

## 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.
2. Buatlah table buku, siswa dan peminjaman.
3. Input 5 record di setiap table menggunakan query INSERT.
4. Input 10 record di setiap table menggunakan stored procedure INSERT.
5. Buatlah stored procedure UPDATE, DELETE di setiap table.
6. Buatlah stored procedure untuk menampilkan seluruh record di setiap table.
7. Stok buku pada saat dipinjam berkurang secara otomatis.
8. Stok buku pada saat dikembalikan bertambah secara otomatis.
9. Buatlah stored procedure untuk mengembalikan buku dan gunakan tanggal pengembalian sesuai dengan tanggal saat mengembalikan (CURRENT DATE).
10. Buatlah stored procedure untuk menampilkan daftar siswa yang pernah meminjam buku.
11. Buatlah stored procedure untuk menampilkan semua siswa, termasuk yang tidak pernah meminjam buku.
12. Buatlah stored procedure untuk menampilkan semua buku, termasuk yang belum pernah dipinjam.

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

# QUERY DATABASE PERPUSTAKAAN

---

## 1. Membuat database db\_perpus

```
1 CREATE DATABASE db_perpus;
```

## 2. Membuat 3 table

```
1 CREATE TABLE buku(  
2     idBuku INT PRIMARY KEY AUTO_INCREMENT,  
3     judulBuku varchar(50),  
4     penulis varchar(50),  
5     kategori varchar(50),  
6     stok int);
```

```
1 CREATE TABLE siswa(  
2     idSiswa int primary key AUTO_INCREMENT,  
3     nama varchar(50),  
4     kelas varchar(20));
```

```
1 create table peminjaman(  
2     idPeminjaman int primary key AUTO_INCREMENT,  
3     idSiswa int,  
4     idBuku int,  
5     tglPinjam DATE,  
6     tglKembali DATE,  
7     status varchar(50));
```

Tables\_in\_db\_perpus

buku

peminjaman

siswa

Table Buku = Menyimpan data buku yang tersedia.

Table Siswa = Menyimpan data siswa yang terdaftar di database perpus.

Table Peminjaman = Menyimpan data riwayat peminjaman dari semua siswa.

### 3. Insert 5 data ke table

Table Buku

```
1 INSERT INTO buku (judulBuku, penulis, kategori, stok) VALUES
2 ("Algoritma dan Pemrograman", "Andi Wijaya", "Teknologi", 5),
3 ("Dasar-dasar Database", "Budi Santoso", "Teknologi", 7),
4 ("Matematika Diskrit", "Rina Sari", "Matematika", 4),
5 ("Sejarah Dunia", "John Smith", "Sejarah", 3),
6 ("Pemrograman Web dengan PHP", "Eko Prasetyo", "Teknologi", 8);|
```

Table Siswa

```
1 INSERT INTO siswa (nama, kelas) VALUES
2 ("Andi Saputra", "X-RPL"),
3 ("Budi Wijaya", "X-TKJ"),
4 ("Citra Lestari", "XI-RPL"),
5 ("Dewi Kurniawan", "XI-TKJ"),
6 ("Eko Prasetyo", "XII-RPL");|
```

Table Peminjaman

```
1 INSERT INTO peminjaman (idSiswa, idBuku, tglPinjam, tglKembali, statusPinjam) VALUES
2 (11, 2, "2025-02-01", "2025-02-08", "Dipinjam"),
3 (2, 5, "2025-01-28", "2025-02-04", "Dikembalikan"),
4 (3, 8, "2025-02-02", "2025-02-09", "Dipinjam"),
5 (4, 10, "2025-01-30", "2025-02-06", "Dikembalikan"),
6 (5, 3, "2025-01-25", "2025-02-01", "Dikembalikan");
```

Note : Tidak ada inputan untuk field id kecuali idSiswa dan idBuku di table peminjaman karena semua primary key bersifat auto\_increment

## 4. Insert 10 data ke masing-masing table menggunakan stored procedure

### Membuat procedure nya

#### Procedure Insert ke table buku

```
1 DELIMITER $$
2 CREATE PROCEDURE insert_buku(
3     pJudul varchar(50),
4     pPenulis varchar(50),
5     pKategori varchar(50),
6     pStok int)
7 BEGIN
8     INSERT INTO buku (judulBuku, penulis, kategori, stok) VALUES (pJudul, pPenulis, pKategori, pStok);
9 END $$
10 DELIMITER ;
```

#### Procedure insert ke table siswa

```
1 DELIMITER $$
2 CREATE PROCEDURE insert_siswa(
3     pNama varchar(50),
4     pKelas varchar(20))
5 BEGIN
6     INSERT INTO siswa (nama, kelas) VALUES (pNama, pKelas);
7 END $$
8 DELIMITER ;
```

#### Procedure insert ke table peminjaman

```
1 DELIMITER $$
2 CREATE PROCEDURE insert_peminjaman(
3     pIdSiswa int,
4     pIdBuku int,
5     pTglPinjam date,
6     pTglKembali date,
7     pStatus varchar(50))
8 BEGIN
9     INSERT INTO peminjaman (idSiswa, idBuku, tglPinjam, tglKembali, statusPinjam) VALUES
10     (pIdSiswa, pIdBuku, pTglPinjam, pTglKembali, pStatus);
11 END $$
12 DELIMITER ;
```

### Input data

Insert data ke table buku :

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

Insert data ke table siswa :

```
1 CALL insert_siswa('Farhan Maulana', 'XII-TKJ');
2 CALL insert_siswa('Gita Permata', 'X-RPL');
3 CALL insert_siswa('Hadi Sucipto', 'X-TKJ');
4 CALL insert_siswa('Intan Permadi', 'XI-RPL');
5 CALL insert_siswa('Joko Santoso', 'XI-TKJ');
6 CALL insert_siswa('Kartika Sari', 'XII-RPL');
7 CALL insert_siswa('Lintang Putri', 'XII-TKJ');
8 CALL insert_siswa('Muhammad Rizky', 'X-RPL');
9 CALL insert_siswa('Novi Andriana', 'X-TKJ');
10 CALL insert_siswa('Olivia Hernanda', 'XI-RPL');|
```

Insert data ke table Peminjaman

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

## 5. Membuat Stored Procedure Update & Delete

Procedure Update & Delete untuk table buku

```
1 DELIMITER $$
2 CREATE PROCEDURE update_buku(
3     pIdBuku INT,
4     pJudulBuku VARCHAR(50),
5     pPengarangBuku VARCHAR(50),
6     pKategoriBuku VARCHAR(50),
7     pStokBuku INT
8 )
9 BEGIN
10     UPDATE buku
11     SET judul = pJudulBuku, pengarang = pPengarangBuku, kategori = pKategoriBuku, stok = pStokBuku
12     WHERE id = pIdBuku;
13 END $$
14 DELIMITER ;
```

```
1 DELIMITER $$
2 CREATE PROCEDURE delete_buku(
3     pIdBuku INT
4 )
5 BEGIN
6     DELETE FROM buku WHERE id = pIdBuku;
7 END $$
8 DELIMITER ;
```

Procedure Update & Delete untuk table siswa

```
1 DELIMITER $$
2 CREATE PROCEDURE update_siswa(
3     pIdSiswa INT,
4     pNamaSiswa VARCHAR(50),
5     pKelasSiswa VARCHAR(50)
6 )
7 BEGIN
8     UPDATE siswa
9     SET nama = pNamaSiswa, kelas = pKelasSiswa
10    WHERE id = pIdSiswa;
11 END $$
12 DELIMITER ;
```

```
1 DELIMITER $$
2 CREATE PROCEDURE delete_siswa(
3     pIdSiswa INT
4 )
5 BEGIN
6     DELETE FROM siswa WHERE id = pIdSiswa;
7 END $$
8 DELIMITER ;
9
```

## Procedure Update & Delete untuk table peminjaman

```
1 DELIMITER $$
2 CREATE PROCEDURE update_peminjaman(
3     pIdPeminjaman INT,
4     pIdBuku INT,
5     pIdSiswa INT,
6     pTanggalPinjam DATE,
7     pTanggalKembali DATE,
8     pStatusPinjam VARCHAR(50)
9 )
10 BEGIN
11     UPDATE peminjaman
12     SET id_buku = pIdBuku, id_siswa = pIdSiswa, tanggal_pinjam = pTanggalPinjam, tanggal_kembali =
13     pTanggalKembali, status = pStatusPinjam
14     WHERE id = pIdPeminjaman;
15 END $$
```

```
1 DELIMITER $$
2 CREATE PROCEDURE delete_peminjaman(
3     pIdPeminjaman INT
4 )
5 BEGIN
6     DELETE FROM peminjaman WHERE id = pIdPeminjaman;
7 END $$
8 DELIMITER ;
```

## 6. Membuat Procedure untuk menampilkan semua data

```
1 DELIMITER $$
2 CREATE PROCEDURE select_buku()
3 BEGIN
4     SELECT idBuku AS 'ID Buku',
5     judulBuku AS 'Judul Buku',
6     penulis AS 'Penulis',
7     kategori AS 'Kategori',
8     stok AS 'Stok'
9     FROM buku;
10 END $$
11 DELIMITER ;
```

```
1 DELIMITER $$
2 CREATE PROCEDURE select_siswa()
3 BEGIN
4     SELECT idSiswa AS 'ID Siswa',
5     nama AS 'Nama',
6     kelas AS 'Kelas'
7     FROM siswa;
8 END $$
9 DELIMITER ;
```



```

1 DELIMITER $$
2 CREATE PROCEDURE select_buku()
3 BEGIN
4     SELECT idPeminjaman AS 'ID Peminjaman',
5     idSiswa AS 'ID Siswa',
6     idBuku AS 'ID Buku',
7     tglPinjam AS 'Tanggal Pinjam',
8     tglKembali AS 'Tanggal Kembali',
9     statusPinjam AS 'Status'
10    FROM peminjaman;
11 END $$
12 DELIMITER ;

```

Note : Tujuan penggunaan “AS” supaya saat table ditampilkan, nama field atau kolom lebih rapih dan sama seperti contoh.

## 7. Trigger supaya stok buku yang dipinjam akan berkurang secara otomatis

```

1 DELIMITER $$
2 CREATE TRIGGER after_insert_peminjaman
3 AFTER INSERT ON peminjaman
4 FOR EACH ROW
5 BEGIN
6     IF NEW.statusPinjam = 'Dipinjam' THEN
7         UPDATE buku
8         SET stok = stok - 1
9         WHERE id = NEW.idBuku;
10    END IF;
11 END $$
12 DELIMITER ;

```

## 8. Trigger supaya stok buku kembali bertambah saat dikembalikan

```

1 DELIMITER $$
2 CREATE TRIGGER after_update_peminjaman
3 AFTER UPDATE ON peminjaman
4 FOR EACH ROW
5 BEGIN
6     IF NEW.statusPinjam = 'Dikembalikan' AND OLD.statusPinjam = 'Dipinjam' THEN
7         UPDATE buku
8         SET stok = stok + 1
9         WHERE id = NEW.idBuku;
10    END IF;
11 END $$
12 DELIMITER ;

```

## 9. Membuat Procedure untuk mengembalikan buku dan menggunakan tanggal sekarang

```
1 DELIMITER $$
2 CREATE PROCEDURE kembalikanBuku(
3     pIdPeminjaman INT
4 )
5 BEGIN
6     UPDATE peminjaman
7     SET status = 'Dikembalikan', tglKembali = CURRENT_DATE
8     WHERE idPeminjaman = pIdPeminjaman;
9 END $$
10 DELIMITER ;
```

## 10. Membuat Procedure untuk menampilkan semua siswa yang pernah meminjam buku

```
1 DELIMITER $$
2 CREATE PROCEDURE selectAllSiswa()
3 BEGIN
4     SELECT DISTINCT s.idSiswa AS 'ID Siswa',
5     s.nama AS 'Nama',
6     s.kelas AS 'Kelas'
7     FROM siswa s
8     INNER JOIN peminjaman p ON s.idSiswa = p.idSiswa;
9 END $$
10 DELIMITER ;
```

## 11. Membuat procedure untuk menampilkan semua siswa baik yang pernah maupun belum pernah meminjam buku

```
1 DELIMITER $$
2 CREATE PROCEDURE selectAllSiswa()
3 BEGIN
4     SELECT s.idSiswa AS 'ID Siswa',
5     s.nama AS 'Nama',
6     s.kelas AS 'Kelas',
7     p.idPeminjaman AS 'ID Peminjaman',
8     p.tglPinjam AS 'Tanggal Pinjam',
9     p.tglKembali AS 'Tanggal Kembali',
10    p.statusPinjam AS 'Status'
11    FROM siswa s
12    LEFT JOIN peminjaman p ON s.idSiswa = p.idSiswa;
13 END $$
14 DELIMITER ;
```

## 12. Menampilkan semua buku baik yang pernah ataupun belum pernah dipinjam

```
1 DELIMITER $$
2 CREATE PROCEDURE selectAllBuku()
3 BEGIN
4     SELECT
5         b.idBuku as 'ID Buku',
6         b.judulBuku as 'Judul Buku',
7         b.penulis as 'Penulis',
8         b.kategori as 'Kategori',
9         b.stok as 'Stok',
10        IFNULL(p.idPeminjaman, 'Belum dipinjam') AS 'ID Peminjaman',
11        IFNULL(p.tglPinjam, 'Belum dipinjam') as 'Tanggal Pinjam',
12        IFNULL(p.statusPinjam, 'Belum dipinjam') as 'Status'
13    FROM buku b
14    LEFT JOIN peminjaman p ON b.idBuku = p.idBuku;
15 END $$
16 DELIMITER ;
```