Judul_Buku, Penulis, Kategori, Stok)  
    VALUES (p_judul, p_penulis, p_kategori, p_stok);  
END //
```

call buku

```
CALL insertBuku('Sistem Operasi', 'Dian Kurniawan', 'Teknologi', 6);  
CALL insertBuku('Jaringan Komputer', 'Ahmad Fauzi', 'Teknologi', 5);  
CALL insertBuku('Cerita Rakyat Nusantara', 'Lestari Dewi', 'Sastra', 9);  
CALL insertBuku('Bahasa Inggris untuk Pemula', 'Jane Doe', 'Bahasa', 10);  
CALL insertBuku('Biologi Dasar', 'Budi Rahman', 'Sains', 7);  
CALL insertBuku('Kimia Organik', 'Siti Aminah', 'Sains', 5);  
CALL insertBuku('Teknik Elektro', 'Ridwan Hakim', 'Teknik', 6);  
CALL insertBuku('Fisika Modern', 'Albert Einstein', 'Sains', 4);  
CALL insertBuku('Manajemen Waktu', 'Steven Covey', 'Pengembangan', 8);  
CALL insertBuku('Strategi Belajar Efektif', 'Tony Buzan', 'Pendidikan', 6);
```

Siswa

DELIMITER //

```
CREATE PROCEDURE insertSiswa(  
    IN p_nama VARCHAR(255),  
    IN p_kelas VARCHAR(50)  
)
```

```
BEGIN
    INSERT INTO siswa (Nama, Kelas)
    VALUES (p_nama, p_kelas);
END //
```

call siswa

```
CALL insertSiswa('Farhan Maulana', 'XII-TKJ');
CALL insertSiswa('Gita Permata', 'X-RPL');
CALL insertSiswa('Hadi Sucipto', 'X-TKJ');
CALL insertSiswa('Intan Permadi', 'XI-RPL');
CALL insertSiswa('Joko Santoso', 'XI-TKJ');
CALL insertSiswa('Kartika Sari', 'XII-RPL');
CALL insertSiswa('Lintang Putri', 'XII-TKJ');
CALL insertSiswa('Muhammad Rizky', 'X-RPL');
CALL insertSiswa('Novi Andriana', 'X-TKJ');
CALL insertSiswa('Olivia Hernanda', 'XI-RPL');
```

Peminjaman

```
DELIMITER //
```

```
CREATE PROCEDURE insertPeminjaman(
    IN p_id_siswa INT,
    IN p_id_buku INT,
    IN p_tanggal_pinjam DATE,
    IN p_tanggal_kembali DATE,
    IN p_status ENUM('Dipinjam', 'Dikembalikan')
)
BEGIN
    INSERT INTO peminjaman (ID_Siswa, ID_Buku, Tanggal_Pinjam, Tanggal_Kembali, Status)
    VALUES (p_id_siswa, p_id_buku, p_tanggal_pinjam, p_tanggal_kembali, p_status);
END //
```

call peminjaman

```
CALL insertPeminjaman(15, 7, '2025-02-01', '2025-02-08', 'Dipinjam');
CALL insertPeminjaman(7, 1, '2025-01-29', '2025-02-05', 'Dikembalikan');
CALL insertPeminjaman(8, 9, '2025-02-03', '2025-02-10', 'Dipinjam');
CALL insertPeminjaman(13, 4, '2025-01-27', '2025-02-03', 'Dikembalikan');
CALL insertPeminjaman(10, 11, '2025-02-01', '2025-02-08', 'Dipinjam');
```

5. Buatlah stored procedure UPDATE, DELETE di setiap table.

(UPDATE & DELETE)

BUKU

UPDATE

```
DELIMITER //
```

```
CREATE PROCEDURE updateBuku(
    IN p_id_buku INT,
```

```

        IN p_judul VARCHAR(255),
        IN p_penulis VARCHAR(255),
        IN p_kategori VARCHAR(100),
        IN p_stok INT
    )
BEGIN
    UPDATE buku
    SET Judul_Buku = p_judul, penulis = p_penulis, kategori = p_kategori, stok = p_stok
    WHERE id_buku = p_id_buku;
END //

```

DELETE

DELIMITER //

```

CREATE PROCEDURE deleteBuku(
    IN p_id_buku INT
)
BEGIN
    DELETE FROM buku WHERE ID_buku = p_id_buku;
END //

```

SISWA

UPDATE

DELIMITER //

```

CREATE PROCEDURE updateSiswa(
    IN p_id_siswa INT,
    IN p_nama VARCHAR(255),
    IN p_kelas VARCHAR(50)
)
BEGIN
    UPDATE siswa
    SET Nama = p_nama, Kelas = p_kelas
    WHERE ID_Siswa = p_id_siswa;
END //

```

DELETE

DELIMITER //

```

CREATE PROCEDURE deleteSiswa(
    IN p_id_siswa INT
)
BEGIN
    DELETE FROM siswa WHERE ID_Siswa = p_id_siswa;
END //

```

PEMINJAMAN

UPDATE

DELIMITER //

```
CREATE PROCEDURE updatePeminjaman(  
    IN p_id_peminjaman INT,  
    IN p_id_siswa INT,  
    IN p_id_buku INT,  
    IN p_tanggal_pinjam DATE,  
    IN p_tanggal_kembali DATE,  
    IN p_status VARCHAR(50)  
)  
BEGIN  
    UPDATE peminjaman  
    SET ID_Siswa = p_id_siswa, id_buku = p_id_buku,  
        Tanggal_Pinjam = p_tanggal_pinjam,  
        Tanggal_Kembali = p_tanggal_kembali,  
        Status = p_status  
    WHERE ID_Peminjaman = p_id_peminjaman;  
END //
```

DELETE

DELIMITER //

```
CREATE PROCEDURE deletePeminjaman(  
    IN p_id_peminjaman INT  
)  
BEGIN  
    DELETE FROM peminjaman WHERE ID_Peminjaman = p_id_peminjaman;  
END //
```

6. Buatlah stored procedure untuk menampilkan seluruh record di setiap table.

BUKU

DELIMITER //

```
CREATE PROCEDURE getAllBuku()  
BEGIN  
    SELECT * FROM buku;  
END //
```

CALL getAllBuku();

SISWA

DELIMITER //

```
CREATE PROCEDURE getAllSiswa()
```

```
BEGIN
    SELECT * FROM siswa;
END //
```

```
*CALL getAllSiswa();*
```

PEMINJAMAN

```
DELIMITER //
```

```
CREATE PROCEDURE getAllPeminjaman()
BEGIN
    SELECT * FROM peminjaman;
END //
```

```
*CALL getAllPeminjaman();*
```

7. Stok buku pada saat dipinjam berkurang secara otomatis.

```
DELIMITER //
```

```
CREATE PROCEDURE pinjamBuku(
    IN p_id_siswa INT,
    IN p_id_buku INT
)
BEGIN
    -- Periksa apakah stok buku masih tersedia
    IF (SELECT stok FROM buku WHERE id_buku = p_id_buku) > 0 THEN
        -- Tambahkan catatan peminjaman ke tabel 'peminjaman'
        INSERT INTO peminjaman (id_siswa, id_buku, tanggal_pinjam, status)
        VALUES (p_id_siswa, p_id_buku, CURDATE(), 'Dipinjam');

        -- Kurangi stok buku di tabel 'buku'
        UPDATE buku
        SET stok = stok - 1
        WHERE id_buku = p_id_buku;
    ELSE
        -- Jika stok habis, berikan pesan error
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Buku tidak tersedia!';
    END IF;
END //
```

```
*CALL pinjamBuku(6, 7);*
```

8. Stok buku pada saat dikembalikan bertambah secara otomatis.

```
DELIMITER //
```

```

CREATE PROCEDURE kembalikanBuku(
    IN p_id_peminjaman INT
)
BEGIN
    DECLARE v_id_buku INT;

    -- Ambil ID buku yang dikembalikan
    SELECT id_buku INTO v_id_buku FROM peminjaman WHERE id_peminjaman =
p_id_peminjaman;

    -- Update status peminjaman menjadi 'Dikembalikan' dan atur tanggal kembali ke hari ini
    UPDATE peminjaman
    SET status = 'Dikembalikan', tanggal_kembali = CURDATE()
    WHERE id_peminjaman = p_id_peminjaman;

    -- Tambahkan stok buku kembali ke tabel 'buku'
    UPDATE buku
    SET stok = stok + 1
    WHERE id_buku = v_id_buku;
END //

*CALL kembalikanBuku(3);*

```

9. Buatlah stored procedure untuk mengembalikan buku dan gunakan tanggal pengembalian sesuai dengan tanggal saat mengembalikan (CURRENT DATE).

DELIMITER //

```

CREATE PROCEDURE returnBuku(
    IN p_id_peminjaman INT
)
BEGIN
    UPDATE peminjaman
    SET tanggal_kembali = CURRENT_DATE,
        status = 'Dikembalikan'
    WHERE id_peminjaman = p_id_peminjaman;
END//

```

10. Buatlah stored procedure untuk menampilkan daftar siswa yang pernah meminjam buku.

DELIMITER //

```

CREATE PROCEDURE showSiswaPeminjam()
BEGIN
    SELECT DISTINCT s.id_siswa, s.nama, s.kelas
    FROM siswa s
    JOIN peminjaman p ON s.id_siswa = p.id_siswa;
END //

```


11. Buatlah stored procedure untuk menampilkan semua siswa, termasuk yang tidak pernah meminjam buku.

DELIMITER //

```
CREATE PROCEDURE showAllSiswa()
BEGIN
    SELECT s.id_siswa, s.nama, s.kelas,
           IFNULL(COUNT(p.id_peminjaman), 0) AS jumlah_peminjaman
    FROM siswa s
    LEFT JOIN peminjaman p ON s.id_siswa = p.id_siswa
    GROUP BY s.id_siswa, s.nama, s.kelas;
END;
```

12. Buatlah stored procedure untuk menampilkan semua buku, termasuk yang belum pernah dipinjam.

DELIMITER //

```
CREATE PROCEDURE showAllBuku()
BEGIN
    SELECT b.id_buku, b.judul_buku, b.penulis, b.kategori, b.stok,
           IFNULL(COUNT(p.id_peminjaman), 0) AS jumlah_dipinjam
    FROM buku b
    LEFT JOIN peminjaman p ON b.id_buku = p.id_buku
    GROUP BY b.id_buku, b.judul_buku, b.penulis, b.kategori, b.stok;
END;
```

~ **PUSH File SQL ke gitHub dengan nama repository Database-Pepustakaan-Sekolah** ~