

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

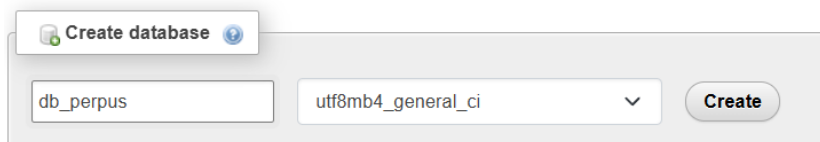
3. Tabel Peminjaman

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



2. Buatlah table buku, siswa dan peminjaman.

```
CREATE TABLE buku (  
  id_buku INT PRIMARY KEY AUTO_INCREMENT,  
  judul_buku VARCHAR(255),  
  penulis VARCHAR(255),  
  kategori VARCHAR(100),  
  stok INT  
);  
  
CREATE TABLE siswa (  
  id_siswa INT PRIMARY KEY AUTO_INCREMENT,  
  nama VARCHAR(255),  
  kelas VARCHAR(50)  
);  
  
CREATE TABLE peminjaman (  
  id_peminjaman INT PRIMARY KEY AUTO_INCREMENT,  
  id_siswa INT,  
  id_buku INT,  
  tanggal_pinjam DATE,  
  tanggal_kembali DATE,  
  status ENUM('Dipinjam', 'Dikembalikan'),  
  FOREIGN KEY (id_siswa) REFERENCES siswa(id_siswa),  
  FOREIGN KEY (id_buku) REFERENCES buku(id_buku)  
);
```

3. Input 5 record di setiap table menggunakan query INSERT.

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

```
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');
```

Table peminjaman

```
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.

Insert Buku :

```
DELIMITER //
CREATE PROCEDURE insertbuku(
    IN judul_buku varchar(255),
    IN penulis varchar(255),
    IN kategori varchar(100),
    IN stok int
)
BEGIN
    INSERT INTO buku (judul_buku, penulis, kategori, stok) VALUES
    (judul_buku, penulis, kategori, stok);
END //

>CALL insertbuku('Jaringan Komputer', 'Ahmad Fauzi', 'Teknologi', 5);
CALL insertbuku('Sistem Operasi', 'Dian Kurniawan', 'Teknologi', 6);
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('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);
```

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

Insert Siswa :

DELIMITER //

CREATE PROCEDURE insertsiswa(

IN nama varchar(255),

IN kelas varchar(50),

)

BEGIN

INSERT INTO siswa (nama, kelas) VALUES

(nama,kelas);

END //

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');

				id_siswa	nama	kelas
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	Andi Saputra	X-RPL
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	Budi Wijaya	X-TKJ
<input type="checkbox"/>	 Edit	 Copy	 Delete	3	Citra Lestari	XI-RPL
<input type="checkbox"/>	 Edit	 Copy	 Delete	4	Dewi Kurniawan	XI-TKJ
<input type="checkbox"/>	 Edit	 Copy	 Delete	5	Eko Prasetyo	XII-RPL
<input type="checkbox"/>	 Edit	 Copy	 Delete	6	Farhan Maulana	XII-TKJ
<input type="checkbox"/>	 Edit	 Copy	 Delete	7	Gita Permata	X-RPL
<input type="checkbox"/>	 Edit	 Copy	 Delete	8	Hadi Sucipto	X-TKJ
<input type="checkbox"/>	 Edit	 Copy	 Delete	9	Intan Permadi	XI-RPL
<input type="checkbox"/>	 Edit	 Copy	 Delete	10	Joko Santoso	XI-TKJ
<input type="checkbox"/>	 Edit	 Copy	 Delete	11	Kartika Sari	XII-RPL
<input type="checkbox"/>	 Edit	 Copy	 Delete	12	Lintang Putri	XII-TKJ
<input type="checkbox"/>	 Edit	 Copy	 Delete	13	Muhammad Rizky	X-RPL
<input type="checkbox"/>	 Edit	 Copy	 Delete	14	Novi Andriana	X-TKJ
<input type="checkbox"/>	 Edit	 Copy	 Delete	15	Olivia Hernanda	XI-RPL

Insert Peminjaman :

DELIMITER //

CREATE PROCEDURE InsertPeminjaman2(

IN Id_siswa int,

IN Id_buku int,

IN Tanggal_pinjam date,

IN Tanggal_kembali date,

IN Status ENUM('Dipinjam','Dikembalikan')

```

)
BEGIN
    INSERT INTO peminjaman (id_siswa, id_buku, tanggal_pinjam, tanggal_kembali, status) VALUES
        (Id_siswa, Id_buku, Tanggal_pinjam, Tanggal_kembali, Status);
END //
CALL insertPeminjaman2(15, 7, '2025-02-01', '2025-02-08', 'Dipinjam');
CALL insertPeminjaman2(7, 1, '2025-01-29', '2025-02-05', 'Dikembalikan');
CALL insertPeminjaman2(8, 9, '2025-02-03', '2025-02-10', 'Dipinjam');
CALL insertPeminjaman2(13, 4, '2025-01-27', '2025-02-03', 'Dikembalikan');
CALL insertPeminjaman2(10, 11, '2025-02-01', '2025-02-08', 'Dipinjam');

```

	id_peminjaman	id_siswa	id_buku	tanggal_pinjam	tanggal_kembali	status
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	11	2	2025-02-01	2025-02-08	Dipinjam
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	2	5	2025-01-28	2025-02-04	Dikembalikan
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	3	8	2025-02-02	2025-02-09	Dipinjam
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	4	10	2025-01-30	2025-02-06	Dikembalikan
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	5	3	2025-01-25	2025-02-01	Dikembalikan
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	15	7	2025-02-01	2025-02-08	Dipinjam
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	7	1	2025-01-29	2025-02-05	Dikembalikan
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	8	9	2025-02-03	2025-02-10	Dipinjam
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	9	13	4	2025-01-27	2025-02-03	Dikembalikan
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	10	11	2025-02-01	2025-02-08	Dipinjam

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

UPDATE

DELIMITER //

```
CREATE PROCEDURE updateBuku(
```

```
    IN buku_id INT,
```

```
    IN new_stok INT
```

```
)
```

```
BEGIN
```

```
    UPDATE buku SET stok = new_stok WHERE id_buku = buku_id;
```

```
END //
```

```
CREATE PROCEDURE updateSiswa(
```

```
    IN siswa_id INT,
```

```
    IN new_kelas VARCHAR(50)
```

```
)
```

```
BEGIN
```

```
    UPDATE siswa SET kelas = new_kelas WHERE id_siswa = siswa_id;
```

```
END //
```

```
CREATE PROCEDURE updatePeminjaman(
```

```
    IN pinjam_id INT,
```

```
    IN new_status ENUM('Dipinjam', 'Dikembalikan')
```

```
)
```

```
BEGIN
```

```
    UPDATE peminjaman SET status = new_status WHERE id_peminjaman = pinjam_id;
```

```
END //
```

DELETE

DELIMITER //

```
CREATE PROCEDURE deleteBuku(IN buku_id INT)
```

```
BEGIN
```

```
    DELETE FROM buku WHERE id_buku = buku_id;
```

```
END //
```

```
CREATE PROCEDURE deleteSiswa(IN siswa_id INT)
```

```
BEGIN
```

```
    DELETE FROM siswa WHERE id_siswa = siswa_id;
```

```
END //
```

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

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

```
DELIMITER //
CREATE PROCEDURE semuabuku()
BEGIN SELECT * FROM buku;
END //
CALL semuabuku();
```

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

```
DELIMITER //
CREATE PROCEDURE semuasiswa()
BEGIN SELECT * FROM siswa;
END //
```

```
>CALL semuasiswa();
```

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

7. Stok buku pada saat dipinjam berkurang secara otomatis.

```
DELIMITER //
CREATE TRIGGER berkurangstokbuku
BEFORE INSERT ON peminjaman
FOR EACH ROW
BEGIN UPDATE buku SET stok = stok - 1
WHERE id_buku = NEW.id_buku;
END //
```

8. Stok buku pada saat dikembalikan bertambah secara otomatis.

```
DELIMITER //
CREATE TRIGGER TambahStokBuku
AFTER UPDATE ON peminjaman
FOR EACH ROW
BEGIN IF
NEW.status = 'Dikembalikan' THEN
UPDATE buku SET stok = stok + 1 WHERE id_buku = NEW.id_buku;
END IF;
END //
```

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

```
DELIMITER //

CREATE PROCEDURE kembalikanBuku(IN p_id_peminjaman INT)
BEGIN

    UPDATE peminjaman
    SET status = 'Dikembalikan',
        tanggal_kembali = CURRENT_DATE
    WHERE id_peminjaman = p_id_peminjaman;

    UPDATE buku
    SET stok = stok + 1
    WHERE id_buku = (SELECT id_buku FROM peminjaman WHERE id_peminjaman = p_id_peminjaman);
END //

CALL kembalikanBuku(1);
```

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

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

CALL SiswaPernahPinjam();
```

id_siswa	nama	kelas
2	Budi Wijaya	X-TKJ
3	Citra Lestari	XI-RPL
4	Dewi Kurniawan	XI-TKJ
5	Eko Prasetyo	XII-RPL
7	Gita Permata	X-RPL
8	Hadi Sucipto	X-TKJ
10	Joko Santoso	XI-TKJ
11	Kartika Sari	XII-RPL
13	Muhammad Rizky	X-RPL
15	Olivia Hernanda	XI-RPL

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

```
DELIMITER //
```

```
CREATE PROCEDURE semuamurid()
BEGIN
    SELECT s.id_siswa, s.nama, s.kelas, COALESCE(p.id_peminjaman, 'Belum Meminjam') AS Status_Peminjaman
    FROM siswa s
    LEFT JOIN peminjaman p ON s.id_siswa = p.id_siswa;
END //
```

```
CALL semuamurid();
```

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

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

```
DELIMITER //
CREATE PROCEDURE showsemuabuku()
BEGIN
    SELECT b.id_buku, b.judul_buku, b.penulis, b.kategori, b.stok,
    COALESCE(p.id_peminjaman, 'Belum Pernah Dipinjam') AS Status_Peminjaman
    FROM buku b
    LEFT JOIN peminjaman p ON b.id_buku = p.id_buku;
END //
```

```
CALL showsemuabuku();
```

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

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