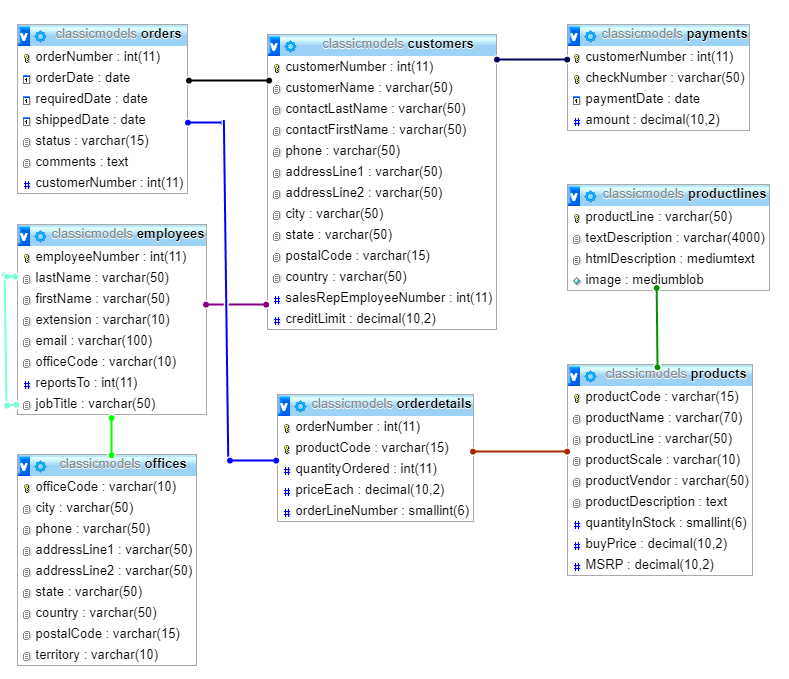
**REMIDI UJIAN TENGAH SEMESTER GANJIL 2017/2018**

**SISTEM MANAGEMENT BASISDATA**

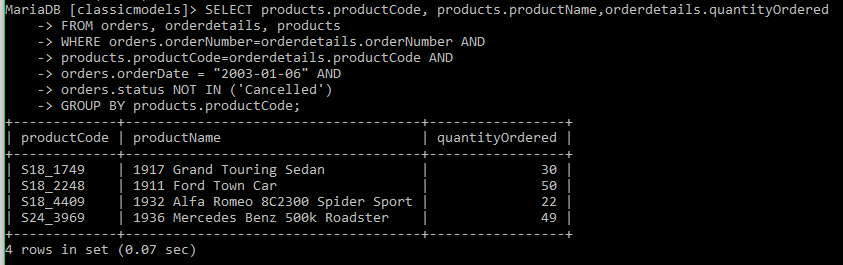
1. Table relasi dan fungsi masing-masing table
2. Tabel relasi

****

* **Relasi dan Kardinalitas**
* **Employees “menempati” Offices**
* Kardinalitas: many-to-one (n:1)
* Alasan: Karena 1 offices bisa ditempati oleh banyak employees
* Primary Key table employees menjadi Foreign Key di dalam table Offices
* **Employees “melayani” customers**
* Kardinalitas: one-to-many (1:n)
* Alasan: Karena 1 employees dapat melayani banyak customers
* Primary Key table Employees menjadi Foreign Key di dalam table customers.
* **Customers “melakukan” orders**
* Kardinalitas: one-to-many (1:n)
* Alasan: Karena 1 customers dapat melakukan banyak orders.
* Primary Key table customers menjadi Foreign Key di dalam table orders.
* **Customers “melakukan” payment**
* Kardinalitas: one-to-many (1:n)
* Alasan: Karena 1 customers dapat melakukan pembayaran (payment) dari banyak transaksi (order).
* Primary Key table order menjadi Foreign Key table payment.
* **Employees “melaporkan” ke Employees (pimpinan)**
* Kardinalitas: many-to-one (n:1)
* Alasan: Karena banyak employees akan melaporkan suatu hasil kepada seorang pimpinan.
* Terdapat Foreign Key “reportsTo” di dalam table employees.
* **Orders “memiliki” orderdetails**
* Kardinalitas: one-to-one (1:1)
* Alasan: Karena satu order hanya akan memiliki tepat satu detail order.
* Primary Key table orders menjadi Foreign Key table orderdetails.
* **Orderdetails “memiliki” Products**
* Kardinalitas: one-to-many (1:n)
* Alasan: Karena data dalam orderdetails terdapat data product yang dipesan oleh customers.
* Primary Key table Products menjadi Foreign Key table Orderdetails.
* **Products “terklasifikasi” dalam productLines**
* Kardinalitas: many-to-one (n:1)
* Alasan: Karena beberapa product yang dijual dapat diklasifikasikan kedalam satu kategori yang sama.
* Primary Key table products menjadi Foreign Key table productLines.

1. Fungsi masing-masing table

* **Customers -->** Berisi data pelanggan.
* **Offices -->** Berisi data lokasi kantor.
* **Employees -->** Berisi data pegawai.
* **Payment -->** Berisi data pembayaran yang dilakukan pelanggan ketika melakukan order.
* **Orders -->** Berisi data pesanan pelanggan.
* **Orderdetails -->** Berisi detail pesanan pelanggan.
* **Products -->** Berisi data product yang dijual
* **ProductLines -->** Berisi kategori pengelompokan product.

1. Query untuk menampilkan daftar barang yang terjual pada tanggal tertentu.

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

FROM orders, orderdetails, products

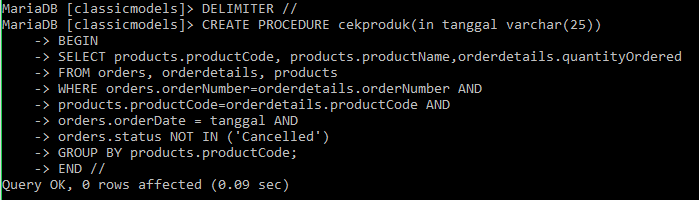
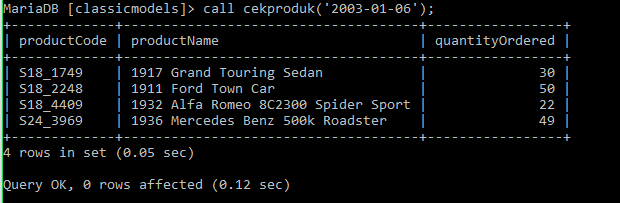
WHERE orders.orderNumber=orderdetails.orderNumber AND

products.productCode = orderdetails.productCode AND orders.orderDate = "2003-01-06" AND

orders.status NOT IN ('Cancelled')

GROUP BY products.productCode;

1. Store procedure soal nomor 2 dengan output nilai uang yang diperoleh tanggal tersebut dan table daftar barang.



DELIMITER //

CREATE PROCEDURE cekproduk(in tanggal varchar(25))

BEGIN

SELECTproducts.productCode, products.productName,orderdetails.quantityOrdered

FROM orders, orderdetails, products

WHERE orders.orderNumber=orderdetails.orderNumber AND

products.productCode=orderdetails.productCode AND

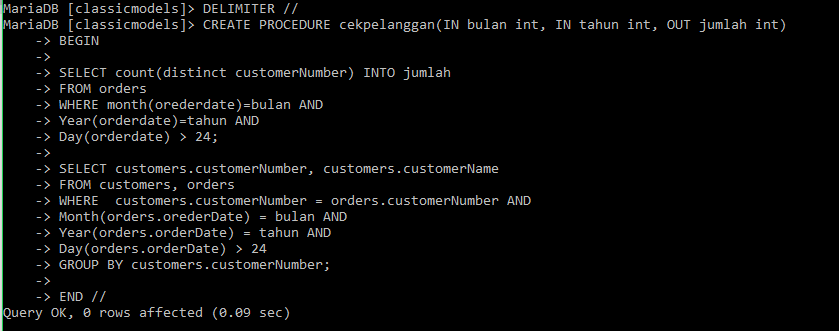
orders.orderDate = "2003-01-06" AND

orders.status NOT IN ('Cancelled')

GROUP BY products.productCode;

END //

DELIMITER;

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

DELIMITER //

CREATE PROCEDURE cekpelanggan(IN bulan int, IN tahun int, OUT jumlah int)

BEGIN

SELECT count(distinct customerNumber) INTO jumlah

FROM orders

WHERE month(orederdate)=bulan AND

Year(orderdate)=tahun AND

Day(orderdate) > 24;

SELECT customers.customerNumber, customers.customerName

FROM customers, orders

WHERE customers.customerNumber=orders.customerNumber AND

Month(orders.orederDate) = bulan AND

Year(orders.orderDate) = tahun AND

Day(orders.orderDate) > 24

GROUP BY customers.customerNumber;

END //

DELIMITER;