

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 ~

~ Jawaban Query ~

1. Membuat Database dengan nama db_perpus

```
CREATE DATABASE db_perpus;
```

2. Membuat 3 table (table buku, siswa, dan peminjaman)

```
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   ID_Siswa int PRIMARY KEY AUTO_INCREMENT,  
3   Nama varchar(50),  
4   Kelas varchar(10));
```

```
CREATE TABLE Peminjaman(  
  ID_Peminjaman int PRIMARY KEY AUTO_INCREMENT,  
  ID_Siswa int,  
  IDBuku int,  
  Tanggal_Pinjam DATE,  
  Tanggal_Kembali DATE,  
  Status varchar(20));
```

3. Insert 5 data ke table

Table buku

```
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);
```

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 Kurnia", "XI-TKJ"),
6 ("Eko Prasetyo", "XII-RPL");
```

Table peminjaman

```
1 INSERT INTO peminjaman (ID_Siswa, IDBuku, Tanggal_Pinjam, Tanggal_Kembali, Status) 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");
```

4. Insert 10 data ke masing masing table menggunakan stored procedure Membuat procedure insert pada table buku

```
1 CREATE PROCEDURE insert_buku(
2     pJudul varchar(50),
3     pPenulis varchar(50),
4     pKategori varchar(50),
5     pStok int)
6 BEGIN
7     INSERT into buku (Judul_Buku, Penulis, Kategori, Stok) VALUES (pJudul, pPenulis, pKategori,
8     pStok);
9 END;
```

(Note : Saya menggunakan delimiter ketika menginput ini tapi entah kenapa di ss an ini hilang)

Input data ke dalam 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);
```

Membuat procedure insert pada table siswa

```
1 DELIMITER $$
2 CREATE PROCEDURE insert_siswa(
3     pNama varchar(50),
4     pKelas varchar(10))
5 BEGIN
6     INSERT INTO siswa (Nama, Kelas) values (pNama, pKelas);
7 END $$
8 DELIMITER ;
```

Input data kedalam 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');
```

Membuat procedure insert pada table peminjaman

```
1 DELIMITER $$
2 CREATE PROCEDURE insert_peminjaman(
3     pIdSiswa int,
4     pIdBuku int,
5     pTglPinjam date,
6     pTglKembali date,
7     pStatus varchar(50))
8 BEGIN
9     insert INTO peminjaman (ID_Siswa, IDBuku, Tanggal_Pinjam, Tanggal_Kembali, Status) VALUES
10     (pIdSiswa, pIdBuku, pTglPinjam, pTglKembali, pStatus);
11 END $$
12 DELIMITER ;
```

Input data kedalam 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 pada table buku

```
1 DELIMITER $$
2 CREATE PROCEDURE update_buku(
3     IDBuku int,
4     JudulBuku varchar(50),
5     PenulisBuku varchar(50),
6     KategoriBuku varchar(50),
7     StokBuku int)
8 BEGIN
9     UPDATE buku
10     SET Judul_Buku = JudulBuku, Penulis = PenulisBuku, Kategori = KategoriBuku, Stok = StokBuku
11     WHERE ID_Buku = IDBuku;
12 END $$
13 DELIMITER ;
```

Procedured Delete pada table buku

```

1 DELIMITER $$
2 CREATE PROCEDURE delete_buku(
3     IDBuku int
4 )
5 BEGIN
6     DELETE FROM buku WHERE ID_Buku = IDBuku;
7 END $$
8 DELIMITER ;

```

Procedure Update pada table siswa

```

1 DELIMITER $$
2 CREATE PROCEDURE update_siswa(
3     IDSiswa int,
4     NamaSiswa varchar(50),
5     KelasSiswa varchar(50)
6 )
7 BEGIN
8     UPDATE siswa
9     SET Nama = NamaSiswa, Kelas = KelasSiswa
10    WHERE ID_Siswa = IDSiswa;
11 END $$
12 DELIMITER ;

```

Procedure Delete pada table siswa

```

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

```

Procedure Update pada table peminjaman

```

1 DELIMITER $$
2 CREATE PROCEDURE update_peminjaman(
3     IDPeminjaman int,
4     IDSiswa int,
5     IDBuku int,
6     TglPinjam date,
7     TglKembali date,
8     StatusPinjam varchar(50)
9 )
10 BEGIN
11     UPDATE peminjaman
12     SET ID_Siswa = IDSiswa, IDBuku = IDBuku, Tanggal_Pinjam = TglPinjam, Tanggal_Kembali = TglKembali, Status = StatusPinjam
13    WHERE ID_Peminjaman = IDPeminjaman;
14 END $$
15 DELIMITER ;

```

Procedure Delete pada table peminjaman

```

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

```

6. Membuat procedure untuk menampilkan semua data

```
1 DELIMITER $$
2 CREATE PROCEDURE tampilkan_buku()
3 BEGIN
4     SELECT * FROM buku;
5 END $$
6 DELIMITER ;
```

```
1 DELIMITER $$
2 CREATE PROCEDURE tampilkan_siswa()
3 BEGIN
4     SELECT * FROM siswa;
5 END $$
6 DELIMITER ;
```

```
1 DELIMITER $$
2 CREATE PROCEDURE tampilkan_peminjaman()
3 BEGIN
4     SELECT * FROM peminjaman;
5 END $$
6 DELIMITER ;
```

7. Membuat trigger supaya stok buku yang di pinjam berkurang secara otomatis

```
1 DELIMITER $$
2 CREATE TRIGGER after_insert_peminjaman
3 AFTER INSERT ON peminjaman
4 FOR EACH ROW
5 BEGIN
6     UPDATE buku
7     SET Stok = Stok - 1
8     WHERE IDBuku = NEW.IDBuku;
9 END $$
10 DELIMITER ;
```

8. Membuat Trigger supaya stok buku kembali bertambah saat dikembalikan

```
1 DELIMITER $$
2 CREATE TRIGGER after_update_peminjaman
3 AFTER UPDATE ON peminjaman
4 FOR EACH ROW
5 BEGIN
6     IF NEW.Status = 'Dikembalikan'
7     THEN UPDATE buku SET Stok = Stok + 1
8     WHERE IDBuku = NEW.IDBuku;
9 END IF;
10 END $$
11 DELIMITER ;
```

9. **Membuat Procedure untuk mengembalikan buku dan menggunakan tanggal yang sekarang**

```
1 DELIMITER $$
2 CREATE PROCEDURE kembalikanbuku(
3     IDPeminjaman int
4 )
5 BEGIN
6     UPDATE peminjaman
7     SET Status = 'Dikembalikan', Tanggal_Kembali = CURRENT_DATE
8     WHERE ID_Peminjaman = IDPeminjaman;
9 END $$
10 DELIMITER ;
```

10. **Membuat procedure untuk menampilkan semua siswa yang pernah meminjam buku**

```
1 DELIMITER $$
2 CREATE PROCEDURE SiswaPernahMinjam()
3 BEGIN
4     SELECT DISTINCT s.ID_Siswa, s>Nama, s.Kelas
5     FROM siswa s
6     INNER JOIN peminjaman p ON s.ID_Siswa = p.ID_Siswa;
7 END $$
8 DELIMITER ;
```

11. **Membuat procedure untuk menampilkan semua siswa (Termasuk yang tidak pernah meminjam)**

```
1 DELIMITER $$
2 CREATE PROCEDURE SemuaSiswa()
3 BEGIN
4     SELECT s.ID_Siswa, s>Nama, s.Kelas,
5     CASE
6         WHEN p.ID_Siswa IS NULL THEN 'Belum Pernah Meminjam'
7         ELSE 'Pernah Meminjam'
8     END AS Status
9     FROM siswa s
10    LEFT JOIN peminjaman p ON s.ID_Siswa = p.ID_Siswa
11    GROUP BY s.ID_Siswa;
12 END $$
13 DELIMITER ;
```

12. **Membuat procedure untuk menampilkan semua buku (Termasuk yang belum pernah di pinjam)**

```
1 DELIMITER $$
2 CREATE PROCEDURE SemuaBuku()
3 BEGIN
4     SELECT b.ID_Buku, b.Judul_Buku, b.Penulis, b.Kategori, b.Stok,
5     CASE
6         WHEN p.IDBuku IS NULL THEN 'Belum Pernah Dipinjam'
7         ELSE 'Pernah Dipinjam'
8     END AS Status
9     FROM buku b
10    LEFT JOIN peminjaman p ON b.ID_Buku = p.IDBuku
11    GROUP BY b.ID_Buku;
12 END $$
13 DELIMITER ;
```