

# 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 ~



# Dokumentasi Query Database Perpustakaan

## 1. Membuat Database db\_perpus

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0005 detik.)

```
CREATE DATABASE db_perpus;
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

### Penjelasan:

Kita bikin database baru dengan nama db\_perpus.

## 2. Membuat Tabel buku, siswa, dan peminjaman

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0006 detik.)

```
CREATE TABLE buku ( idBuku int PRIMARY KEY AUTO_INCREMENT, judulBuku varchar(50), penulis varchar(50), kategori varchar(50), stok int);
```

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0003 detik.)

```
CREATE TABLE siswa ( idSiswa int PRIMARY KEY AUTO_INCREMENT, nama varchar(50), kelas varchar(50));
```

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0004 detik.)

```
CREATE TABLE peminjaman ( idPeminjaman int PRIMARY KEY AUTO_INCREMENT, idSiswa int, idBuku int, tanggalPeminjaman date, tanggalKembali date, _status varchar(50) );
```

### Penjelasan:

- buku: Buat nyimpen info buku di perpustakaan.
- siswa: Data siswa yang bisa pinjam buku.
- peminjaman: Nyatet siapa yang minjem buku, kapan pinjemnya, dan statusnya (masih dipinjam atau udah dikembalikan).

## 3. Insert Data ke Tabel

✓ 5 baris ditambahkan.

Nomor baris baru: 6 (Pencarian dilakukan dalam 0,0015 detik.)

```
INSERT into buku (idBuku, judulBuku, penulis, kategori, stok) VALUES (1, "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);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ 5 baris ditambahkan.  
Nomor baris baru: 6 (Pencarian dilakukan dalam 0,0095 detik.)

```
INSERT INTO siswa (idSiswa, nama, kelas) VALUES (1, "Andi Saputra", "X-RPL"), ("", "Budi Wijaya", "X-TKJ"), ("", "Citra Lestari", "X-RPL"), ("", "Dewi Kurniawan", "X-TKJ"), ("", "Eko Prasetyo", "X-RPL");
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ 5 baris ditambahkan.  
Nomor baris baru: 6 (Pencarian dilakukan dalam 0,0018 detik.)

```
INSERT INTO peminjaman ( idPeminjaman, idSiswa, idBuku, tanggalPeminjaman, tanggalKembali, _status) VALUES (1, 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");
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

## Penjelasan:

Masukin beberapa buku biar ada datanya di perpustakaan.

## 4. Stored Procedure Insert 10 Record

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0044 detik.)

```
CREATE PROCEDURE insert_buku( _judulBuku varchar(50), _penulis varchar(50), _kategori varchar(50), _stok int ) BEGIN INSERT into buku (judulBuku,penulis,kategori,stok) values (_judulBuku,_penulis,_kategori,_stok); END;
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0028 detik.)

```
CALL insert_buku("Sistem Operasi","Dian Kurniawan", "Teknologi", 6);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0015 detik.)

```
CALL insert_buku("Jaringan Komputer","Ahmad Fauzi", "Teknologi",5);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0016 detik.)

```
CALL insert_buku("Cerita Rakyat Nusantara","Iestari Dewi", "Sastra",9);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0013 detik.)

```
CALL insert_buku("Bahasa Inggris untuk Pemula","Jane Doe", "Bahasa",10);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0015 detik.)

```
CALL insert_buku("Biologi Dasar","Budi Rahman", "Sains",7);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0015 detik.)

```
CALL insert_buku("Kimia Organik","Siti Aminah", "Sains",5);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0012 detik.)

```
CALL insert_buku("Teknik Elektro","Ridwan Hakim", "Teknik",6);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0012 detik.)

```
CALL insert_buku("Fisika Modern","Albert Einstein", "Sains",4);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0013 detik.)

```
CALL insert_buku("Manajemen Waktu","steven Covey", "Pengembangan",8);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0014 detik.)

```
CALL insert_buku("Strategi Belajar Efektif","Tony Buzan", "Pendidikan",6);
```

[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0039 detik.)</p> <pre>CREATE PROCEDURE insert_siswa( _nama varchar(50), _kelas varchar(50) ) BEGIN INSERT into siswa (nama,kelas) values (_nama, _kelas); END;</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0014 detik.)</p> <pre>CALL insert_siswa("Farhan Maulana", "XII-TKJ");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0015 detik.)</p> <pre>CALL insert_siswa("Gita Permata", "X-RPL");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0013 detik.)</p> <pre>CALL insert_siswa("Hadi Sucipto", "X-TKJ");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0015 detik.)</p> <pre>CALL insert_siswa("Intan Permadi", "XI-RPL");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0013 detik.)</p> <pre>CALL insert_siswa("Joko Santoso", "XI-TKJ");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0013 detik.)</p> <pre>CALL insert_siswa("Kartika Sari", "XII-RPL");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0020 detik.)</p> <pre>CALL insert_siswa("Lintang Putri", "XII-TKJ");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0014 detik.)</p> <pre>CALL insert_siswa("Muhammad Rizky", "X-RPL");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0020 detik.)</p> <pre>CALL insert_siswa("Novi Andriana", "X-TKJ");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>
<p>✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0020 detik.)</p> <pre>CALL insert_siswa("Olivia Hernanda", "XI-RPL");</pre> <p>[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]</p>

```

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0035 detik.)
CREATE PROCEDURE insert_peminjaman( _idSiswa int, _idBuku int, _tanggalPeminjaman date, _tanggalKembali date, _Status varchar(50) ) BEGIN INSERT into peminjaman (idSiswa,idBuku,tanggalPeminjaman,tanggalKembali,_status) values (_idSiswa,_idBuku,_tanggalPeminjaman,_tanggalKembali,_Status); END;
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0015 detik.)
CALL insert_peminjaman(15, 7, '2025-02-01', '2025-02-08', 'Dipinjam');
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0015 detik.)
CALL insert_peminjaman(7, 1, '2025-01-29', '2025-02-05', 'Dikembalikan');
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0015 detik.)
CALL insert_peminjaman(8, 9, '2025-02-03', '2025-02-10', 'Dipinjam');
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0024 detik.)
CALL insert_peminjaman(13, 4, '2025-01-27', '2025-02-03', 'Dikembalikan');
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0014 detik.)
CALL insert_peminjaman(10, 11, '2025-02-01', '2025-02-08', 'Dipinjam');
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

```

### Penjelasan:

Biar bisa insert 10 buku langsung tanpa ngetik ulang.

## 5. Stored Procedure Update & Delete

```

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0024 detik.)
CREATE PROCEDURE update_buku( _idBuku int, _judulBuku varchar(50), _penulis varchar(50), _kategori varchar(50), _stok int) BEGIN UPDATE buku SET judulBuku = _judulBuku, penulis = _penulis, kategori = _kategori, stok = _stok WHERE idBuku = _idBuku; END;
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0158 detik.)
CREATE PROCEDURE update_siswa( _idSiswa INT, _nama VARCHAR(50), _kelas VARCHAR(50) ) BEGIN UPDATE siswa SET nama = _nama, kelas = _kelas WHERE idSiswa = _idSiswa; END;
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0039 detik.)
CREATE PROCEDURE update_peminjaman( _idPeminjaman INT, _idSiswa INT, _idBuku INT, _tanggalPeminjaman DATE, _tanggalKembali DATE, _Status VARCHAR(50) ) BEGIN UPDATE peminjaman SET idSiswa = _idSiswa, idBuku = _idBuku, tanggalPeminjaman = _tanggalPeminjaman, tanggalKembali = _tanggalKembali, Status = _Status WHERE idPeminjaman = _idPeminjaman; END;
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0109 detik.)
CREATE PROCEDURE delete_buku( _idBuku INT ) BEGIN DELETE FROM buku WHERE idBuku = _idBuku; END;
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0037 detik.)
CREATE PROCEDURE delete_siswa( _idSiswa INT ) BEGIN DELETE FROM siswa WHERE idSiswa = _idSiswa; END;
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0218 detik.)
CREATE PROCEDURE delete_peminjaman(_idPeminjaman INT) BEGIN DELETE FROM peminjaman WHERE idPeminjaman = _idPeminjaman; END;
[ Edit dikotak ] [ Ubah ] [ Buat kode PHP ]

```

### Penjelasan:

Update stok buku atau hapus buku kalau udah nggak ada.

## 6. Stored Procedure Tampilkan Semua Data

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0193 detik.)

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

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0152 detik.)

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

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0229 detik.)

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

### Penjelasan:

Buat lihat semua data.

## 7. Trigger Stok Berkurang Saat Dipinjam

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0214 detik.)

```
CREATE TRIGGER KurangiStok AFTER INSERT ON peminjaman FOR EACH ROW BEGIN UPDATE buku SET stok = stok - 1 WHERE idBuku = NEW.idBuku; END;
```

### Penjelasan:

Setiap kali ada peminjaman baru, stok bukunya langsung berkurang otomatis.

## 8. Trigger Stok Bertambah Saat Dikembalikan

✓ MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0053 detik.)

```
CREATE TRIGGER TambahStok AFTER UPDATE ON peminjaman FOR EACH ROW BEGIN IF NEW._status = 'Dikembalikan' THEN UPDATE buku SET stok = stok + 1 WHERE idBuku = NEW.idBuku; END IF;
```

### Penjelasan:

Kalau status peminjaman berubah jadi "Dikembalikan", stok buku bakal nambah otomatis.

## 9. Stored Procedure Pengembalian Buku

MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0041 detik.)

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

### Penjelasan:

Biar pas buku dikembalikan, tanggalnya langsung pakai tanggal hari ini (CURRENT DATE).

## 10. Menampilkan Siswa yang Pernah Meminjam

MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0041 detik.)

```
CREATE PROCEDURE SiswaPernahPinjam() BEGIN SELECT DISTINCT siswa.* FROM siswa JOIN peminjaman ON siswa.idSiswa = peminjaman.idSiswa; END;
```

### Penjelasan:

Kenapa siswa. Pake Titik? Karena kita pakai JOIN dengan tabel lain (peminjaman), jadi biar SQL nggak bingung, kita spesifik pakai siswa.\* buat ambil semua data dari tabel siswa

## 11. Menampilkan Semua Siswa (Termasuk yang Tidak Pernah Meminjam)

MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0041 detik.)

```
CREATE PROCEDURE SemuaSiswa() BEGIN SELECT siswa.*, IFNULL(peminjaman.idPeminjaman, 'Belum Meminjam') AS _status FROM siswa LEFT JOIN peminjaman ON siswa.idSiswa = peminjaman.idSiswa; END;
```

### Penjelasan:

Kenapa IFNULL? Karena kalau ada siswa yang belum pernah pinjam buku, peminjaman.id\_peminjaman bakal NULL. IFNULL() dipakai buat ganti NULL jadi "Belum Meminjam" biar gampang dibaca.

## 12. Menampilkan Semua Buku (Termasuk yang Belum Pernah Dipinjam)

MySQL memberikan hasil kosong (atau nol baris). (Pencarian dilakukan dalam 0,0037 detik.)

```
CREATE PROCEDURE SemuaBuku() BEGIN SELECT buku.*, IFNULL(peminjaman.idPeminjaman, 'Belum Dipinjam') AS _status FROM buku LEFT JOIN peminjaman ON buku.idBuku = peminjaman.idBuku; END;
```

### Penjelasan:

Biar semua buku muncul, termasuk yang belum pernah dipinjam.