

## 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.

### Databases

Create database ?

Create

2. Buatlah table buku, siswa dan peminjaman.

Create new table

Table name

Number of columns

Create

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 id_buku	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	2 judul_buku	varchar(35)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/>	3 penulis	varchar(35)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/>	4 kategori	varchar(35)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/>	5 stok	int(5)			No	None			Change  Drop  More

## Tabel siswa

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 <b>id_siswa</b>	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	2 <b>nama</b>	varchar(35)	utf8mb4_general_ci		No	None			Change  Drop  More
<input type="checkbox"/>	3 <b>kelas</b>	varchar(35)	utf8mb4_general_ci		No	None			Change  Drop  More

## Tabel Peminjaman

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1 <b>id_peminjaman</b>	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	2 <b>id_siswa</b>	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	3 <b>id_buka</b>	int(11)			No	None			Change  Drop  More
<input type="checkbox"/>	4 <b>tanggal_pinjam</b>	date			No	None			Change  Drop  More
<input type="checkbox"/>	5 <b>tanggal_kelas</b>	date			No	None			Change  Drop  More
<input type="checkbox"/>	6 <b>status</b>	varchar(35)	utf8mb4_general_ci		No	None			Change  Drop  More

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

### INSERT BUKU

INSERT INTO buku (id\_Buku, judul\_buku, penulis, kategori, stok) VALUES

- (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);

### INSERT SISWA

INSERT INTO siswa (id\_siswa, nama, kelas) VALUES

- (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');

## INSERT PEMINJAMAN

```
INSERT INTO Peminjaman (id_peminjaman, id_siswa, id_buka, tanggal_pinjam, tanggal_kelas,status)VALUES
(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');
```

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

## TABLE BUKU

DELIMITER //

CREATE PROCEDURE insertbuku(

IN id\_buku int(11),

IN judul\_buku varchar(35),

IN penulis varchar(35),

IN kategori varchar(35),

IN stok int (5)

)

BEGIN

INSERT INTO buku(id\_buku, judul\_buku, penulis, kategori, stok)

VALUES (id\_buku, judul\_buku, penulis, kategori, stok);

END //

CALL insertbuku(6, 'Sistem Operasi', 'Dian Kurniawan', 'Teknologi', 6);

CALL insertbuku(7, 'Jaringan Komputer', 'Ahmad Fauzi', 'Teknologi', 5);

CALL insertbuku(8, 'Cerita Rakyat Nusantara', 'Lestari Dewi', 'Sastra', 9);

CALL insertbuku(9, 'Bahasa Inggris untuk Pemula', 'Jane Doe', 'Bahasa', 10);

CALL insertbuku(10, 'Biologi Dasar', 'Budi Rahman', 'Sains', 7);

CALL insertbuku(11, 'Kimia Organik', 'Siti Aminah', 'Sains', 5);

CALL insertbuku(12, 'Teknik Elektro', 'Ridwan Hakim', 'Teknik', 6);

```
CALL insertbuku(13, 'Fisika Modern', 'Albert Einstein', 'Sains', 8);
```

```
CALL insertbuku(14, 'Manajemen Waktu', 'Steven Covey', 'Pengembangan', 8);
```

```
CALL insertbuku(15, 'Strategi Belajar Efektif', 'Tony Buzan', 'Pendidikan', 6);
```

#### **TABEL SISWA**

```
DELIMITER //
```

```
CREATE PROCEDURE insertsiswa(
```

```
IN id_siswa int(11),
```

```
IN nama varchar(35),
```

```
IN kelas varchar(35)
```

```
)
```

```
BEGIN
```

```
INSERT INTO siswa(id_siswa, nama, kelas)
```

```
VALUES (id_siswa, nama, kelas);
```

```
END //
```

```
CALL insertsiswa(6, 'Farhan Maulana', 'XII-TKJ');
```

```
CALL insertsiswa(7, 'Gita Permata', 'X-RPL');
```

```
CALL insertsiswa(8, 'Hadi Sucipto', 'X-TKJ');
```

```
CALL insertsiswa(9, 'Intan Permadi', 'XI-RPL');
```

```
CALL insertsiswa(10, 'Joko Santoso', 'XI-TKJ');
```

```
CALL insertsiswa(11, 'Kartika Sari', 'XII-RPL');
```

```
CALL insertsiswa(12, 'Lintang Putri', 'XII-TKJ');
```

```
CALL insertsiswa(13, 'Muhammad Rizky', 'X-RPL');
```

```
CALL insertsiswa(14, 'Novi Andriana', 'X-TKJ');
```

```
CALL insertsiswa(15, 'Olivia Hernanda', 'XI-RPL');
```

#### **TABEL PEMINJAMAN**

```
DELIMITER //
```

```
CREATE PROCEDURE insertpeminjaman(
```

```
IN id_peminjaman int(11),
```

```
IN id_siswa int(11),
```

```
IN id_buku int(11),
```

```

IN tanggal_pinjam date,
IN tanggal_kembali date,
IN status varchar(35)
)
BEGIN
INSERT INTO
peminjaman(id_peminjaman,id_siswa,id_buku,tanggal_pinjam,tanggal_kembali,status)
VALUES (id_peminjaman,id_siswa,id_buku,tanggal_pinjam,tanggal_kembali,status);
END //

```

```

CALL insertpeminjaman(6, 15, 7, '2025-02-01', '2025-02-08', 'Dipinjam');
CALL insertpeminjaman(7, 7, 1, '2025-01-29', '2025-02-05', 'Dikembalikan');
CALL insertpeminjaman(8, 8, 9, '2025-02-03', '2025-02-10', 'Dipinjam');
CALL insertpeminjaman(9, 13, 4, '2025-01-27', '2025-02-03', 'Dikembalikan');
CALL insertpeminjaman(10, 10, 11, '2025-02-01', '2025-02-08', 'Dipinjam');

```

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

#### **TABLE BUKU UPDATE**

```

DELIMITER //
CREATE PROCEDURE updatebuku(
    IN id_buku_baru int(11),
    IN judul_buku_baru varchar(35),
    IN penulis_baru varchar(35),
    IN kategori_baru varchar(35),
    IN stok_baru int (5)
)
BEGIN
    UPDATE buku
    SET judul_buku = judul_buku_baru, penulis = penulis_baru, kategori = kategori_baru
    WHERE id_buku = id_buku_baru;
END //

```

DELIMITER ;

#### **TABLE BUKU DELETE**

DELIMITER //

CREATE PROCEDURE DELETEbuku(IN id\_buku\_baru int(11))

BEGIN

DELETE FROM buku WHERE id\_buku = id\_buku\_baru;

END //

DELIMITER ;

#### **TABLE SISWA UPDATE**

DELIMITER //

CREATE PROCEDURE updatesiswa(

IN id\_siswa\_baru int(11),

IN nama\_baru varchar(35),

IN kelas\_baru varchar(35)

)

BEGIN

UPDATE siswa

SET nama = nama\_baru, kelas = kelas\_baru

WHERE id\_siswa = id\_siswa\_baru;

END //

DELIMITER ;

#### **TABLE SISWA DELETE**

DELIMITER //

CREATE PROCEDURE DELETEsiswa(IN id\_siswa\_baru int(11))

BEGIN

DELETE FROM buku WHERE id\_siswa = id\_siswa\_baru;

END //

DELIMITER ;

## **TABLE PEMINJAMAN UPDATE**

DELIMITER //

CREATE PROCEDURE updatespeminjaman(

IN id\_peminjaman\_baru int(11),

IN id\_siswa\_baru int(11),

IN id\_buku\_baru int(11),

IN tanggal\_pinjam\_baru date,

IN tanggal\_kembali\_baru date,

IN status\_baru varchar(35)

)

BEGIN

UPDATE peminjamann

SET id\_siswa = id\_siswa\_baru, id\_buku = id\_buku\_baru, tanggal\_pinjam =  
tanggal\_pinjam\_baru, tanggal\_kembali = tanggal\_kembali\_baru, status = status\_baru

WHERE id\_peminjaman = id\_peminjaman\_baru;

END //

DELIMITER ;

## **TABLE PEMINJAMAN DELETE**

DELIMITER //

CREATE PROCEDURE DELETEpeminjaman(IN id\_peminjaman\_baru int(11))

BEGIN

DELETE FROM buku WHERE id\_peminjaman = id\_peminjaman\_baru;

END //

DELIMITER ;



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

**TABLE BUKU**

DELIMITER //

CREATE PROCEDURE selectbuku()

BEGIN

SELECT \* FROM buku;

END; //

**TABLE SISWA**

DELIMITER //

CREATE PROCEDURE selectsiswa()

BEGIN

SELECT \* FROM siswa;

END; //

**TABLE PEMINJAMAN**

DELIMITER //

CREATE PROCEDURE selectpeminjaman()

BEGIN

SELECT \* FROM peminjaman;

END; //

**7. Stok buku pada saat dipinjam berkurang secara otomatis.**

DELIMITER \$\$

CREATE TRIGGER kurangi\_stok\_buku

AFTER INSERT ON peminjaman

FOR EACH ROW

BEGIN

UPDATE buku

SET stok = stok - 1

WHERE id\_buku = NEW.id\_buku;

END \$\$

DELIMITER ;

SHOW TRIGGERS FROM db\_perpus;

**8. Stok buku pada saat dikembalikan bertambah secara otomatis.**

DELIMITER \$\$

CREATE TRIGGER tambah\_stok\_buku

AFTER INSERT ON peminjaman

FOR EACH ROW

BEGIN

    UPDATE buku

    SET stok = stok + 1

    WHERE id\_buku = (SELECT id\_buku FROM peminjaman WHERE id\_peminjaman = NEW.id\_peminjaman);

END \$\$

DELIMITER ;

SHOW TRIGGERS FROM db\_perpus;

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

DELIMITER //

CREATE PROCEDURE KembalikanBuku(

    IN IdPeminjamanBaru INT

)

BEGIN

    UPDATE peminjaman

    SET tanggal\_kembali = CURDATE(), status = 'Dikembalikan'

    WHERE id\_peminjaman = IdPeminjamanBaru;

END; //

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

DELIMITER //

```

CREATE PROCEDURE DaftarSiswaPeminjam()
BEGIN
    SELECT DISTINCT siswa.id_siswa, siswa.nama, siswa.kelas
    FROM siswa
    JOIN peminjaman ON siswa.id_siswa = peminjaman.id_siswa
    WHERE peminjaman.status IN ('dipinjam','dikembalikan');
END //

DELIMITER ;

CALL DaftarSiswaPeminjam();

```

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

```

DELIMITER //

CREATE PROCEDURE SemuaSiswaPeminjam()
BEGIN
    SELECT siswa.id_siswa, siswa.nama, siswa.kelas,
           COALESCE(peminjaman.status, 'Belum Pernah Meminjam') AS status_peminjaman
    FROM siswa
    LEFT JOIN peminjaman ON siswa.id_siswa = peminjaman.id_siswa
    ORDER BY siswa.nama;
END //

DELIMITER ;

CALL SemuaSiswaPeminjam();

```

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

```

DELIMITER //

CREATE PROCEDURE SemuaBukuPerpus()
BEGIN

```

```
SELECT buku.id_buku, buku.judul_buku, buku.penulis, buku.kategori, buku.stok,  
CASE  
    WHEN peminjaman.id_peminjaman IS NULL THEN 'Belum Pernah Dipinjam'  
    ELSE 'Pernah Dipinjam'  
END AS status_peminjaman  
FROM buku  
LEFT JOIN peminjaman ON buku.id_buku = peminjaman.id_buku  
GROUP BY buku.id_buku, buku.judul_buku, buku.penulis, buku.kategori, buku.stok  
ORDER BY buku.judul_buku;  
END //  
DELIMITER ;  
  
CALL SemuaBukuPerpus();  
~ PUSH File SQL ke gitHub dengan nama repository Database-Pepustakaan-Sekolah ~
```