Join Table & View

Worksheet 4

Nama : M.Irfan Hamdi

NIM : 211100065

Asal : Universitas Metamedia

SOAL 4.1

Tampilkan data berikut menggunakan join table:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Pesanan** | | | **Pelanggan** | | **Kartu** | |
| id | tanggal | total | kode | nama | nama kartu | diskon |

*SELECT ...*

*SELECT pesanan.id, pesanan.tanggal, pesanan. total, pelanggan.kode, pelanggan.nama,*

*kartu.nama as nama\_kartu, kartu.diskon*

*FROM pesanan INNER JOIN pelanggan ON pesanan.pelanggan\_id = pelanggan.id*

*INNER JOIN kartu ON pelanggan.kartu\_id = kartu.id;*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Pembelian** | | | | | **Produk** | **Vendor** | |
| id | tanggal | nomor | jumlah | harga | nama | nama | kontak |

SELECT pembelian.id, pembelian.tanggal, pembelian.nomor, pembelian.jumlah, pembelian.harga,

produk.nama, vendor.nama, vendor.kontak FROM pembelian INNER JOIN produk

ON pembelian.produk\_id = produk.id

INNER JOIN vendor ON pembelian.vendor\_id = vendor.id;

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Pesanan** | | | **Pelanggan** | **Produk** | | **Jenis produk** | **Pesanan items** | |
| id | tanggal | total | nama | kode | nama | nama | qty | harga\_jual |

SELECT pesanan.id, pesanan.tanggal, pesanan.total, pelanggan.nama,

produk.kode, produk.nama as produk, jenis\_produk.nama as jenis,

pesanan\_items.qty, pesanan\_items.harga as harga\_jual

FROM pesanan\_items

INNER JOIN produk ON pesanan\_items.produk\_id = produk.id

INNER JOIN jenis\_produk ON produk.jenis\_produk\_id = jenis\_produk.id

INNER JOIN pesanan ON pesanan\_items.pesanan\_id = pesanan.id

INNER JOIN pelanggan ON pesanan.pelanggan\_id = pelanggan.id;

SOAL 4.2

Buatlah view berdasarkan query yang menampilkan data berikut ini:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Pesanan** | | | **Pelanggan** | | **Kartu** | |
| id | tanggal | total | kode | nama | nama kartu | diskon |

*CREATE VIEW ...*

*CREATE VIEW pesanan\_pelanggan\_kartu*

*AS SELECT pesanan.id, pesanan.tanggal, pesanan. total, pelanggan.kode, pelanggan.nama,*

*kartu.nama as nama\_kartu, kartu.diskon*

*FROM pesanan INNER JOIN pelanggan ON pesanan.pelanggan\_id = pelanggan.id*

*INNER JOIN kartu ON pelanggan.kartu\_id = kartu.id;*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Pembelian** | | | | | **Produk** | **Vendor** | |
| id | tanggal | nomor | jumlah | harga | nama | nama | Kontak |

SELECT \* FROM vendor;

CREATE VIEW pembelian\_produk\_vendor

AS SELECT p.id, p.tanggal, p.nomor, p.jumlah, p.harga,

pr.nama, v.nama as nama\_vendor, v.kontak FROM pembelian p INNER JOIN produk pr

ON p.produk\_id = pr.id

INNER JOIN vendor v ON p.vendor\_id = v.id;

SELECT \* FROM pembelian\_produk\_vendor;

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Pesanan** | | | **Pelanggan** | **Produk** | | **Jenis produk** | **Pesanan items** | |
| id | tanggal | total | nama | kode | nama | nama | qty | harga\_jual |

CREATE VIEW pesanan\_pelanggan\_produk\_jenis\_item AS SELECT pesanan.id,

pesanan.tanggal, pesanan.total, pelanggan.nama, produk.kode, produk.nama as produk,

jenis\_produk.nama as jenis, pesanan\_items.qty, pesanan\_items.harga as harga\_jual

FROM pesanan\_items

INNER JOIN produk ON pesanan\_items.produk\_id = produk.id

INNER JOIN jenis\_produk ON produk.jenis\_produk\_id = jenis\_produk.id

INNER JOIN pesanan ON pesanan\_items.pesanan\_id = pesanan.id

INNER JOIN pelanggan ON pesanan.pelanggan\_id = pelanggan.id;

Soal 4.3 Transaction

1. Buatlah sebuah transaction dengan skenario-skenario statement sebagai berikut:

* Mulai transaction

START TRANSACTION;

* Insert data produk sebanyak 3 record

INSERT INTO produk (kode, nama, harga\_beli, harga\_jual, stok, min\_stok, jenis\_produk\_id) VALUES

('KS001','Kasur Empuk','3500000','5200000','15','15','1'),

('SF001','Sofa Panjang 5m','1800000','3100000','15','20','2'),

('SF002','Sofa 2m','600000','900000','40','25','2');

* Update data stok salah satu produk

UPDATE produk SET stok='25' WHERE id=36;

* Buat savepoint

SAVEPOINT update\_produk;

* Hapus salah satu data pembayaran

DELETE FROM pembayaran WHERE pembayaran.id = 4;

* Kembali ke savepoint

ROLLBACK TO update\_produk;

* Update data iuran salah satu kartu

UPDATE kartu SET iuran=5000 WHERE id=7;

* Akhiri transaction dengan commit

COMMIT;

*START TRANSACTION ...*

1. Berikan penjelasan kapan saat yang tepat menggunakan LOCK TABLES READ

*...* LOCK TABLES READ digunakan untuk mengunci tabel secara eksklusif dalam mode baca (read) untuk mencegah perubahan data oleh pengguna lain selama operasi yang memerlukan akses eksklusif pada tabel tersebut. Fungsi ini berguna ketika Anda ingin melakukan operasi yang memerlukan akses eksklusif pada tabel, seperti:

\* Membuat backup data

\* Menghapus data yang tidak diperlukan

\* Membuat indeks atau view

\* Mengubah struktur tabel

LOCK TABLES READ memungkinkan Anda untuk mengunci tabel secara eksklusif dalam mode baca, sehingga tidak ada pengguna lain yang dapat mengubah data pada tabel tersebut selama operasi Anda berlangsung. Namun, Anda masih dapat membaca data dari tabel yang terkunci.

LOCK TABLES READ tidak mengunci seluruh database, hanya tabel yang dipilih saja.

1. Berikan penjelasan kapan saat yang tepat menggunakan LOCK TABLES WRITE

*...* LOCK TABLES WRITE digunakan untuk mengunci tabel secara eksklusif dalam mode tulis (write) untuk mencegah perubahan data oleh pengguna lain selama operasi yang memerlukan akses eksklusif pada tabel tersebut. Fungsi ini berguna ketika Anda ingin melakukan operasi yang memerlukan akses eksklusif pada tabel, seperti:

- Menjalankan operasi DDL seperti ALTER TABLE untuk mengubah struktur tabel.

- Melakukan operasi yang memerlukan penulisan data besar ke tabel.

- Menjalankan operasi yang memerlukan konsistensi data tinggi.

LOCK TABLES WRITE memungkinkan Anda untuk mengunci tabel secara eksklusif dalam mode tulis, sehingga tidak ada pengguna lain yang dapat membaca atau menulis data pada tabel tersebut selama operasi Anda berlangsung. Ini memastikan konsistensi data dan mencegah konflik yang dapat terjadi jika ada operasi yang bersamaan pada tabel yang sama.

Penggunaan LOCK TABLES WRITE harus dilakukan dengan hati-hati karena dapat menyebabkan penundaan akses ke tabel oleh pengguna lain dan mengakibatkan deadlock jika tidak dikelola dengan benar. Sebaiknya digunakan dengan bijaksana dan hanya saat diperlukan.