

# 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

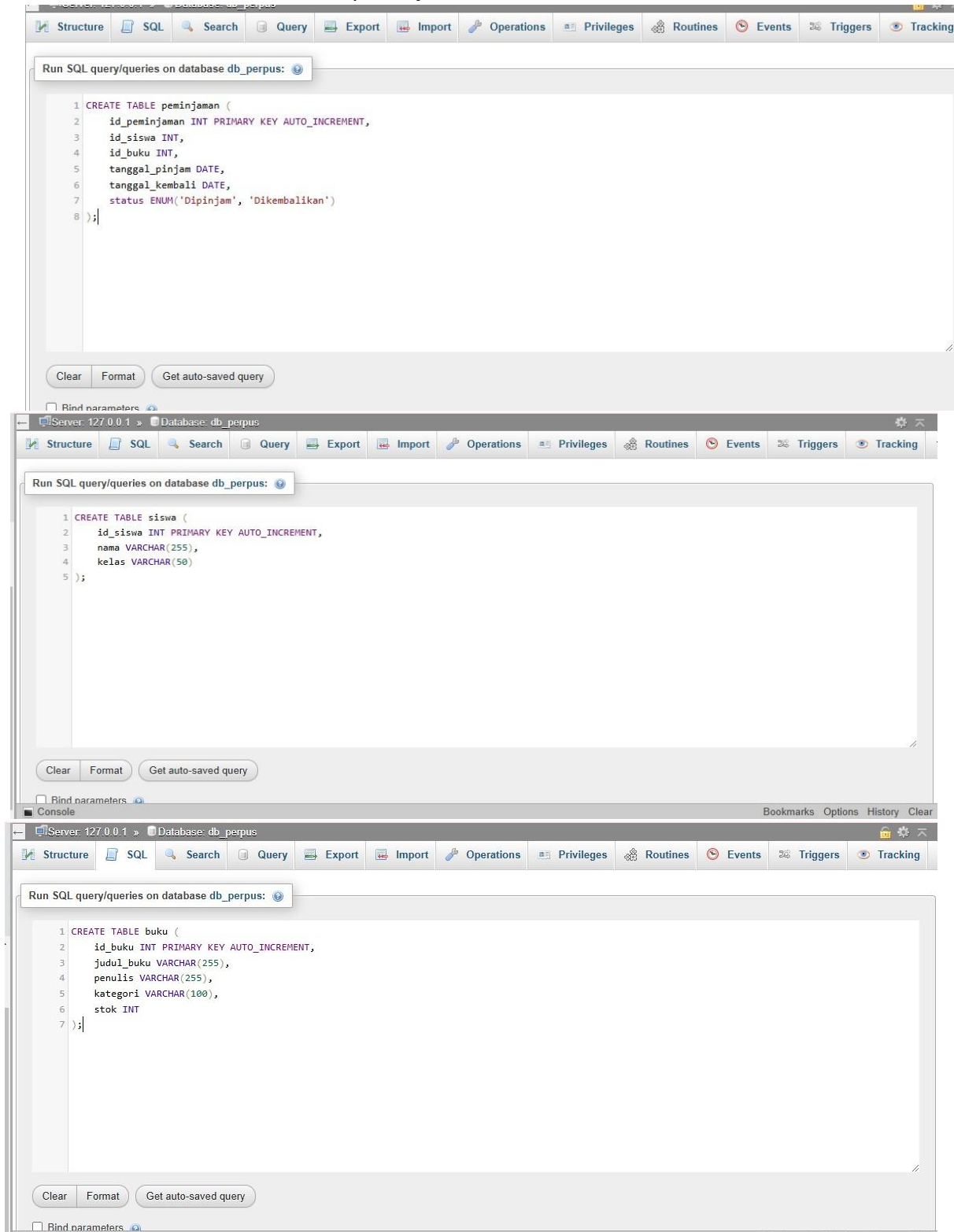
ID Peminjaman	ID Siswa	ID Buku	Tanggal Pinjam	Tanggal Kembali	Status
1	11	2	2025-02-01	2025-02-08	Dipinjam
2	2	5	2025-01-28	2025-02-04	Dikembalikan
3	3	8	2025-02-02	2025-02-09	Dipinjam
4	4	10	2025-01-30	2025-02-06	Dikembalikan
5	5	3	2025-01-25	2025-02-01	Dikembalikan
6	15	7	2025-02-01	2025-02-08	Dipinjam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

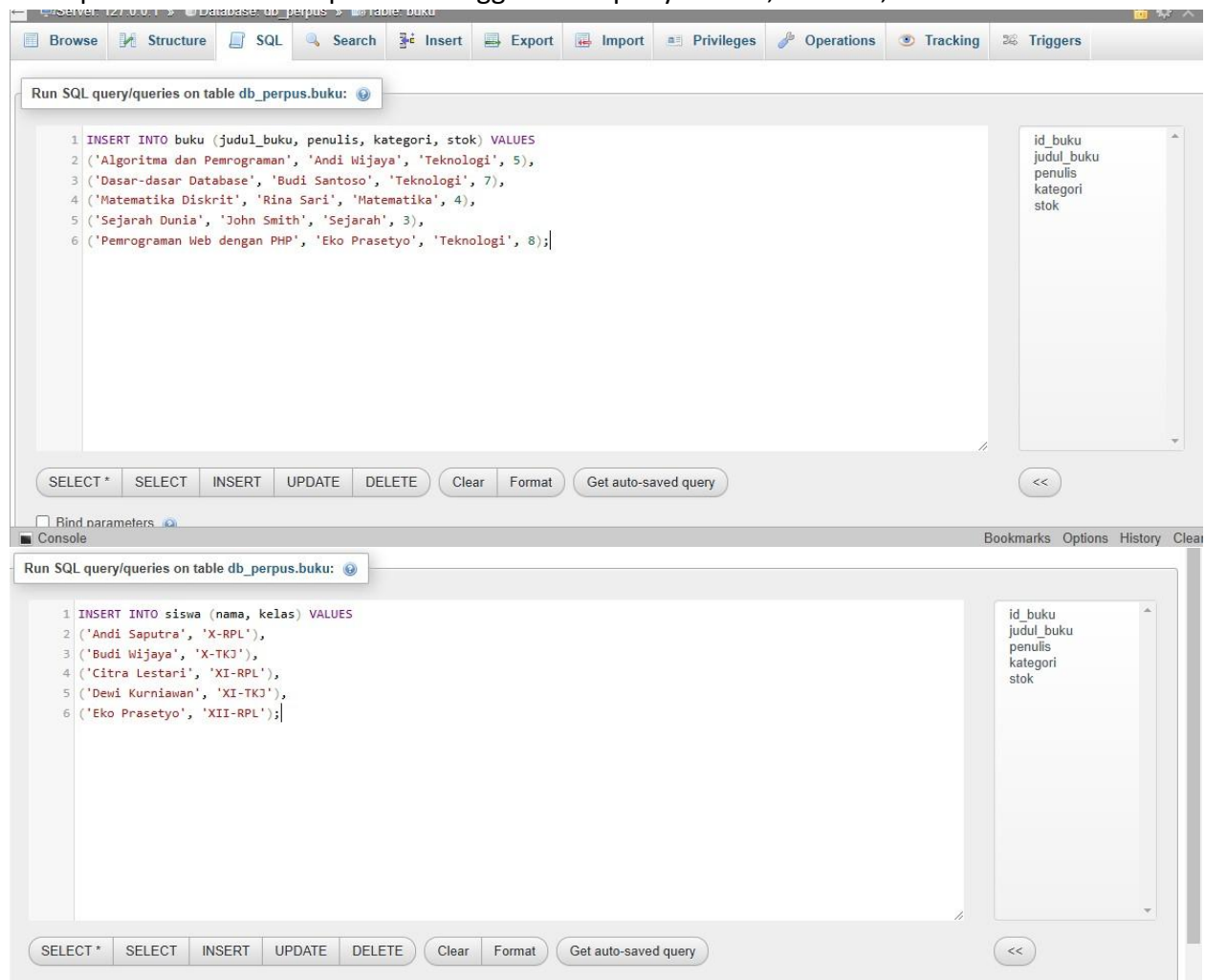
1. Buatlah database dengan nama db\_perpus.

```
1 CREATE DATABASE db_perpus;
```

## 2. Buatlah table buku, siswa dan peminjaman.



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



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

#### TABEL BUKU

```
1 DELIMITER //
2 CREATE PROCEDURE INSERTBUKU(
3     IN judul_buku VARCHAR(255),
4     IN penulis VARCHAR(255),
5     IN kategori VARCHAR(100),
6     IN stok INT
7 )
8 BEGIN
9     INSERT INTO buku (judul_buku,penulis,kategori,stok)VALUES
10     (judul_buku,penulis,kategori,stok);
11 END //
```

Run SQL query/queries on table db\_perpus.buku: ⓘ

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

## TABEL SISWA

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

## TABEL PEMINJAMAN

Run SQL query/queries on table db\_perpus.peminjaman: ⓘ

```
1 CALL INSERTPEMINJAMAN(6, 6, '2025-02-01', '2025-02-08', 'Dipinjam');
2 CALL INSERTPEMINJAMAN(7, 7, '2025-01-29', '2025-02-05', 'Dikembalikan');
3 CALL INSERTPEMINJAMAN(8, 8, '2025-02-03', '2025-02-10', 'Dipinjam');
4 CALL INSERTPEMINJAMAN(9, 9, '2025-01-27', '2025-02-03', 'Dikembalikan');
5 CALL INSERTPEMINJAMAN(10, 10, '2025-02-01', '2025-02-08', 'Dipinjam');
```

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

## BUKU

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0202 seconds.)

```
CREATE PROCEDURE UpdateBuku(IN id INT, IN stok_baru INT) BEGIN UPDATE buku SET stok = stok_baru WHERE id_buku = id; END;
```

[Edit inline](#) [\[ Edit \]](#) [\[ Create PHP code \]](#)

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0258 seconds.)

```
CREATE PROCEDURE DeleteBuku(IN id INT) BEGIN DELETE FROM buku WHERE id_buku = id; END;
```

[\[ Edit inline \]](#) [\[ Edit \]](#) [\[ Create PHP code \]](#)



## PEMINJAMAN

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0206 seconds.)

```
CREATE PROCEDURE UpdatePeminjaman(IN id INT, IN status_baru ENUM('Dipinjam','Dikembalikan')) BEGIN UPDATE peminjaman SET status = status_baru WHERE id_peminjaman = id; END;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0199 seconds.)

```
CREATE PROCEDURE DeletePeminjaman(IN id INT) BEGIN DELETE FROM peminjaman WHERE id_peminjaman = id; END;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

## SISWA

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0238 seconds.)

```
CREATE PROCEDURE UpdateSiswa(IN id INT, IN nama_baru VARCHAR(255), IN kelas_baru VARCHAR(50)) BEGIN UPDATE siswa SET nama = nama_baru, kelas = kelas_baru WHERE id_siswa = id; END;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0249 seconds.)

```
CREATE PROCEDURE DeleteSiswa(IN id INT) BEGIN DELETE FROM siswa WHERE id_siswa = id; END;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

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

## BUKU

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0188 seconds.)

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

[ Edit inline ] [ Edit ] [ Create PHP code ]

## PEMINJAMAN

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0214 seconds.)

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

[ Edit inline ] [ Edit ] [ Create PHP code ]

## SISWA

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0246 seconds.)

```
CREATE PROCEDURE GetAllSiswa() BEGIN SELECT * FROM siswa; END;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

7. Stok buku pada saat dipinjam berkurang secara otomatis.

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0277 seconds.)

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

[ Edit inline ] [ Edit ] [ Create PHP code ]

8. Stok buku pada saat dikembalikan bertambah secara otomatis.

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0277 seconds.)

```
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;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

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

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0206 seconds.)

```
CREATE PROCEDURE KembalikanBuku(IN id_peminjaman INT) BEGIN UPDATE peminjaman SET status = 'Dikembalikan', tanggal_kembali = CURDATE() WHERE id_peminjaman = id_peminjaman; END;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

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

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0254 seconds.)

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

[ Edit inline ] [ Edit ] [ Create PHP code ]

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

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0233 seconds.)

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

[ Edit inline ] [ Edit ] [ Create PHP code ]

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

✓ MySQL returned an empty result set (i.e. zero rows). (Query took 0.0234 seconds.)

```
CREATE PROCEDURE GetAllBukuWithPeminjaman() BEGIN SELECT b.id_buku, b.judul_buku, b.penulis, b.kategori, b.stok, IFNULL(p.id_peminjaman, 'Belum Pernah Dipinjam') AS status_peminjaman FROM buku b LEFT JOIN peminjaman p ON b.id_buku = p.id_buku; END;
```

[ Edit inline ] [ Edit ] [ Create PHP code ]

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