Join Table & View

Worksheet 4

Nama : Dewa Gede Andika Andara Putra

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 |

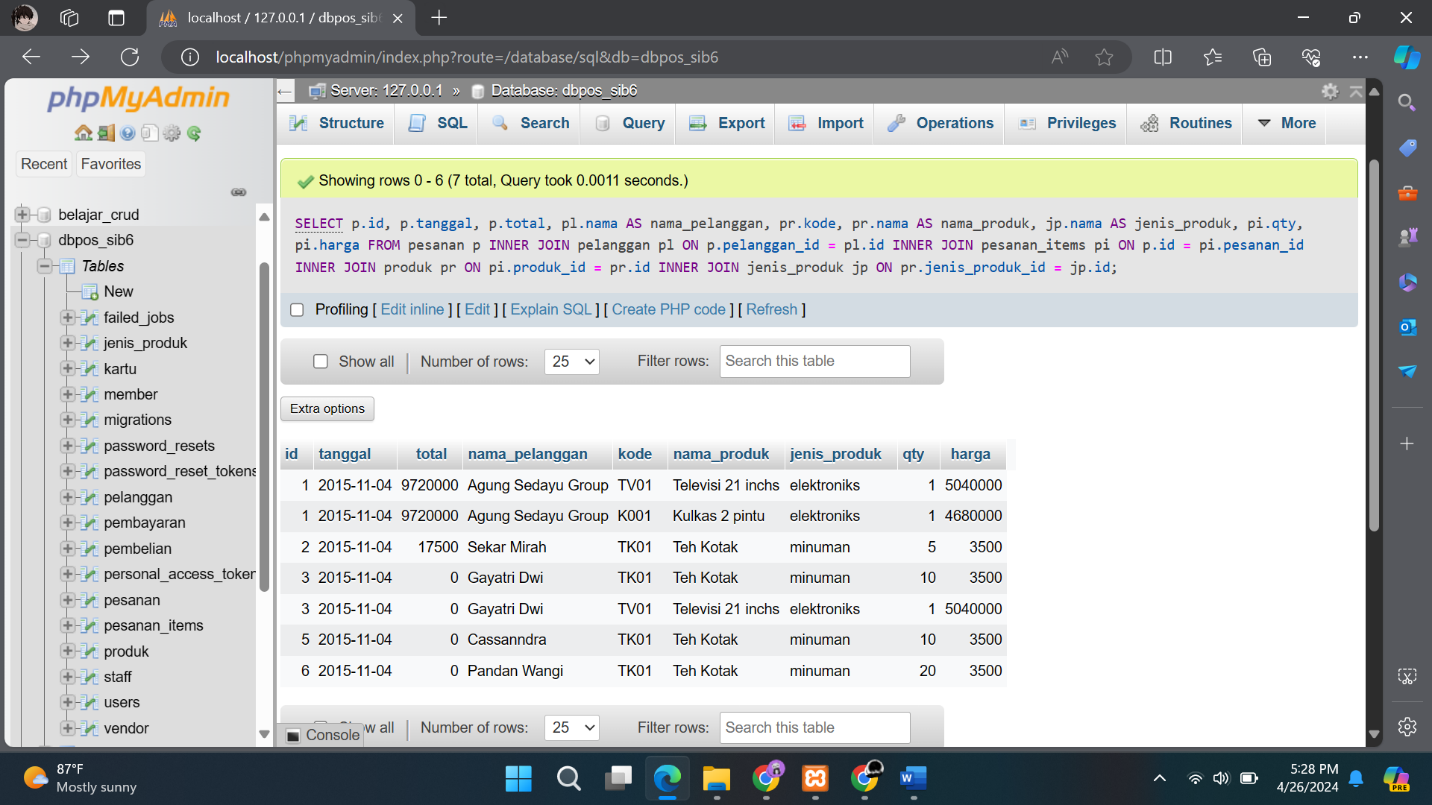
SELECT pembelian.id, pembelian.tanggal, pembelian.nomor, pembelian.jumlah, pembelian.harga,

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 p.id, p.tanggal, p.total, pl.nama AS nama\_pelanggan, pr.kode, pr.nama AS nama\_produk, jp.nama AS jenis\_produk, pi.qty, pi.harga FROM pesanan p INNER JOIN pelanggan pl ON p.pelanggan\_id = pl.id INNER JOIN pesanan\_items pi ON p.id = pi.pesanan\_id INNER JOIN produk pr ON pi.produk\_id = pr.id INNER JOIN jenis\_produk jp ON pr.jenis\_produk\_id = jp.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

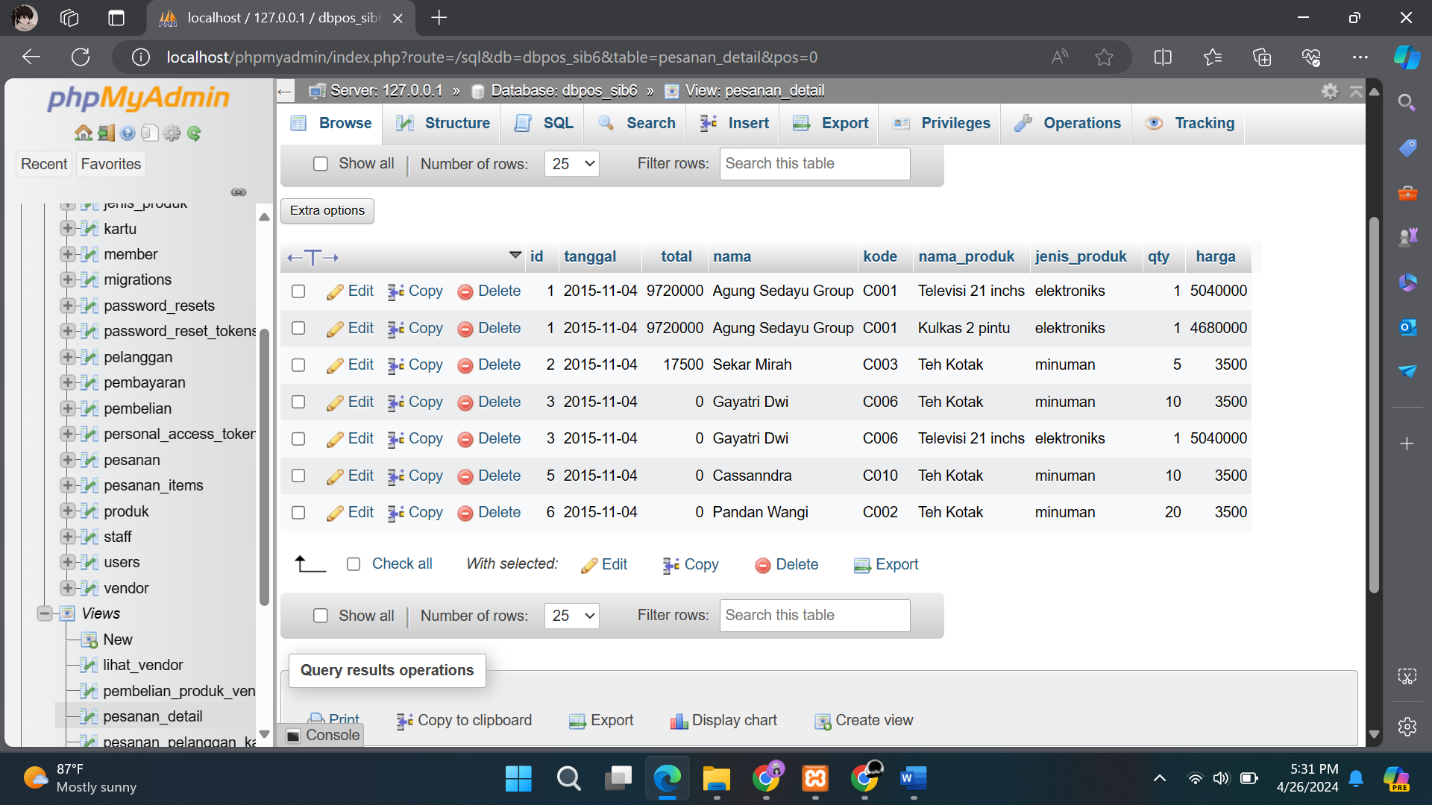
ON p.produk\_id = pr.id

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\_detail AS SELECT p.id, p.tanggal, p.total, pel.nama, pel.kode, pr.nama AS nama\_produk, jp.nama AS jenis\_produk, pi.qty, pi.harga FROM pesanan p JOIN pelanggan pel ON p.pelanggan\_id = pel.id JOIN pesanan\_items pi ON p.id = pi.pesanan\_id JOIN produk pr ON pi.produk\_id = pr.id JOIN jenis\_produk jp ON pr.jenis\_produk\_id = jp.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

MariaDB [dbpos\_sib6]> INSERT INTO jenis\_produk (nama) VALUES

-> ('Pakaian'),

-> ('Keyboard'),

-> ('Mouse');

Query OK, 3 rows affected (0.005 sec)

Records: 3 Duplicates: 0 Warnings: 0

* Update data stok salah satu produk

MariaDB [dbpos\_sib6]> UPDATE produk

-> SET stok = 50

-> WHERE id IN (

-> SELECT id

-> FROM produk

-> WHERE jenis\_produk\_id = (

-> SELECT id

-> FROM jenis\_produk

-> WHERE nama = 'Pakaian')

-> );

Query OK, 0 rows affected (0.041 sec)

Rows matched: 0 Changed: 0 Warnings: 0

* Buat savepoint

MariaDB [dbpos\_sib6]> SAVEPOINT update\_stok\_pakaian;

Query OK, 0 rows affected (0.001 sec)

* Hapus salah satu data pembayaran

MariaDB [dbpos\_sib6]> DELETE FROM jenis\_produk WHERE id=10;

Query OK, 1 row affected (0.001 sec)

* Kembali ke savepoint

MariaDB [dbpos\_sib6]> ROLLBACK TO SAVEPOINT update\_stok\_pakaian;

Query OK, 0 rows affected (0.004 sec)

* Update data iuran salah satu kartu

MariaDB [dbpos\_sib6]> UPDATE kartu SET diskon = 0.1 WHERE nama = 'Silver';

Query OK, 1 row affected (0.001 sec)

Rows matched: 1 Changed: 1 Warnings: 0

* Akhiri transaction dengan commit

MariaDB [dbpos\_sib6]> COMMIT;

Query OK, 0 rows affected (0.003 sec)

*START TRANSACTION ...*

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

LOCK TABLES READ digunakan ketika kita ingin membaca data dari satu atau beberapa tabel dan memastikan tidak ada perubahan data selama kita membaca. Ini berguna saat kita melakukan query kompleks atau reporting yang memerlukan konsistensi data selama proses berlangsung. Namun, LOCK TABLES READ akan memblokir operasi DML (INSERT, UPDATE, DELETE) pada tabel yang dikunci sampai kunci dilepas.

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

LOCK TABLES WRITE digunakan ketika kita ingin mengubah data pada satu atau beberapa tabel dan memastikan tidak ada aktivitas lain yang mengakses tabel tersebut selama proses berlangsung. Ini berguna saat kita melakukan operasi DML yang berat atau operasi yang melibatkan beberapa tabel secara bersamaan. LOCK TABLES WRITE akan memblokir semua operasi lain (DML dan SELECT) pada tabel yang dikunci sampai kunci dilepas.

Baik LOCK TABLES READ maupun WRITE harus digunakan dengan hati-hati karena dapat mengganggu operasi database lain yang memerlukan akses ke tabel yang sama. Lock harus dilepas setelah proses selesai untuk menghindari deadlock atau kekacauan sistem.