**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 |

1. 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 |

1. 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.

CREATE DATABASE perpustakaan;

1. Buatlah table buku, siswa dan peminjaman.

CREATE TABLE Buku (id\_buku int PRIMARY KEY AUTO\_INCREMENT,judul\_buku varchar(50),penulis varchar(50),kategori varchar(30),stok int);

CREATE TABLE Siswa (id\_siswa int PRIMARY KEY AUTO\_INCREMENT,nama varchar(50),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 varchar(50));

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

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);

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');

INSERT INTO peminjaman (id\_siswa, id\_buku, tanggal\_pinjam, tanggal\_kembali, status) VALUES

(1, 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');

1. Input 10 record di setiap table menggunakan stored procedure INSERT.

DELIMITER //

CREATE PROCEDURE insertBuku(

IN judul\_bukuBaru varchar(50),

IN penulisBaru varchar(50),

IN kategoribaru varchar(50),

IN stokBaru int

)

BEGIN

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

VALUES (judul\_bukuBaru, penulisBaru, kategoriBaru, stokBaru);

END //

DELIMITER ;;

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

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

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('Kimia Organik', 'Siti Aminah', 'Sains', 5);

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);

DELIMITER //

CREATE PROCEDURE insertSiswa(

IN namaBaru varchar(50),

IN kelasBaru varchar(50)

)

BEGIN

INSERT INTO siswa (nama. kelas)

VALUES (namaBaru, kelasBaru);

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');

DELIMITER //

CREATE PROCEDURE insertPeminjaman(

IN id\_siswaBaru int,

IN id\_bukuBaru int,

IN tanggal\_pinjamBaru date,

IN tanggal\_kembaliBaru date,

IN statusBaru varchar(50)

)

BEGIN

INSERT INTO peminjaman (id\_siswa, id\_buku, tanggal\_pinjam, tanggal\_kembali, status)

VALUES (id\_siswaBaru, id\_bukuBaru, tanggal\_pinjamBaru,tanggal\_kembaliBaru,statusBaru);

END //

CALL insertPeminjaman(15, 7, '2025-02-01', '2025-02-08', 'Dipinjam');

CALL insertPeminjaman(7, 1, '2025-01-29', '2025-02-05', 'Dikembalikan');

CALL insertPeminjaman(8, 9, '2025-02-03', '2025-02-10', 'Dipinjam');

CALL insertPeminjaman(13, 4, '2025-01-27', '2025-02-03', 'Dikembalikan');

CALL insertPeminjaman(10, 11, '2025-02-01', '2025-02-08', 'Dipinjam');

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

DELIMITER //

CREATE PROCEDURE updateBuku(

IN idBuku INT,

IN judulBaru VARCHAR(50),

IN penulisBaru VARCHAR(50),

IN kategoriBaru VARCHAR(50),

IN stokBaru INT

)

BEGIN

UPDATE buku

SET judul\_buku = judulBaru,

penulis = penulisBaru,

kategori = kategoriBaru,

stok = stokBaru

WHERE id\_buku = idBuku;

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE deleteBuku(

IN idBuku INT

)

BEGIN

DELETE FROM buku WHERE id\_buku = idBuku;

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE updateSiswa(

IN idSiswa INT,

IN namaBaru VARCHAR(50),

IN kelasBaru VARCHAR(50)

)

BEGIN

UPDATE siswa

SET nama = namaBaru,

kelas = kelasBaru

WHERE id\_siswa = idSiswa;

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE deleteSiswa(

IN idSiswa INT

)

BEGIN

DELETE FROM siswa WHERE id\_siswa = idSiswa;

END //

DELIMITER ;

DELIMITER //

CREATE PROCEDURE updatePeminjaman(

IN p\_id\_peminjaman INT,

IN p\_id\_siswa INT,

IN p\_id\_buku INT,

IN p\_tanggal\_pinjam DATE,

IN p\_tanggal\_kembali DATE,

IN p\_status VARCHAR(50)

)

BEGIN

UPDATE peminjaman

SET ID\_Siswa = p\_id\_siswa, id\_buku = p\_id\_buku,

Tanggal\_Pinjam = p\_tanggal\_pinjam,

Tanggal\_Kembali = p\_tanggal\_kembali,

Status = p\_status

WHERE ID\_Peminjaman = p\_id\_peminjaman;

END //

DELIMITER//

CREATE PROCEDURE deletePeminjaman(

IN p\_id\_peminjaman INT

)

BEGIN

DELETE FROM peminjaman WHERE ID\_Peminjaman = p\_id\_peminjaman;

END //

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

DELIMITER //

CREATE PROCEDURE showBuku()

BEGIN

SELECT \* FROM buku;

END//

DELIMITER //

CREATE PROCEDURE showSiswa()

BEGIN

SELECT \* FROM siswa;

END//

DELIMITER //

CREATE PROCEDURE showPeminjaman()

BEGIN

SELECT \* FROM peminjaman;

END//

1. Stok buku pada saat dipinjam berkurang secara otamatis.

DELIMITER //

CREATE TRIGGER after\_peminjaman\_insert

AFTER INSERT ON peminjaman

FOR EACH ROW

BEGIN

UPDATE buku

SET stok = stok - 1

WHERE id\_buku = NEW.id\_buku;

END;

//

1. Stok buku pada saat dikembalikan bertambah secara otomatis.

DELIMITER //

CREATE TRIGGER after\_peminjaman\_update

AFTER UPDATE ON peminjaman

FOR EACH ROW

BEGIN

-- Jika status berubah menjadi "Dikembalikan", tambahkan stok

IF OLD.status = 'Dipinjam' AND NEW.status = 'Dikembalikan' THEN

UPDATE buku

SET stok = stok + 1

WHERE id\_buku = NEW.id\_buku;

END IF;

END//

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

DELIMITER //

CREATE PROCEDURE returnBuku(

IN p\_id\_peminjaman INT

)

BEGIN

UPDATE peminjaman

SET tanggal\_kembali = CURRENT\_DATE,

status = 'Dikembalikan'

WHERE id\_peminjaman = p\_id\_peminjaman;

END//

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

DELIMITER //

CREATE PROCEDURE showSiswaPeminjam()

BEGIN

SELECT DISTINCT s.id\_siswa, s.nama, s.kelas

FROM siswa s

JOIN peminjaman p ON s.id\_siswa = p.id\_siswa;

END;

//

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

DELIMITER //

CREATE PROCEDURE showAllSiswa()

BEGIN

SELECT s.id\_siswa, s.nama, s.kelas,

IFNULL(COUNT(p.id\_peminjaman), 0) AS jumlah\_peminjaman

FROM siswa s

LEFT JOIN peminjaman p ON s.id\_siswa = p.id\_siswa

GROUP BY s.id\_siswa, s.nama, s.kelas;

END;

//

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

DELIMITER //

CREATE PROCEDURE showAllBuku()

BEGIN

SELECT b.id\_buku, b.judul\_buku, b.penulis, b.kategori, b.stok,

IFNULL(COUNT(p.id\_peminjaman), 0) AS jumlah\_dipinjam

FROM buku b

LEFT JOIN peminjaman p ON b.id\_buku = p.id\_buku

GROUP BY b.id\_buku, b.judul\_buku, b.penulis, b.kategori, b.stok;

END;

//

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