|  |  |  |
| --- | --- | --- |
| Nama | : | Farida Angguntina |
| NIM | : | L200150006 |
| Kelas | : | A |

**PERBAIKAN NILAI UTS**

**SISTEM MANAGEMEN BASISDATA**

1. (a) Buatlah semua relasi antar table pada database ‘classicmodels’, dan terangkan jenis relasinya.

(b) Jelaskan fungsi masing-masing table (digunakan untuk menyimpan data apa?)

**Jawab:**

1. Relasi dan Jenis Relasi antar Tabel
2. Customers 🡪 Employees: Customers **– layan –** Employees [n:1]

Satu customer dilayani oleh satu pegawai. Satu pegawai dapat melayani banyak customer.

1. Employees 🡪 Offices: Employees – **tempat –** Offices [n:1]

Satu pegawai hanya menempati satu kantor. Satu kantor ditempati oleh banyak pegawai.

1. Employees 🡪 Employees: Employees – **pimpin –** Employees [n:1]

Satu pegawai dipimpin oleh satu pegawai berkedudukan di atasnya. Satu pegawai dapat memimpin banyak pegawai berkedudukan di bawahnya.

1. Products 🡪 Product Lines: Product – **klasifikasi –** ProductLines [n:1]

Satu produk memiliki satu product line (label). Satu product line dapat digunakan oleh banyak produk.

1. Customers 🡪 Orders: Customers – **lakukan –** Orders [n:1]

Satu customer dapat melakukan order lebih dari satu. Satu order terdaftar dilakukan oleh satu customer.

1. Customers 🡪 Payments: Customers – **lakukan –** Payments [n:1]

Satu customer melakukan payment lebih dari satu jika memiliki order lebih dari satu juga. Namun satu payment yang terbayarkan hanya dilakukan oleh satu customer.

1. Orders 🡪 Products: Orders – **contains –** Products [n:m]

Satu order terdiri dari banyak produk begitu pula satu produk akan memiliki banyak order. Dari keduanya muncul tabel baru yaitu **orderdetails**.

(b) Fungsi Tabel

1. Orders: Data pemesanan barang yang dilakukan oleh pelangggan.

2. Employees: Data pegawai perusahaan.

3. Offices: Data kantor yang dimiliki perusahaan.

4. Customers: Data pelanggan perusahaan.

5. Order Details: Data spesifik dari order pelanggan.

6. Payments: Data pembayaran pelanggan atas order.

7. Product Lines: Data deskripsi dari label barang.

8. Products: Data barang yang dijual perusahaan.

1. Buatlah query untuk menampilkan daftar barang yang terjual pada tanggal tertentu. Kolom output minimal: kode, nama barang, dan jumlah barang terjual.

**Jawab:**

SELECT products.productCode, products.productName, orderdetails.quantityOrdered, orderdetails.priceEach

FROM products, orders, orderdetails

WHERE products.productCode = orderdetails.productCode AND

orders.orderNumber = orderdetails.orderNumber AND

orders.orderDate = "2003-01-06" AND

NOT orders.status = "Cancelled";

1. Sama dengan soal kedua namun dalam bentuk store procedure dengan tanggal sebagai input dan output adalah nilai uang yang diperoleh di tanggal tersebut dan tabel daftar barang.

**Jawab:**

DELIMITER //

CREATE PROCEDURE orderbydate (IN tanggal varchar(10), OUT total int)

BEGIN

SELECT SUM(orderdetails.quantityOrdered\*orderdetails.priceEach) INTO total

FROM orders, orderdetails

orders.orderNumber = orderdetails.orderNumber AND

orders.orderDate) = tanggal AND

NOT orders.status = "Cancelled";

SELECT products.productCode, products.productName, orderdetails.quantityOrdered, orderdetails.priceEach

FROM products, orders, orderdetails

WHERE products.productCode = orderdetails.productCode AND

orders.orderNumber = orderdetails.orderNumber AND

orders.orderDate) = tanggal AND

NOT orders.status = "Cancelled";

END //

DELIMITER;

1. Buatlah store procedure untuk menampilkan daftar customer yang melakukan pembelian (order) selama satu minggu terakhir, sehingga output adalah jumlah pelanggan dan daftar pelanggan.

**Jawab:**

DELIMITER //

CREATE PROCEDURE customerdata (IN bulan int(2), IN tahun int(4), OUT jumlah int)

BEGIN

SELECT COUNT(DISTINCT orders.customerNumber) INTO jumlah

FROM orders

WHERE month(orderDate) = bulan AND

year(orderDate) = tahun AND

day(orders.orderDate) >= 24;

SELECT customers.customerNumber, customers.customerName

FROM customers, orders

WHERE customer.customerNumber = orders.customerNumber AND

month(orders.orderDate) = bulan AND

year(orders.orderDate) = tahun AND

day(orders.orderDate) >= 24;

END //

DELIMITER;